frontiers | Research Topics

Deep learning for marine science

Edited by
Haiyong Zheng, Mark C. Benfield, Hongsheng Bi and Xuemin Cheng

Published in
Frontiers in Marine Science



FRONTIERS EBOOK COPYRIGHT STATEMENT

The copyright in the text of individual articles in this ebook is the property of their respective authors or their respective institutions or funders. The copyright in graphics and images within each article may be subject to copyright of other parties. In both cases this is subject to a license granted to Frontiers.

The compilation of articles constituting this ebook is the property of Frontiers.

Each article within this ebook, and the ebook itself, are published under the most recent version of the Creative Commons CC-BY licence. The version current at the date of publication of this ebook is CC-BY 4.0. If the CC-BY licence is updated, the licence granted by Frontiers is automatically updated to the new version.

When exercising any right under the CC-BY licence, Frontiers must be attributed as the original publisher of the article or ebook, as applicable.

Authors have the responsibility of ensuring that any graphics or other materials which are the property of others may be included in the CC-BY licence, but this should be checked before relying on the CC-BY licence to reproduce those materials. Any copyright notices relating to those materials must be complied with.

Copyright and source acknowledgement notices may not be removed and must be displayed in any copy, derivative work or partial copy which includes the elements in question.

All copyright, and all rights therein, are protected by national and international copyright laws. The above represents a summary only. For further information please read Frontiers' Conditions for Website Use and Copyright Statement, and the applicable CC-BY licence.

ISSN 1664-8714
ISBN 978-2-8325-4905-6
DOI 10.3389/978-2-8325-4905-6

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Deep learning for marine science

Topic editors

Haiyong Zheng — Ocean University of China, China
Mark C. Benfield — Louisiana State University, United States
Hongsheng Bi — University of Maryland, United States
Xuemin Cheng — Tsinghua University, China

Citation

Zheng, H., Benfield, M. C., Bi, H., Cheng, X., eds. (2024). *Deep learning for marine science*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-4905-6

Table of contents

07 Editorial: Deep learning for marine science
Haiyong Zheng, Hongsheng Bi, Xuemin Cheng and Mark C. Benfield

12 An offshore subsurface thermal structure inversion method by coupling ensemble learning and tide model for the South Yellow Sea
Fangjie Yu, Fengzhi Sun, Jianchao Li and Ge Chen

28 A new non-invasive tagging method for leopard coral grouper (*Plectropomus leopardus*) using deep convolutional neural networks with PDE-based image decomposition
Yangfan Wang, Chun Xin, Boyu Zhu, Mengqiu Wang, Tong Wang, Ping Ni, Siqi Song, Mengran Liu, Bo Wang, Zhenmin Bao and Jingjie Hu

41 Lightweight object detection algorithm based on YOLOv5 for unmanned surface vehicles
Jialin Zhang, Jiucai Jin, Yi Ma and Peng Ren

54 ESPC-BCS-Net: A network-based CS method for underwater image compression and reconstruction
Zhenyue Li, Ge Chen and Fangjie Yu

62 TSI-SD: A time-sequence-involved space discretization neural network for passive scalar advection in a two-dimensional unsteady flow
Ning Song, Hao Tian, Jie Nie, Haoran Geng, Jinjin Shi, Yuchen Yuan and Zhiqiang Wei

77 Estimating catch rates in real time: Development of a deep learning based *Nephrops* (*Nephrops norvegicus*) counter for demersal trawl fisheries
Ercan Avsar, Jordan P. Feekings and Ludvig Ahm Krag

92 Scaling whale monitoring using deep learning: A human-in-the-loop solution for analyzing aerial datasets
Justine Boulent, Bertrand Charry, Malcolm McHugh Kennedy, Emily Tissier, Raina Fan, Marianne Marcoux, Cortney A. Watt and Antoine Gagné-Turcotte

105 Spatial-temporal transformer network for multi-year ENSO prediction
Dan Song, Xinqi Su, Wenhui Li, Zhengya Sun, Tongwei Ren, Wen Liu and An-An Liu

120 Few-shot fine-grained fish species classification *via* sandwich attention CovaMNet
Jiping Zhai, Lu Han, Ying Xiao, Mai Yan, Yueyue Wang and Xiaodong Wang

132 A Multi-Mode Convolutional Neural Network to reconstruct satellite-derived chlorophyll-a time series in the global ocean from physical drivers
Joana Roussillon, Ronan Fablet, Thomas Gorgues, Lucas Drumetz, Jean Littaye and Elodie Martinez

152 Underwater target detection algorithm based on improved YOLOv4 with SemiDSConv and FIoU loss function
Chengpengfei Zhang, Guoyin Zhang, Heng Li, Hui Liu, Jie Tan and Xiaojun Xue

165 Estimating precision and accuracy of automated video post-processing: A step towards implementation of AI/ML for optics-based fish sampling
Jack H. Prior, Matthew D. Campbell, Matthew Dawkins, Paul F. Mickle, Robert J. Moorhead, Simegnew Y. Alaba, Chiranjibi Shah, Joseph R. Salisbury, Kevin R. Rademacher, A. Paul Felts and Farron Wallace

181 EchoAI: A deep-learning based model for classification of echinoderms in global oceans
Zhinuo Zhou, Ge-Yi Fu, Yi Fang, Ye Yuan, Hong-Bin Shen, Chun-Sheng Wang, Xue-Wei Xu, Peng Zhou and Xiaoyong Pan

190 Automatic detection and classification of coastal Mediterranean fish from underwater images: Good practices for robust training
Ignacio A. Catalán, Amaya Álvarez-Ellacuría, José-Luis Lisani, Josep Sánchez, Guillermo Vizoso, Antoni Enric Heinrichs-Maquilón, Hilmar Hinz, Josep Alós, Marco Signarioli, Jacopo Aguzzi, Marco Francescangeli and Miquel Palmer

201 Token-Selective Vision Transformer for fine-grained image recognition of marine organisms
Guangzhe Si, Ying Xiao, Bin Wei, Leon Bevan Bullock, Yueyue Wang and Xiaodong Wang

212 See you somewhere in the ocean: few-shot domain adaptive underwater object detection
Lu Han, JiPing Zhai, Zhibin Yu and Bing Zheng

224 Instance segmentation ship detection based on improved Yolov7 using complex background SAR images
Muhammad Yasir, Lili Zhan, Shanwei Liu, Jianhua Wan, Md Sakaouth Hossain, Arife Tugsan Isiacik Colak, Mengge Liu, Qamar Ul Islam, Syed Raza Mehdi and Qian Yang

239 ULL-SLAM: underwater low-light enhancement for the front-end of visual SLAM
Zhichao Xin, Zhe Wang, Zhibin Yu and Bing Zheng

257 Classification of inbound and outbound ships using convolutional neural networks
Doudou Guo, Dazhi Gao, Zhuo Chen, Yuzheng Li, Xiaojing Zhao, Wenhua Song and Xiaolei Li

268 Remote sensing and machine learning method to support sea surface pCO_2 estimation in the Yellow Sea
Wei Li, Chunli Liu, Weidong Zhai, Huizeng Liu and Wenjuan Ma

280 An underwater imaging method of enhancement via multi-scale weighted fusion
Hao Zhang, Longxiang Gong, Xiangchun Li, Fei Liu and Jiawei Yin

292 An iterative labeling method for annotating marine life imagery
Zhiyong Zhang, Pushyami Kaveti, Hanumant Singh, Abigail Powell, Erica Fruh and M. Elizabeth Clarke

304 From shallow sea to deep sea: research progress in underwater image restoration
Wei Song, Yaling Liu, Dongmei Huang, Bing Zhang, Zhihao Shen and Huifang Xu

328 Generalised deep learning model for semi-automated length measurement of fish in stereo-BRUVS
Daniel Marrable, Sawitchaya Tippaya, Kathryn Barker, Euan Harvey, Stacy L. Bierwagen, Mathew Wyatt, Scott Bainbridge and Marcus Stowar

339 Automatic single fish detection with a commercial echosounder using YOLO v5 and its application for echosounder calibration
Jianfeng Tong, Weiqi Wang, Minghua Xue, Zhenhong Zhu, Jun Han and Siqian Tian

352 Demystifying image-based machine learning: a practical guide to automated analysis of field imagery using modern machine learning tools
Byron T. Belcher, Eliana H. Bower, Benjamin Burford, Maria Rosa Celis, Ashkaan K. Fahimipour, Isabela L. Guevara, Kakani Katija, Zulekha Khokhar, Anjana Manjunath, Samuel Nelson, Simone Olivetti, Eric Orenstein, Mohamad H. Saleh, Brayan Vaca, Salma Valladares, Stella A. Hein and Andrew M. Hein

376 Edge computing at sea: high-throughput classification of *in-situ* plankton imagery for adaptive sampling
Moritz S. Schmid, Dominic Daprano, Malhar M. Damle, Christopher M. Sullivan, Su Sponaugle, Charles Cousin, Cedric Guigand and Robert K. Cowen

387 Southwestern Atlantic ocean fronts detected from the fusion of multi-source remote sensing data by a deep learning model
Zhi Wang, Ge Chen, Chunyong Ma and Yalong Liu

396 Simultaneous restoration and super-resolution GAN for underwater image enhancement
Huiqiang Wang, Guoqiang Zhong, Jinxuan Sun, Yang Chen, Yuxiao Zhao, Shu Li and Dong Wang

411 Real-time GAN-based image enhancement for robust underwater monocular SLAM
Ziqiang Zheng, Zhichao Xin, Zhibin Yu and Sai-Kit Yeung

423 Using deep learning to assess temporal changes of suspended particles in the deep sea
Naoki Saito, Travis W. Washburn, Shinichiro Yano and Atsushi Suzuki

436 Subtidal seagrass detector: development of a deep learning seagrass detection and classification model for seagrass presence and density in diverse habitats from underwater photoquadrats
Lucas A. Langlois, Catherine J. Collier and Len J. McKenzie

449 DCC-GAN-based channel emulator for underwater wireless optical communication systems
Huanxin Huo, Min Fu, Xuefeng Liu and Bing Zheng

462 An acoustic tracking model based on deep learning using two hydrophones and its reverberation transfer hypothesis, applied to whale tracking
Kangkang Jin, Jian Xu, Xuefeng Zhang, Can Lu, Luochuan Xu and Yi Liu

478 SymmetricNet: end-to-end mesoscale eddy detection with multi-modal data fusion
Yuxiao Zhao, Zhenlin Fan, Haitao Li, Rui Zhang, Wei Xiang, Shengke Wang and Guoqiang Zhong

493 A meta-deep-learning framework for spatio-temporal underwater SSP inversion
Wei Huang, Deshi Li, Hao Zhang, Tianhe Xu and Feng Yin

515 Toward efficient deep learning system for *in-situ* plankton image recognition
Junbai Yue, Zhenshuai Chen, Yupu Long, Kaichang Cheng, Hongsheng Bi and Xuemin Cheng

529 Hybrid quantum-classical convolutional neural network for phytoplankton classification
Shangshang Shi, Zhimin Wang, Ruimin Shang, Yanan Li, Jiaxin Li, Guoqiang Zhong and Yongjian Gu

541 CLOINet: ocean state reconstructions through remote-sensing, *in-situ* sparse observations and deep learning
Eugenio Cutolo, Ananda Pascual, Simon Ruiz, Nikolaos D. Zarokanellos and Ronan Fablet

frontiers | Frontiers in Marine Science

TYPE Editorial
PUBLISHED 03 May 2024
DOI 10.3389/fmars.2024.1407053

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Hervé Claustre,
Centre National de la Recherche Scientifique (CNRS), France

*CORRESPONDENCE
Haiyong Zheng
zhenghaiyong@ouc.edu.cn

RECEIVED 26 March 2024
ACCEPTED 23 April 2024
PUBLISHED 03 May 2024

CITATION
Zheng H, Bi H, Cheng X and Benfield MC (2024) Editorial: Deep learning for marine science.
Front. Mar. Sci. 11:1407053.
doi: 10.3389/fmars.2024.1407053

COPYRIGHT
© 2024 Zheng, Bi, Cheng and Benfield. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Editorial: Deep learning for marine science

Haiyong Zheng[1]*, Hongsheng Bi[2], Xuemin Cheng[3] and Mark C. Benfield[4]

[1]College of Electronic Engineering, Ocean University of China, Qingdao, China, [2]Center for Environmental Science, University of Maryland, College Park, Cambridge, MA, United States, [3]Shenzhen International Graduate School, Tsinghua University, Shenzhen, China, [4]College of the Coast & Environment, Louisiana State University, Baton Rouge, LA, United States

KEYWORDS

research survey, marine/underwater image enhancement/restoration/compression, marine/underwater visual recognition/detection, dataset and labeling, marine process/phenomenon prediction/detection, marine physical/biogeochemical variable prediction/reconstruction, marine optics/acoustics

Editorial on the Research Topic
Deep learning for marine science

In recent years, Deep Learning (DL) technology has been widely used in marine science and technology research, and provides powerful technical support for related researches and applications. As ocean observation technology continues to advance, the volume of data generated by marine scientific research is steadily increasing. This offers vast potential for data-driven DL to demonstrate its capabilities and has therefore emerged as a valuable technology across multiple research fields, including biology, ecosystems, climate, energy, as well as physical and chemical interactions.

The Research Topic "Deep Learning for Marine Science" aims to provide a research collection to collect relevant research work on the application of DL technology in marine science. A total of 39 papers are published with contributions by 236 authors. The contents in these papers focus on the following aspects: research survey, marine/underwater image enhancement/restoration/compression, marine/underwater visual recognition/detection, dataset and labeling, marine process/phenomenon prediction/detection, marine physical/biogeochemical variable prediction/reconstruction, and marine optics/acoustics. Here, we summarize the contents of these papers and highlight their key contributions to the Research Topic.

1 Research survey

Although machine learning tools hold great promise, they are still not being used to their full potential in several areas, such as species and environmental monitoring, biodiversity surveys, fisheries abundance and size estimation, rare events, and species detection, the study of animal behavior, and citizen science. To help researchers effectively apply image-based machine learning methods in their research problems, Belcher et al. write a review article that provides an easily approachable end-to-end guide.

In terms of underwater image restoration technology, Song et al. make a systematic review to bridge the gap between shallow sea and deep-sea image restoration through experimental analysis. The review mainly describes the core concepts and methods of the three types of shallow sea image restoration methods. It also summarizes the research status and main challenges of deep-sea image restoration, discusses potential solutions, conducts experiments and in-depth discussions, and proposes several development directions for deep-sea image restoration in the future.

2 Marine/underwater image enhancement/ restoration/compression

It is a challenging task to store and transmit high-quality underwater images. To improve the performance of adaptive sampling and reconstruction of underwater images, Li et al. combine the advantages of compressed sensing and DL to propose ESPC-BCS-Net. The method obtains parameters (such as sampling matrix, sparse transforms, and shrinkage thresholds) through end-to-end learning. The experimental results are visually and quantitatively evaluated, demonstrating that the proposed method has good compression and reconstruction effects.

Xin et al. introduce an end-to-end network for Simultaneous Localization And Mapping (SLAM) pre-processing in low-light underwater environments, aiming to address the limitations of visual SLAM systems based on feature point extraction. The proposed network comprises a low-light enhancement branch with a non-reference loss function, a self-supervised feature point detector, and a descriptor extraction branch. Additionally, a unique matrix transformation method is designed to enhance the feature similarity between two adjacent video frames, thereby improving the performance of underwater SLAM.

In order to solve the important problems of blur and color distortion in underwater optical imaging and improve the ability to accurately perceive underwater images, Zhang et al. propose a multi-scale weighted fusion method. By merging, enhancing, and reconstructing images, the clarity and color fidelity of underwater images are effectively improved, and the quality of underwater images presented is improved. Excellent results have been obtained in many experimental indexes.

Zheng et al. propose a solution to improve the performance of underwater monocular visual SLAM systems. The existing SLAM algorithms are often impractical or invalid due to the complex aquatic environment and the poor image quality obtained in such conditions. The proposed solution involves using a Generative Adversarial Network (GAN) to enhance the underwater images before SLAM processing. To reduce the inference cost, the GAN is compressed through knowledge distillation. This approach ensures real-time inference and high-fidelity underwater image enhancement.

To improve the quality of underwater images and achieve simultaneous restoration and super-resolution, Wang et al. propose an end-to-end trainable model named Simultaneous Restoration and Super-Resolution GAN (SRSRGAN). The model uses GANs and consists of two stages of a cascading architecture to restore and super-resolve damaged underwater images coarse to fine. The proposed method is experimentally validated and demonstrates its superiority in underwater image restoration, super-resolution, and simultaneous restoration and super-resolution.

3 Marine/underwater visual recognition/detection

In order to realize the fast navigation of Unmanned Surface Vehicle (USV) in complex marine environments, a target detection algorithm with high detection speed and accuracy is essential. To address this Research Topic, Zhang et al. propose a YOLOv5 lightweight object detection algorithm that leverages the Ghost module and Transformer, resulting in high-efficiency and high-precision object detection. The proposed algorithm is tested on ship videos collected by the “JiuHang 750” USV in different marine environments and demonstrates promising results.

To address the problem of ship instance segmentation in Synthetic Aperture Radar (SAR) images with high resolution and complex backgrounds, Yasir et al. propose a unique YOLOv7 improved high-resolution remote sensing (HR-RS) image segmentation single-stage detection method. The method enhances the accuracy, efficiency, and model robustness of ship instance segmentation through improvements made to the single-stage detector, backbone network, and network feature fusion part, and promising results have been achieved.

To enhance the economic and environmental performance of the fishery, Avsar et al. utilize underwater images captured by an in-trawl video recording system to obtain quantitative information on the capture rate of *Nephrops norvegicus*, a target species. The study employs real-time detection, tracking, and counting techniques to monitor the entry of the target species into the trawl. The detection is done using the YOLOv4 algorithm, which has a proven track record in real-time processing underwater images to determine the target species’ capture rate. Additionally, the algorithm has the potential to process multiple species simultaneously.

Saito et al. utilize DL to investigate the suspended particles in the depths of the sea. To analyze the variability of suspended particle abundance in the images taken by the standard fixed camera “Edokko Mark 1”, they implement object detection technology through the YOLOv5 algorithm to create a suspended particle detection model. They conduct the first excavation test of cobalt-rich ferromanganese crust in the world. The ability of the model to measure changes in the concentration of deep-sea suspended particles is assessed, and the effectiveness of the proposed method in detecting temporal changes of suspended particles and detecting significant abrupt changes, such as mining effects, is validated.

Collecting data on marine fish can be a challenging task due to the nature of their environment, often resulting in poor-quality data. Moreover, identifying various fish categories from small sample images can be difficult, especially regarding fine-grained classification. Zhai et al. propose a new attention network called the

Sandwich Attention Covariance Metric Network (SACovaMNet), which applies metric learning and incorporates attention modules to comprehensively improve the feature extraction capability from global and local perspectives. The result is an excellent performance in the task of fine-grained fish classification.

Prior et al. develop automated video post-processing models to implement automated image analysis of commercially important Gulf of Mexico fish species and habitats. In addition to traditional metrics used to measure the performance of Artificial Intelligence and Machine Learning (AI/ML) models, such as mean Average Precision (mAP), the automated counts are compared to validated set counts to ensure accuracy. The adapting comparative otolith aging methods and metrics are used to measure the model performance, which helps researchers analyze and make management decisions. This approach provides a valuable tool for analyzing Gulf of Mexico fish species and habitats.

Han et al. propose a few-shot domain adaptive underwater object detection framework to address the issues of expensive establishment of marine species database and unstable domain shifting of underwater objects caused by the complex marine environment. The framework includes a novel two-stage training method and a lightweight feature correction module that can adapt to image-level and instance-level domain shifting on multiple datasets. The method quickly demonstrates its knowledge transfer capability in detecting two similar marine species.

Through the sea trial experimental data, Guo et al. propose to automatically identify inbound and outbound ships by utilizing the phenomenon that the sound field interference structures of inbound and outbound ships are different due to the variation of the topography of the shallow continental shelf. The approach utilizes only a single scalar hydrophone to collect data and employs four convolutional neural networks to classify inbound and outbound ships. And this research method can be applied to the intelligent monitoring of ships entering and leaving ports.

To address the challenge of applying DL algorithms to underwater target detection tasks due to the complex underwater environment and low image quality, Zhang et al. propose an underwater target detection algorithm based on an improved version of YOLOv4. This proposed method achieves superior detection performance and efficiency in experiments by incorporating a newly designed convolutional network module, loss function, and detector strategy.

Large-scale research on plankton classification, which uses machine learning techniques, requires powerful computing resources. The exponential computing power of quantum computers makes quantum machine learning a potential solution for large-scale data processing. Therefore, Shi et al. propose a hybrid quantum-classical convolutional neural network (CNN) for the identification task of phytoplankton. The model demonstrates the feasibility of using quantum deep neural networks for phytoplankton classification for the first time. The proposed model exhibits a faster convergence rate, higher classification accuracy, and lower accuracy fluctuation compared to classic CNN-based models.

Commercial fishing vessels face difficulties in collecting acoustic data required for species classification and population evaluation due to the limited calibration capability and frequent data loss of current commercial echo sounders. To address this issue, Tong et al. develop an automatic detection and classification model for Pacific saury (*Cololabis saira*) echo trace using the YOLOv5m algorithm. This model enables the measurement of *in-situ* values of Pacific saury using a single fish echo trace. Furthermore, the living fish calibration method is utilized to facilitate rapid calibration of commercial echo sounders.

To measure the fish without disturbing their natural habitat and overcome the limitation of manual measurement with potentially harmful intervention, Marrable et al. propose a generalized, semi-automatic method that combines the DL method with the high-precision stereo-BRUVS calibration method. The calibration cube is used to ensure that the accuracy of the calculated length is within a few millimeters and that the measurement accuracy is close to the accuracy of human measurements.

In order to distinguish the subtle changes of marine organisms and achieve accurate fine-grained classification, Si et al. propose a new transformer-based framework, token-selective vision transformer, and also propose a token-selective self-attention to select important tokens with discrimination for attention calculation, so as to limit attention to more accurate local areas. Experiments on three marine biological datasets verify that the proposed method can achieve state-of-the-art performance.

Current DL methods face challenges in processing *in-situ* plankton images due to large computation and long consumption time. To address this issue, Yue et al. propose an inter-class similarity distillation algorithm. This method enables the student network (small scale) to acquire excellent plankton recognition ability under the guidance of the teacher network (large scale). The experiment proves helpful in improving the accuracy and speed of plankton recognition, establishing effective DL models, and facilitating the deployment of underwater plankton imaging systems.

To address the ever-changing marine environments and diverse marine life, Schmid et al. implement edge computing technology by integrating the latest *In-situ* Ichthyoplankton Imaging System-3 (ISIIS-3) in the Northern California Current. The edge server utilizes DL techniques to achieve high-throughput *in-situ* plankton classification technology for real-time data adaptive sampling.

In order to develop and evaluate a subtidal seagrass detector method, Langlois et al. adopt a DL model to detect most forms of seagrass appearing in various habitats in the seascape of northeast Australia from underwater images, and classify them according to the coverage degree of seagrass to obtain high accuracy, and better application value and prospects.

To create a non-invasive method to recognize leopard coral grouper (*Plectropomus leopardus*), Wang et al. develop a multiscale image processing method based on matched filters with Gaussian kernels and partial differential equation (PDE) multiscale hierarchical decomposition with the deep convolutional neural network models VGG19 and ResNet50 to extract shape and texture image features of individuals. They then use these features to identify individual *Plectropomus leopardus* in sequence images captured over 50 days. To achieve this, they employ random forest, support vector machine, and multi-layer perceptron methods for individual recognition. The experimental results demonstrate that

the CNN based on PDE decomposition can identify *Plectropomus leopardus* effectively and with great accuracy.

4 Dataset and labeling

Catalán et al. create a new labeling dataset with the aim to further study and improve the application of DL techniques in identifying and classifying fish in underwater images. The dataset consists of more than 18,400 recorded Mediterranean fish from 20 different species, which are obtained through various operations such as different backgrounds, sample size, labeling quality, etc. These fish were extracted from underwater images captured from over 1,600 diverse backgrounds, which will assist in improving the use of DL in studying underwater life.

To achieve efficient data labeling and reduce the cost of manual labeling, Zhang et al. propose a weakly supervised learning framework for labeling marine biological data. This method utilizes crowdsourcing interfaces to converge to a labeled image dataset through multiple training and production loops. Experimental results demonstrate that training with a small subset and iterating over the results can converge to a large, highly annotated dataset with a small number of iterations.

Remote sensing technology can potentially capture aerial images of cetaceans across a vast observation area. However, current limitations in automated analysis techniques require biologists to manually analyze all images, leading to exorbitant tagging costs. Boulent et al. propose a human-in-the-loop approach that merges the proficiency of biologists with DL-based automation capabilities to create a reliable AI-assisted annotation tool for large-scale cetacean monitoring.

DL has been applied to the image classification of marine echinoderms in response to the need for automatic classification in marine biology research worldwide. Zhou et al. collect image data of marine echinoderms and classify them according to systematic taxonomy. Based on the DL model EfficientNetV2, an automatic classification tool (EchoAI) is developed. The EchoAI tool, along with methods and strategies, can classify images of other categories of marine organisms, thus helping researchers investigate the diversity, abundance, and distribution of marine species.

5 Marine process/phenomenon prediction/detection

Song et al. propose a new method called Time-Sequence-Involved Space Discretization neural network (TSI-SD) to solve the problem of large computation amount and high complexity of the fluid numerical model. This method extracts grid correlations from both spatial and temporal views simultaneously and combines TSI-SD with finite volume format as an advection solver for passive scalar advection in a two-dimensional unsteady flow field. Compared to the previous method that only considers spatial context, TSI-SD achieves higher simulation accuracy and reduces the calculation amount. Comprehensive experiments have verified the superior computational efficiency and accuracy of this method.

Song et al. propose a spatio-temporal transformer network that overcomes the defects of existing methods in network structure design and prediction errors to accurately, quickly and effectively predict ENSO events. This network simulates the inherent characteristics of spatio-temporal variations of sea surface temperature anomaly maps and heat content anomaly maps and takes into account the influence of seasonal variations on the prediction of ENSO phenomena. Additionally, an effective recurrent forecasting strategy is proposed, which takes previous predictions as prior knowledge to improve the reliability of long-term forecasting.

Aiming at addressing the problem that the current method only uses single-modal Sea Surface Height (SSH) data to detect mesoscale eddy, which often leads to inaccurate results, Zhao et al. propose an end-to-end mesoscale eddy detection method based on multi-modal data fusion, and add the data of the Sea Surface Temperature (SST) and the velocity of flow. The superior performance of the proposed method is demonstrated on various multi-modal mesoscale eddy datasets.

In view of the problem that the ocean front detection method in the Southwestern Atlantic Front (SAF) mainly adopts the thermal gradient method while ignoring dynamic features, which leads to inaccurate manifestation of SAF. Wang et al. develop a DL model, SAFNet, to detect the SAF through the synergistic effect of satellite SST and SSH observation data in 10 years (2010-2019), to achieve high-precision SAF detection with the fusion of thermal and dynamic features.

6 Marine physical/biogeochemical variable prediction/reconstruction

Based on satellite observations, machine learning has successfully reconstructed the high-resolution ocean subsurface thermohaline structure. However, due to the macro-tidal environment and limited *in-situ* observations, the offshore subsurface parameter estimation accuracy will be affected. Yu et al. propose a new approach by coupling the TPXO tidal model and light gradient boosting machine algorithm to develop an inversion model of offshore subsurface thermal structure for the South Yellow Sea (SYS) using sea surface data and *in-situ* observations. The experimental results show that the reconstruction is reliable in the SYS area, and the proposed method also provides a new exploration direction for reconstructing offshore ocean thermal structures.

For the reconstruction of satellite-derived chlorophyll-a concentration in a global scale, Roussillon et al. propose a method based on physical predictors, and uses a multi-mode convolutional neural network to globally account for interregional variabilities via learning and combining different modes spatially. The different modes show regional consistency with ocean dynamics, and the work contributes to new insights into the physical-biogeochemical processes that control temporal and spatial variability in phytoplankton on a global scale.

The current status of the sea surface carbon dioxide partial pressure (pCO2) in the Yellow Sea is unclear due to limited availability of *in-situ* spatial and temporal distribution data. To

address this problem, Li et al. develop a pCO2 model using a random forest algorithm. The model uses 14 cruise datasets from 2011 to 2019, as well as input variables such as remote sensing satellite sea surface temperature, chlorophyll concentration, diffuse attenuation of downwelling irradiance, and *in-situ* salinity. The model is trained and tested, yielding excellent prediction and evaluation results.

Cutolo et al. develop a CLuster Optimal Interpolation Neural Network (CLOINet) to combine remote-sensing data with *in-situ* observation and create a comprehensive 3D reconstruction of the ocean state. CLOINet combines the robust mathematical framework of the optimal interpolation scheme with a self-supervised clustering method and also effectively segments remote sensing images into clusters to reveal non-local correlations and enhance fine-scale ocean reconstruction. The network is trained using the output of the Ocean General Circulation Model and shows good reconstruction results in various testing scenarios.

7 Marine optics/acoustics

Huang et al. propose a Task-driven Meta-Deep-Learning (TDML) framework to solve the problem that the nonuniform distribution of sound speed will bring difficulties to underwater accurate positioning. It learns the common features of the Sound Speed Profile (SSP) through multiple base learners, accelerates the model convergence on new tasks, and enhances the model's sensitivity to changes in sound field data through metatraining. Thus, the over-fitting effect is weakened, and the inversion accuracy is improved. Experimental results show that the proposed TDML method can achieve fast and accurate spatio-temporal SSP inversion.

To fully consider how water environment and communication equipment affect signal transmission and accurately simulate the complex characteristics of the Underwater Wireless Optical Communication (UWOC) systems, Huo et al. develop a UWOC channel emulator based on deep convolutional conditional generative adversarial networks, which are tested in experiments to verify their excellent performance in the time domain, frequency domain, and universality under different water turbidity levels.

To achieve full acoustic tracking of whales with reverberation interference, Jin et al. propose an intelligent acoustic tracking model that enables horizontal direction discrimination and distance/depth perception by mining unpredictable features of position information directly from signals received from two hydrophones. The proposed method not only achieves satisfactory prediction performance, but also effectively avoids the reverberation effect of signal propagation over long distances.

Author contributions

HZ: Writing – original draft. HB: Writing – review & editing. XC: Writing – review & editing. MB: Writing – review & editing.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 19 December 2022
DOI 10.3389/fmars.2022.1075938

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, United States

REVIEWED BY
Young-Heon Jo,
Pusan National University,
Republic of Korea
Shengqiang Wang,
Nanjing University of Information Science and Technology, China

*CORRESPONDENCE
Ge Chen
gechen@ouc.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 21 October 2022
ACCEPTED 06 December 2022
PUBLISHED 19 December 2022

CITATION
Yu F, Sun F, Li J and Chen G (2022) An offshore subsurface thermal structure inversion method by coupling ensemble learning and tide model for the South Yellow Sea.
Front. Mar. Sci. 9:1075938.
doi: 10.3389/fmars.2022.1075938

COPYRIGHT
© 2022 Yu, Sun, Li and Chen. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

An offshore subsurface thermal structure inversion method by coupling ensemble learning and tide model for the South Yellow Sea

Fangjie Yu[1,2], Fengzhi Sun[1], Jianchao Li[3] and Ge Chen[1,2]*

[1]College of Information Science and Engineering, Ocean University of China, Qingdao, China, [2]Laboratory for Regional Oceanography and Numerical Modeling, Qingdao National Laboratory for Marine Science and Technology, Qingdao, China, [3]Key Laboratory of Mariculture, Ministry of Education, Ocean University of China, Qingdao, China

The South Yellow Sea Cold Water Mass (SYSCWM), which occurs in the South Yellow Sea (SYS) during summer, significantly impacts the hydrological characteristics and marine ecosystems but lacks fine interior data. With satellite observations, significant achievements have been made in reconstructing high-resolution ocean subsurface thermohaline structure based on machine learning. However, the accuracy of offshore subsurface parameter estimation will be affected due to the macro-tidal environment and fewer *in situ* observations. In this paper, we coupled the TPXO tide model and Light Gradient Boosting Machine algorithm to develop an inversion model of offshore subsurface thermal structure for the SYS using sea surface data and *in situ* observations. After light modelling, the subsurface temperature structure in the SYS is retrieved from sea surface parameters with a spatial resolution of 0.25° at depths of 0-55 m. Observation-based dataset (ARMOR3D) and *in situ* observations are used for model evaluation. According to the validation of the mooring buoy observations, the overall coefficient of determination (R^2), which determines the percentage of variance in the dependent variable that can be explained by the independent variable, is more than 0.95. Furthermore, the R^2 is improved by 12% due to coupling tide model below the thermocline during the maturity stage of SYSCWM, which is helpful for a better reconstruction of SYSCWM. Comparing with the cruise data, the average R^2 of the proposed model is 0.927 which is slightly better than the accuracy of the observation-based ARMOR3D dataset. Since the R^2 exceeds 0.8 in the most area of 121° E~123.5°E, 33°N~36°N, the reconstruction is reliable in this area. The method provides a new explorable direction for reconstructing the ocean thermal structure in offshore areas.

KEYWORDS

offshore thermal structure, tide model data, lightGBM, satellite observations, the South Yellow Sea

1 Introduction

The South Yellow Sea (SYS) is a shallow (average depth of 46 m), semi-enclosed marginal sea in the northwestern Pacific between the Chinese mainland and the Korean Peninsula. Due to the vast and shallow continental shelf, seasonally atmospheric conditions, such as the Asian monsoon, significantly impact the thermal structure of SYS (Chu et al., 1997; Sun et al., 2022). In the winter, strong northwest winds drive the water column to be well-mixed until spring. Weak southeasterly winds prevail in summer, so enhanced solar radiation causes the rapid formation of a strong and stable seasonal thermocline, preventing vertical mixing between the upper mixed layer and deep layer so that the cold water from the previous winter is reserved below the thermocline (Lee et al., 2016). It is called the South Yellow Sea Cold Water Mass (SYSCWM; Li et al., 2017a) in the SYS, which occupies the bottom layers of the central part with a large temperature difference between the surface and the bottom. The SYSCWM plays an important role in the field of hydrodynamics and biochemistry (Wang et al., 2014; Liu et al., 2015; Xin et al., 2015; Li et al., 2016; Guo et al., 2021; Li et al., 2021). The Yellow Sea Warm Current in winter is another prominent feature in the SYS, which transports warm saline water from the Tsushima Warm Current to the SYS (Zhang et al., 2008; Diao et al., 2022; Yu et al., 2022). In addition, SYS is a macro-tidal environment with a huge tidal range and strong tidal currents (Lü et al., 2010; Hwang et al., 2014). These features lead to the water mass of the SYS having high variability. As yet, the knowledge of the SYS has primarily depended on *in situ* observations (Yang et al., 2019). Despite many subsurface *in situ* measurements in the SYS, continuous and fine observations remain sparse. Satellite observations provide multiple data at different spatiotemporal scales but are limited to the surface layer (Ali et al., 2004). To better comprehend the dynamical processes, it is necessary to have continuous and high spatiotemporal resolution subsurface data in the SYS.

Compared to the temperature profiles, the vertical variation of the salinity profiles is slight (less than 2 PSU; Li et al., 2017b). Hence, extensive studies have been conducted to reconstruct the temperature field by dynamical methods in the SYS, which have the advantage of being physically consistent. Lü et al. (2010) reproduced the three-dimensional temperature field and dominant tidal system in the Yellow Sea (YS) based on a wave-tide-circulation coupled numerical model. Zhu et al. (2018) used Princeton Ocean Model to simulate the process of the Yellow Sea Cold Water Mass (YSCWM) and added tidal forcing and freshwater input. Yang et al. (2019) reconstructed the cooling process of sea surface temperature (SST) with a high spatiotemporal resolution during the typhoon passage over the YS by a one-dimensional mixed-layer model. Wan et al. (2022) rebuilt temperature structure and circulation of the YS in winters based on a high-resolution Regional Ocean Modeling System. Relative to the above, the numerical model has well reconstructed ocean temperature structure. Nonetheless, the typical dynamical methods, including numerical simulation and data assimilation, are complex and computationally time-consuming.

Many ocean internal processes have manifestations at surface, so it is possible to retrieve ocean interior parameters from satellite observations for the dynamical connections (Meng et al., 2022). Meantime, machine learning methods are flexible and popular for the ability to extract nonlinear relationships. Therefore, diverse machine learning methods have been applied to estimate ocean interior information in recent years. The self-organizing mapping neural network and support vector machine methods were used to reconstruct the subsurface temperature anomaly (STA) from multisource satellite observations in the Atlantic Ocean and the Indian Ocean (Wu et al., 2012; Su et al., 2015). Meantime, the importance of sea surface salinity (SSS) and sea surface wind (SSW) was revealed by the fact that they can improve the inversion accuracy. Lu et al. (2019) found that the clustering method helps to obtain a better estimated thermal structure. To tackle the challenge of estimating ocean subsurface temperature (OST) in regions with huge seasonal changes, establishing seasonal models is an effective method that could reduce the error of estimated OST, especially in the upper ocean (Su et al., 2021). It may therefore be more efficient that clustering the temperature profiles by seasonal feature. However, it will lead to a sharp reduction of training samples, so the ensemble learning methods were used to predict the OST because they are more appropriate for small sample training than deep learning and classic machine learning approaches (Su et al., 2019; Su et al., 2021). The aforementioned results demonstrate that machine learning algorithms can successfully rebuild the large-scale ocean temperature structure. However, the accuracy will be affected when estimating the thermal structure of the offshore areas using classic machine learning algorithms for the complex tidal environment and fewer data. Therefore, it is worth exploring but challenging to improve the accuracy of estimating offshore subsurface temperature by considering tides and ensemble learning algorithms.

In this study, we propose a framework that couples a tide model with the Light Gradient Boosting Machine algorithm, which is less computational and more appropriate for small samples, to retrieve the subsurface temperature (ST) of the SYS by combining sparse *in situ* measurements with multiple satellite observations. The rest of the paper is organized as follows: Section 2 introduces the datasets and tide model. The methods to retrieve the ST are described in Section 3. In Section 4, we evaluate the reconstruction method and discuss the importance of tides in the model. Finally, a brief conclusion and some prospects are presented in Section 5.

2 Data

2.1 *In situ* data

As the labeled data, three measurements are used in this study: the mooring system, high-resolution profiler, and shipboard survey cruises. A time series of temperature profiles over 9 months (from 22 July 2019 to 15 May 2020), recorded by a mooring system (named M1) which deployed in the SYS, near the western boundary of SYSCWM (35.18°N,122.26°E, Figures 1A, B). The M1 data has 244 temperature profiles after quality control, including 17 depth levels (from 1 m to 55 m), covering the maturation to disappearance of the SYSCWM. The moored high-resolution profiler (named H1), which was deployed at the same location as M1 from 3 June 2022 to 4 July 2022, provides a fine temperature profiles time series. This profiler recorded vertical temperature profiles from 1 m to 50 m during the growth to maturity of the SYSCWM. The sample interval of H1 is 30 min and the vertical resolution is 0.1 m. In this study, the spatiotemporal resolution of the H1 data is averaged to daily and 1 m. In addition, the 55 m depth level of H1 data is extrapolated from several adjacent temperatures for their similarity. Cruise observations were carried out with 1 m vertical resolution in the western SYS in April, July and October 2019. The cruise covered the sea west of 124°E, from 33°N to 37°N, and a total of 5 latitude sections were used in this study. The five temperature latitude sections obtained by CTD castings during the cruise survey along different latitudes (33°N, 34°N, 35°N, 36°N, 37°N), named S33-S37 (Figure 1B).

2.2 Satellite data

Multisource satellite observations are used as input data, including absolute dynamical topography (ADT), SST, SSS, and SSW. The SSW contains u and v components (USSW, VSSW). The ADT data are provided by SSALTO/Data Unification and Altimeter Combination System (DUACS) and were available through the Copernicus Marine Environment Monitoring Service (CMEMS, https://marine.copernicus.eu/). The product merged multiple L3 along-track measurements and conducted the tidal corrections (Taburet et al., 2019). The SST data are obtained from Daily Optimum Interpolation Sea Surface Temperature (DOISST, https://psl.noaa.gov/), developed by National Oceanic and Atmospheric Administration Physical Sciences Laboratory (NOAA PSL). It is a blend of *in situ* SST with satellite SST derived from the Advanced Very High Resolution Radiometer (Banzon et al., 2016; Huang et al., 2021). The SSS data are obtained from SMOS L3OS 2Q Debiased daily valid ocean salinity values product (https://sextant.ifremer.fr/), which are distributed by Centre Aval de Traitement des Données SMOS (CATDS) and corrected the offshore SSS through various *in situ* observations (Boutin et al., 2018). The SSW data are provided by the Cross-Calibrated Multi-Platform (CCMP; https://rda.ucar.edu/datasets/ds745.1/). The CCMP uses a variational analysis method to smoothly fuse multisource surface wind data into the gridded data at 6 hours intervals (Atlas et al., 2011). The temporal resolution of the CCMP data is 6 hourly while the rest is daily, and the spatial resolution of all these data is 0.25°×0.25°.

2.3 Tide model data

We coupled the tide model data into the inputs of machine learning model. The tide model data, including surface tidal elevation and tidal currents, are estimated by the TPXO7 global tidal model provided by Oregon State University, which was built hourly on a 0.25°×0.25° grid. The tide model is based on the hydrodynamic equation and uses the generalized inversion

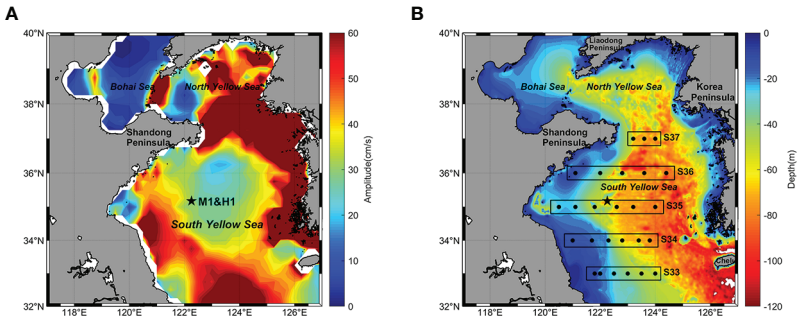


FIGURE 1
M_2 tidal current amplitude and topography of the South Yellow Sea (SYS) and the location of different *in situ* observations. M1 and H1 with the same site, indicated by the black star. **(A)** The amplitude of M_2 tidal current from TPXO7 global tidal model in which the tidal currents are stronger. **(B)** The topography and geography of the SYS. The color contours denote bathymetry. The black dots in the rectangles show the CTD casts along five latitudinal sections (S33-S37) in the cruise survey.

method to assimilate the measured data, including satellite altimetry data and tide observations. Furthermore, it was recently used for the hydrographic study in the YS (Bi et al., 2021; Lin et al., 2021; Sun et al., 2022). The M_2 tide is the most dominant tidal component in the SYS, having stronger tidal current (Figure 1A). The tides have complex structures in the SYS, which is detrimental to temperature inversion. In this study, the tidal time series of eight basic tidal components (M_2, S_2, N_2, K_2, K_1, O_1, P_1, and M_4) are extracted by the Matlab Tide Model Driver toolbox (https://www.esr.org/research/polar-tide-models/tmd-software/). The tide model data and satellite observations, which have the same spatial resolution, were co-located with the temperature profiles by the nearest neighbour method, and the temporal resolution is unified to daily.

2.4 ARMOR3D dataset

We also validate the temperature estimation with the ARMOR3D dataset (Guinehut et al., 2012), which was obtained through CMEMS. The ARMOR3D used multiple linear regression and optimal interpolation, providing the weekly temperature and salt fields at 0.25° × 0.25° resolution over 15 regularly spaced vertical levels between surface and 80 m depth. The weekly averaged three-dimensional temperature field in April, July and October 2019 from ARMOR3D is used to compare. The YSCWM below the thermocline is clearly visible in the observation-based ARMOR3D data (Figure 2A). In addition, the M1 temperature data are used to evaluate ARMOR3D. In order to match the temporal resolution, the M1 data are first calculated as weekly average and then compared to the nearest neighboring grid in ARMOR3D. As shown in Figure 2B, most of the data points are distributed along the equal line with low bias, absolute error and high Pearson's correlation coefficient. The evident seasonal temperature variations in ARMOR3D are well simulated compared to the M1 observations (Figures 2C, D). Even though ARMOR3D presents a shallower mixed layer and a more durable YSCWM which lasts until October, it well reproduces the vertical thermal structure at the M1 station and is worth to refer for the thermal structure of SYS.

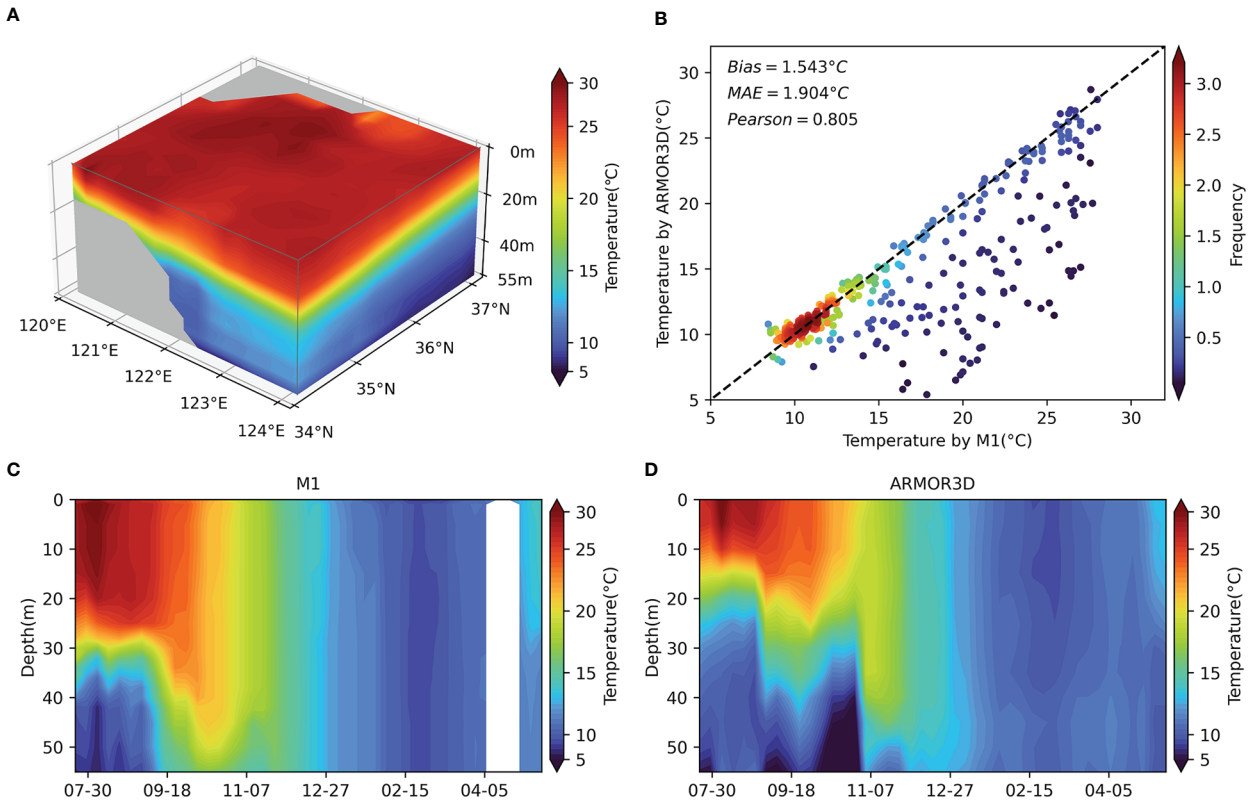


FIGURE 2
YSCWM phenomena in ARMOR3D temperature data and comparison of ARMOR3D, M1 temperature field at M1 location during July 2019 to May 2020. **(A)** The distribution of weekly average surface and subsurface temperature (°C) in the SYS with a spatial resolution of 0.25°×0.25° between 25 July and 31 July 2019 from the ARMOR3D data, selecting 0-55 m depth to correspond to the M1 data. The YSCWM is below the thermocline. **(B)** Scatter plots for M1 temperature and ARMOR3D temperature from all depth. **(C)** Weekly average temperature data from M1 with gaps representing interruptions in the measurements. **(D)** ARMOR3D temperature fields at M1 site.

3 Methods

3.1 Gaussian mixture model clustering

Considering the large seasonal variation of the thermal structure in the SYS, we use unsupervised GMM clustering techniques to shrink the sample space and improve the accuracy (Landschutzer et al., 2013; Parard et al., 2015). As a probabilistic model, GMM is often used for data clustering (Attal et al., 2015). First, the GMM randomly initializes the Gaussian distribution parameters of each cluster. Then the posterior probability of each sample is calculated and used to compute the new Gaussian distribution parameters. The process is repeated until the expectation function is maximized. Compared with the K-means method, GMM is more suitable for non-spherical clusters with different sizes and densities (Wang et al., 2019; Askari, 2021). Therefore, it is appropriate for the classification of ocean temperature profiles (Maze et al., 2017; Sambe and Suga, 2022). GMM requires the number of classes (K) as an input parameter. Therefore, the Davies-Bouldin index (DBI) is used to determine the appropriate number of classes in this study. The number of classes having the minimized DBI is considered the optimal result. Since the initial values of the Expectation-Maximization algorithm are randomized, the GMM clustering was applied 20 times, and 80% of the data were randomly selected from the M1 and H1 data each time to stabilize the clustering results. Figure 3 shows the DBI from clustering results with different K. As a result, we judge that stable and good clustering results could be obtained if K = 3. The clustering results are shown in Figure 4. Although the YSCWM temperature structure from H1 data is still growing, it is approaching maturity. Therefore, they are named after a specific stage of YSCWM: the maturity stage, the declining stage, and the disappearance stage. During the maturity stage of YSCWM with weaker wind, the sea surface is subjected to strong thermal radiation, forming a stable upper mixed layer and a strong thermocline, which prevents heat transfer, so the bottom water stays cold (Lee et al., 2016). It leads to a multi-layer temperature structure in the SYS, with a large temperature difference between the sea surface and the bottom (Figure 4A). In the YSCWM declining stage, the cooling at the sea surface and stronger mixing lead to a thicker and colder upper mixed layer and the subsequent weakening and deepening of the thermocline (Figure 4B). Meanwhile, critical tidal currents raise the temperature at the bottom layer then decline the YSCWM (Li et al., 2016). Thermal forcing at the air-ocean interface and agitation by strong winds together cause strong vertical mixing, forming a well-mixed low temperature structure (Figure 4C) from the sea surface to the bottom in the YSCWM disappearance stage (Chu et al., 1997).

3.2 Light gradient boosting machine

To tackle the limitations of small data and complex computations, we adopt the LGBM algorithm to predict the temperature by taking advantage of its lightweight. LGBM is a gradient boosting framework based on decision trees, which has been well used in the marine field and shown a faster training speed and higher accuracy for small data (Su et al., 2021; Dong et al., 2022). Same as the other boosting algorithms, it sums the results of multiple decision trees as the final prediction output. Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) are two important features of LGBM. The GOSS excludes most of the samples with small gradients and calculates the precise information gain by the remaining

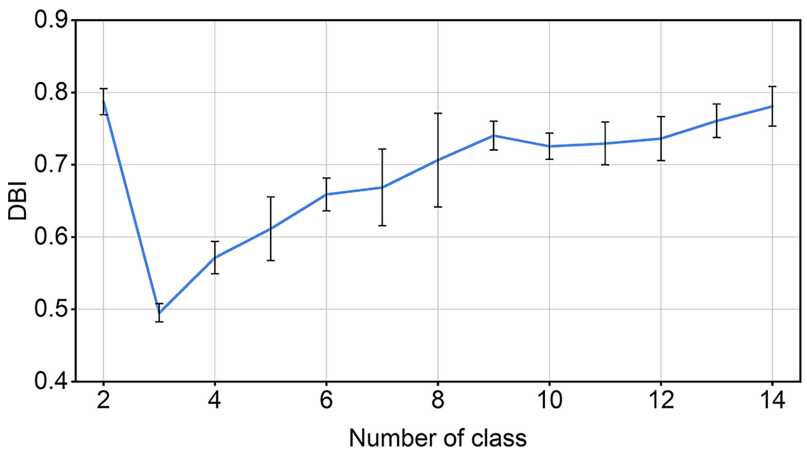


FIGURE 3
The mean value (the blue line) and confidence intervals (one σ, the black error bar) of the Davies-Bouldin index (DBI) from 20 trials of Gaussian mixture model (GMM) clustering for the different number of classes.

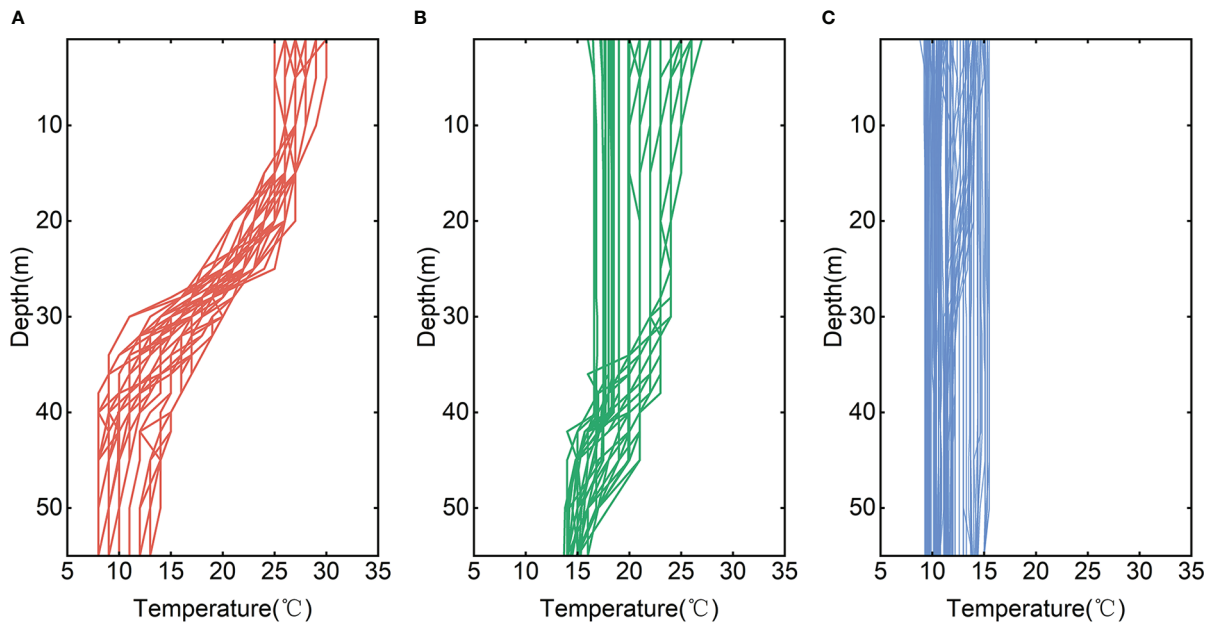


FIGURE 4
Vertical temperature structure of the classified profiles from M1 data, which represents different stages of the YSCWM: **(A)** the maturity stage, **(B)** the declining stage, and **(C)** the disappearance stage.

samples. The EFB approach integrates many mutually exclusive features and reduces the data dimension. To build a better model, the Bayesian optimization strategy is used to optimize several important parameters of LGBM. The optimization method is a Gaussian process with a faster speed. According to previous studies, three essential hyperparameters need to be adjusted: the number of leaf nodes (num_leaves), the learning rate, and the number of iterations (n_estimators). The bounds of n_estimators were set 100 and 1000, and the best n_estimators is 400 without overfitting. It improves the accuracy by 16.6% compared to n_estimators=100. However, the accuracy at n_estimators=1000 is only increased by 0.1% compared to the best n_estimators. When the learning_rate is increased to 0.01 from 0.001, the performance is improved by 21% compared to the starting learning_rate=0.001, but the effect does not enhance when it is increased further until 0.1. The test range of num_leveas is from 5 to 30. The performance of the model at the best num_leveas=5 improved by 3.4% over num_leveas=30. The optimal parameters are shown in Table 1. In addition, the max depth is set to 5, to prevent overfitting due to excessive complexity of the model. The other parameters are set to default values.

3.3 Experimental setup

First, we input the eight harmonic components (M_2, S_2, N_2, K_2, K_1, O_1, P_1, and Q_1), geographic location and time parameters

TABLE 1 Design of experiments and parameter values.

Case	Coupling tide model or not	Clustering or not	Training Models	Parameter values
GLGBM-tides	Yes	Yes	ST = LGBM (SST, ADT, SSS, SSW, tides)	n_estimators = 400, learning_rate = 0.01, max_depth = 5, num_leaves=5
GLGBM	No	Yes	ST = LGBM (SST, ADT, SSS, SSW)	n_estimators = 400, learning_rate = 0.01, max_depth = 5, num_leaves=5
SVR	No	No	ST = SVR (SST, ADT, SSS, SSW)	C = 2.5, gamma = 1.2, kernel = rbf
ANN	No	No	ST = ANN (SST, ADT, SSS, SSW)	Number of neural network layers = 2, number of neurons per layer = 40, learning_rate = 0.01, loss function = MSE

The SSW contains its two components (USSW and VSSW) and the tides include tidal elevation and tidal currents.

into the TPXO7.2 global tidal model, to extract tidal elevation and tidal currents data. TPXO7.2 fits best the Laplace tidal equation in the least squares sense. Second, the datasets consisting of tide model data, satellite observations and *in situ* temperature profiles are divided into three different stages by GMM clustering. The surface parameters (ADT, SST, SSS, SSW, tidal elevation, and tidal currents) are used as independent input variables and the temperature time series are used as labels to prepare the training and test data. To ensure that the training and test sets have a similar seasonal distribution, all samples at the location of M1 are normalized and randomly sampled into the training set (60%) and the test set (40%) by month (Figure 5). Third, the model is tuned and trained using the Bayesian optimization method to obtain suitable temperature estimators at 17 depth levels. Figure 6 shows the technique flowchart of one stage at a certain depth. We use a total of 162 samples to train and 114 samples to test when using mooring observations for validation. Finally, temperature predictions are applied to a larger horizontal space and verified with cruise observations in the SYS where the number of training data and evaluating data are 276 and 78, respectively.

To evaluate the tide model coupled temperature inversion method, we designed comparative trials named GLGBM-tides and GLGBM. They both use the LGBM method with pre-clustering process but the former couples the tide model while the latter does not. Additionally, we compared other reconstruction methods. Case SVR and Case ANN use Support Vector Regression (SVR) model and Artificial Neural Network (ANN) model, respectively. Table 1 summarizes the different trials. These are optimized by the Bayesian optimization strategy, and the parameters of different models are shown in Table 1. The ARMOR3D dataset is also used for comparison.

4 Results and discussion

The sea surface data of the test samples are input into the different models to obtain the reconstructed vertical temperature structure. Based on the test data, we first examine the importance of tides in offshore temperature prediction from the time series data. Then the performance of the different models is compared. Finally, we estimate the temperature structure of each latitude section (S33-S37) and compared it with the ARMOR3D dataset.

4.1 The performance of tide model data on the temperature field reconstruction

Previous studies have shown that strong tidal mixing has an important effect on the temperature structure and enhances vertical heat exchange in the water column during summer in the YS (Lü et al., 2010; Yao et al., 2012; Li et al., 2016; Yu et al., 2016). Here, we first compared GLGBM-tides and GLGBM to investigate how tides affect temperature estimation in this study. Figure 7 shows the comparison between the temperature profiles obtained by the two models and *in situ* observations. The profiles are randomly selected according to spring tide and neap tide in the maturity stage of YSCWM. In this stage, bottom vertical disturbances are stronger (Li et al., 2016), which affects the heat transfer and thermal structure significantly. Besides, the air-sea heat flux and the cooling process of the previous winter strongly influences the intensity of YSCWM (Zhu et al., 2018). This leads to machine learning models having more difficulty accessing these temperature variations and more considerable differences between *in situ* and estimated temperature (Figure 7). However, it can be seen that the temperature profiles obtained from

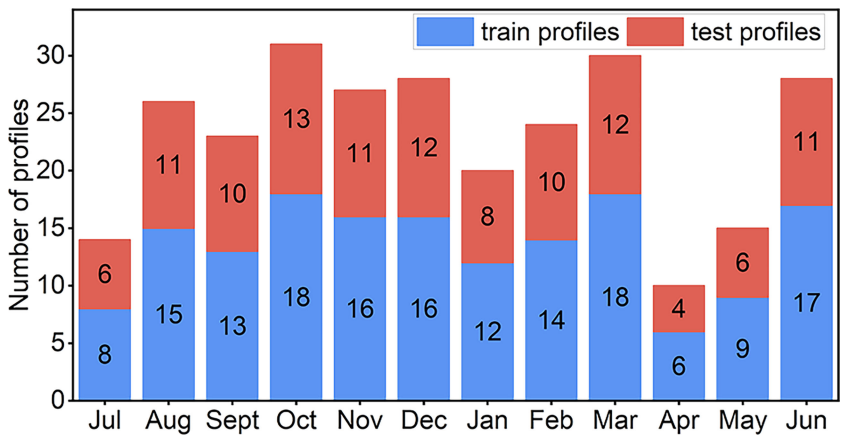


FIGURE 5
Monthly distribution of the number of temperature profiles from M1 and H1 data.

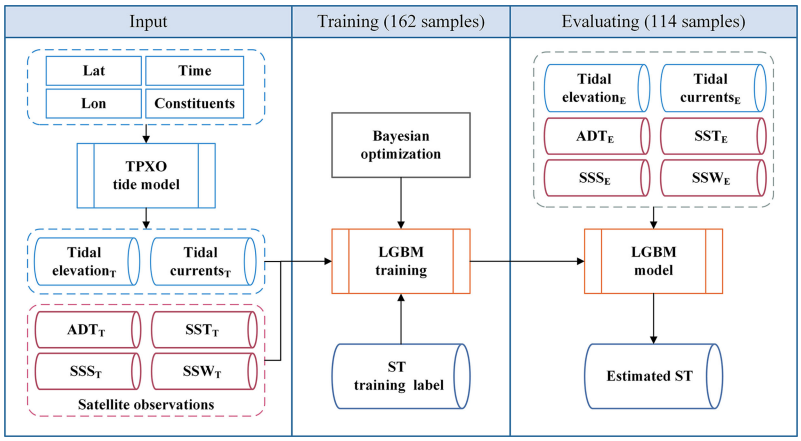


FIGURE 6
Flowchart of the subsurface temperature (ST) estimation at different depth levels using LGBM models for a certain class. In the moored buoy observation validation, a total of 162 samples were used for training and 114 samples for testing. In the cruise survey validation, the training data and validation data are 276 and 78, respectively.

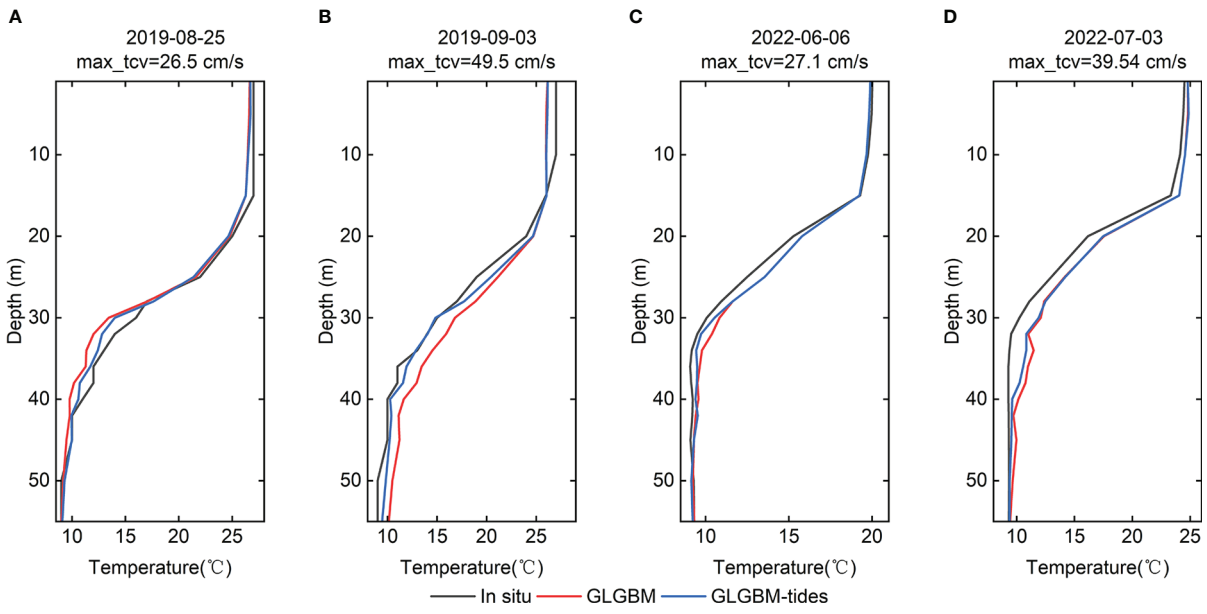


FIGURE 7
Comparison among the vertical structure of temperature at depths of 1-55 m obtained by observed ST (black), GLGBM-tides (blue) and GLGBM (red) during maturity stage of the YSCWM. The profiles are randomly selected according to spring tide **(B, D)** and neap tide **(A, C)**. The max_tcv represents the daily maximum tidal current speed.

GLGBM-tides are more consistent with the measured profiles, especially deeper than 30 m. This confirms that the method coupled with tide model can effectively improve the structure of the predicted temperature profiles during the maturity stage. To further validate the above results, several evaluation indicators metrics are used to assess the two models. Except for root mean square error (RMSE), coefficient of determination (R^2) and absolute difference, the error (defined as the proportion of RMSE in the actual mean temperature observations) is also used to evaluate the accuracy and reliability of the model. The

evaluation indicators are computed as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_i^{'}\right)^2} \quad (1)$$

$$R^2 = \frac{\sum_i^N \left(T_i^{'} - \bar{T}\right)^2}{\sum_i^N \left(T_i - \bar{T}\right)^2} \quad (2)$$

$$Error = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(T_i - T_i^{'}\right)^2}}{\bar{T}} \quad (3)$$

Here, T_i denotes the observed temperature while $T_i^{'}$ is the estimated temperature by models. The $\bar{T}$ is the mean values of T_i over the whole observation. N is the number of test samples.

From Figures 8A-C, the evaluation indicators of the two methods are similar within the 1-28 m depth layer. However, in the 40 m depth level, the RMSEs of the two are 0.806 and 0.863, respectively. Meanwhile, the accuracy of other layers has been improved by different degrees from 30 m to 55 m. Figure 8D shows that the smaller absolute errors occupy a larger proportion in the GLGBM-tides model. In addition, the enhancement is mainly manifested during the maturity stage (Figure 9). It may be attributed to the tidal mixing primarily influencing the range up to 30 m from the bottom during summer (Qiao et al., 2004b). In this trial, GLGBM-tides coupled the tide model while GLGBM not. Meanwhile, strong tides affect the heat transfer and thermal structure of the profile, especially the bottom layer. As a result, GLGBM-tides better learn the temperature variation affected by tidal mixing, and it presents a more consistent vertical thermal structure with *in situ* observations (Figure 7) and better performance than GLGBM (Figure 8).

Furthermore, we analyzed the accuracy of the models at three specific stages from 30 m to 50 m (Figure 9). The averaged R^2 and RMSE are significantly different in the maturity stage of YSCWM and similar in the decline and disappearance stages. It performs less well in the maturity stage than the other two stages in the YSCWM deep. Strong stratification leads to a large difference in temperature between YSCWM and the upper layer. Besides, YSCWM is influenced not only by the air-sea heat flux but also by the cooling process of the previous winter (Zhu et al., 2018). It means that the thermal structure of YSCWM is more difficult to be described by sea surface parameters in the machine learning models hence lower R^2 and higher RMSE. The averaged R^2 of GLGBM-tides and GLGBM are 0.614/0.547, with approximately 12% improvement. It results from the stronger influence of tidal mixing on the temperature structure in summer. Therefore, tides are worth considering in the offshore temperature field reconstruction.

Overall, the GLGBM-tides has good accuracy with errors of less than 8% at all depth layers and most absolute difference of less than 2°C (Figures 8C, D). It is worth noting that a bump appears above 30 m in Figure 8B. This phenomenon may be related to the depth of the mixed layer. According to previous research, the depth of the mixed layer in SYSCWM is about 5-25 m (Qiao et al., 2004b). The temperature does not vary significantly within the mixed layer, which causes the lower RMSE and higher R^2. The tidal mixing primarily influences the

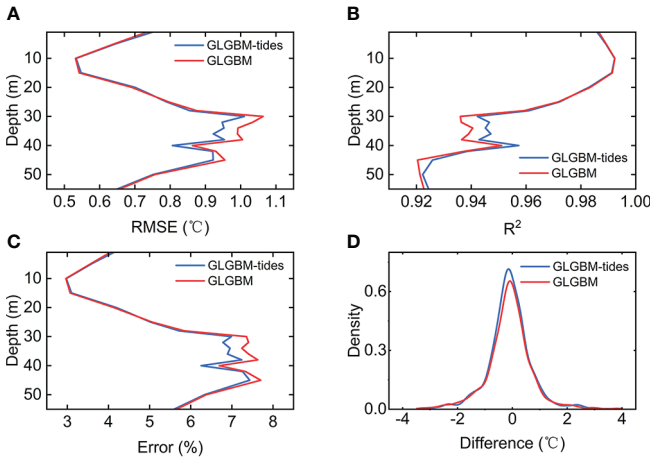


FIGURE 8
The average RMSE **(A)**, R^2 **(B)**, Error **(C)** at the 17 depth levels and absolute difference density distribution **(D)** between the test datasets and estimated ST from GLGBM-tides (blue) and GLGBM (red).

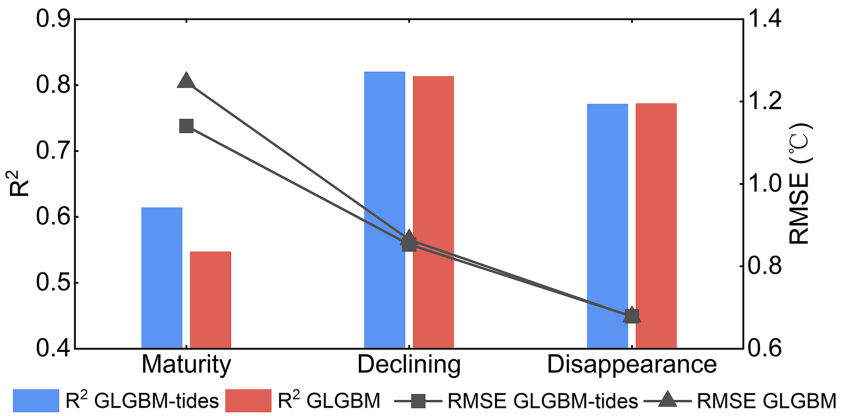


FIGURE 9
The average RMSE and R^2 between 30 and 50 m depth using GLGBM-tides and GLGBM in three YSCWM stages (the lines indicate the RMSE and the bars indicate the R^2).

range up to 30 m from the bottom, enhancing the vertical temperature variability (Qiao et al., 2004a) and the particular structure of the YSCWM makes it difficult for the model to accurately describe the temperature variations. Therefore, the accuracy of reconstruction at these depths will be worse (Figures 8A-C).

It helps to understand the different effects of each sea surface parameter on the ST, by analyzing the importance of sea surface parameters at different depths. The LGBM reflects the importance of different features by calculating the number of times the sea surface parameters are used to segment the data across all trees. The relative importance of each parameter is calculated by summing and normalizing the feature importance from the LGBM. Figure 10A shows the relative importance of each sea surface parameter from GLGBM-tides. According to previous studies, the vertical thermal structure in the Yellow Sea (YS) is influenced by air-sea heat flux, the wind, tidal vertical mixing, and freshwater input (Chu et al., 1997). The temperature in the mixed layer is vertically quasi-uniform due to the mixing of multiple dynamic processes, such as wave motion and wind. Meanwhile, the mixed layer gradually thickens from the maturity stage to the disappearance stage of YSCWM, which means that the sea surface temperature (SST) can explain more subsurface temperature variations. Consequently, SST is the main driver of the model, with a more than 30% contribution at 17 depth levels (Figure 10A). However, below the mixed layer, the heat transfer

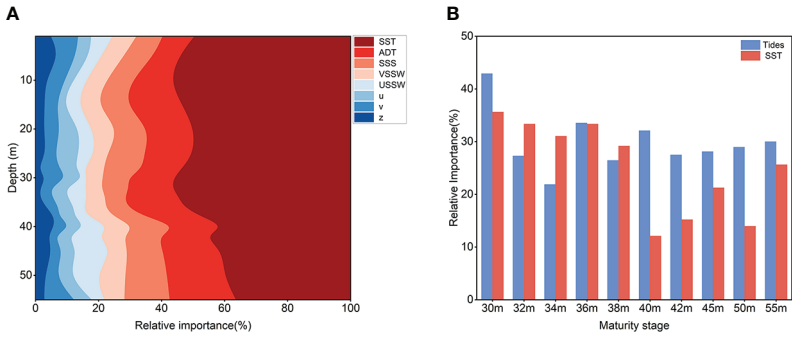


FIGURE 10
The relative importance of each sea surface parameters in three stages and maturity stage at different depths. **(A)** Average relative importance by three stages of all input parameters. The parameters of tides include tidal elevation (z) and tidal currents (u, v). **(B)** The relative importance of the SST and tides in YSCWM maturity stage below 30 m.

is blocked, and it is difficult to explain the temperature change by relying on SST alone. Therefore, the trend of SST contribution decreases with deepening (Figure 10A).

Warming or cooling mainly drives density changes, causing sea level changes since salinity variation is not significant in the SYS. There is a close correlation between ADT and subsurface thermal structure. The sea level variations are influenced more significantly by those depths where temperature sharply changes, such as the thermocline. Therefore, the ADT contribution is higher at those depths where the temperature fluctuates drastically (Figure 10A), such as the thermocline in the maturity and declining stages and the bottom layer affected by tides. It leads to an average relative importance of 10% and 16% for ADT above and below 15 m depth, respectively.

SSS and SSW are also important parameters (Wu et al., 2012; Klemas and Yan, 2014; Su et al., 2015). The SSS is related to freshwater input (Nieves et al., 2014), which causes density anomalies and then affects the dynamics. The contribution of SSS is less variable from surface to 40 m depth but increases at the bottom (Figure 10A). This may be related to the Yellow Sea Warm Current (YSWC) in the winter, which brings a more salty and warmer water mass, especially at the bottom and manifests in the SSS. Wind forcing changes sea level and also affects ocean mixing, intensifying heat exchange between layers. Southerly winds prevail in summer and northerly winds during winter in the SYS, which causes VSSW to contribute more than USSW (Figure 10A). The vertical distribution of the wind (USSW and VSSW) contribution is roughly same but increases slightly at the bottom (Figure 10A), which is due to the mixed layer deepening during the declining stage of YSCWM.

Tide-induced mixing causes changes in the ocean heat vertical distribution. Even though the overall tidal contribution is weak and less variable, it may be important for a particular stage. During the maturity stage of YSCWM, the tides contribution (u, v, and z in Figure 10A) is about 15% within the mixed layer but can exceed 30% below the mixed layer (Figure 10B) causing the tidal-induced mixing mainly affects the bottom and above 30 m range (Qiao et al., 2004b). It is comparable to the SST contribution (Figure 10B).

4.2 Comparison with other methods

We compared other temperature prediction methods. The SVR and ANN methods have no pre-clustering process and tides. The overall R^2 of SVR and ANN are 0.862/0.888 with the RMSE of 1.506/1.22°C, respectively on the time series. It shows that the GLGBM coupled tides have better accuracy from Figures 11A-C. However, the ANN has similar accuracy above 20 m compared to GLGBM-tides, which may be related to the dominance of SST in this depth range. Additionally, GLGBM-tides allows errors to be smaller and more concentrated, effectively improving model performance, as revealed by the error density distribution (Figure 11D).

We choose H1 data to demonstrate the performance of different methods for fine and continuous data. Since deep learning is more applicable to large data, ANN performs unstable. We implement ANN 20 times to obtain the average temperature estimation. Figure 12 shows the observation from H1 and the reconstructed temperature structure from different

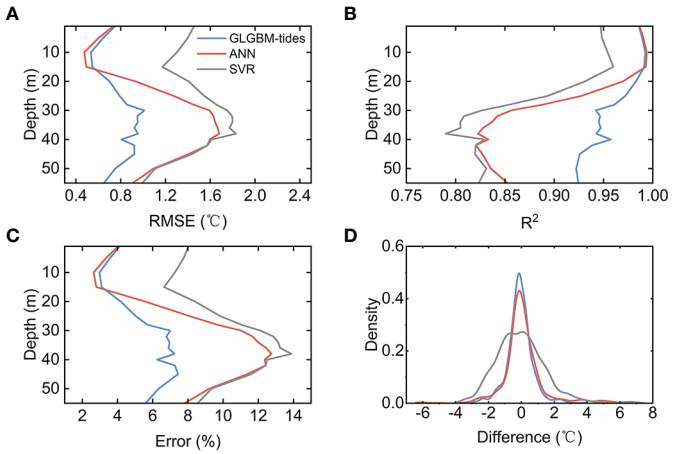


FIGURE 11
The average RMSE **(A)**, R^2 **(B)**, Error **(C)** at the 17 depth levels and absolute difference density distribution **(D)** between the test datasets and estimated ST from GLGBM-tides (blue), ANN (red) and SVR (grey).

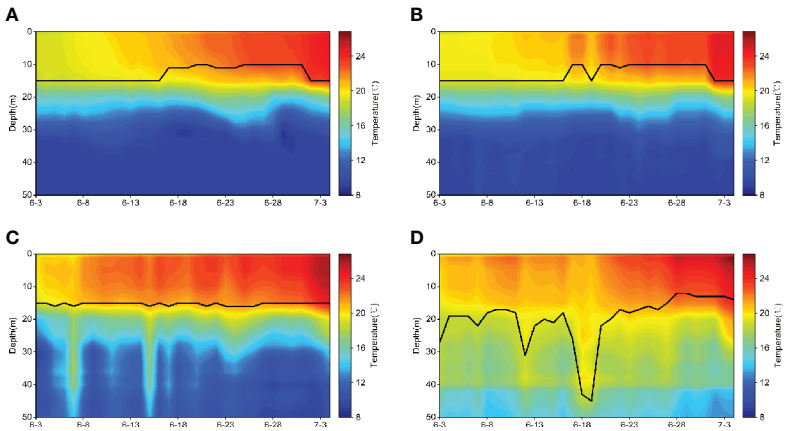


FIGURE 12
Comparison H1 observations **(A)** and reconstruction from GLGBM-tides **(B)**, ANN **(C)** and SVR **(D)** from 0-50 m in H1 period. The MLD is indicated by the solid black line.

methods. The seasonal warming in the upper mixed layer has been reproduced by all methods. Here we adopt the upper boundary of the thermocline as the mixed layer depth (MLD) to further evaluate the performance of models. The reconstructed temperature fields are interpolated to 1 m vertical resolution before calculating MLD. The results show that the MLD is maintained around 10-15 m in the H1 observations (Figure 12A). For the reconstructed temperature field by GLGBM-tides (Figure 12B), the MLD changed generally consistent with the H1 observation. Influenced by atmospheric processes, the MLD becomes shallower from 17 June to 1 July. This process is well reproduced by GLGBM-tides. Reconstructions from other methods failed to capture this variation. The MLD from reconstructed temperature by ANN is stabilized at about 15 m (Figure 12C) while the MLD reconstructed by SVR (Figure 12D) is too deep. The reconstructed temperature from ANN can indicate the trend of YSCWM but has large noises (Figure 12C). The temperature field estimated from SVR fails to reproduce the strong thermocline and YSCWM (Figure 12D). GLGBM-tides can reproduce the vertical temperature structure well compared to the observations. However, the overall estimate of the YSCWM by GLGBM-tides is slightly warmer than the observations from surface to bottom. Hence, the intensity of YSCWM from estimation is weaker. It is noticeable that the reconstruction of the thermocline is well, which assists in predicting the depth of the YSCWM.

We attempt to apply the temperature estimation at the locations of the cruise observations by training the samples from H1 and M1 and use S33-S37 data for verification. The ARMOR3D reanalysis data is used to compare as well. The temperature estimation beyond the topography is deleted. Figure 13 shows the temperature structure of 35°N and 36°N sections (S35 and S36) in three stages of YSCWM. The overall RMSE by all samples of GLGBM-tides and ARMOR3D is 1.781/2.133°C, respectively. It is higher than above due to the spatial heterogeneity of the thermal structure in SYS but the reconstructed vertical temperature structure is still in general agreement with the observations. In the mixed layer, the reconstructed temperature was colder than observation while the ARMOR3D is warmer and the reconstruction has a small zonal variation. It is the result of the training data containing inadequate spatial features. In contrast, the reanalysis data shows a clear spatial difference for fully considering spatial features during production but shows a shallower mixed layer, such as Figures 13B, D. In the declining and disappearance stages, the temperature reconstruction is better for the strong mixing but the ARMOR3D still shows a significant temperature gradient from surface to 35 m depth (Figures 13C, D). The estimates provide a better reconstruction of the thermocline than ARMOR3D (Figures 13A, B). The intensity of the thermocline in the ARMOR3D data is strong (Figures 13A, B) in maturity stage while it is weak in declining stage (Figures 13C, D). The estimated temperature of YSCWM by the GLGBM-tides is slightly warmer especially in the declining stage (Figures 13C, D), but consistent in terms of depth and spatial distribution. The ARMOR3D have the shallower upper boundary of the YSCWM so the temperature of YSCWM is cold as observations (see Figure 13B). Both have

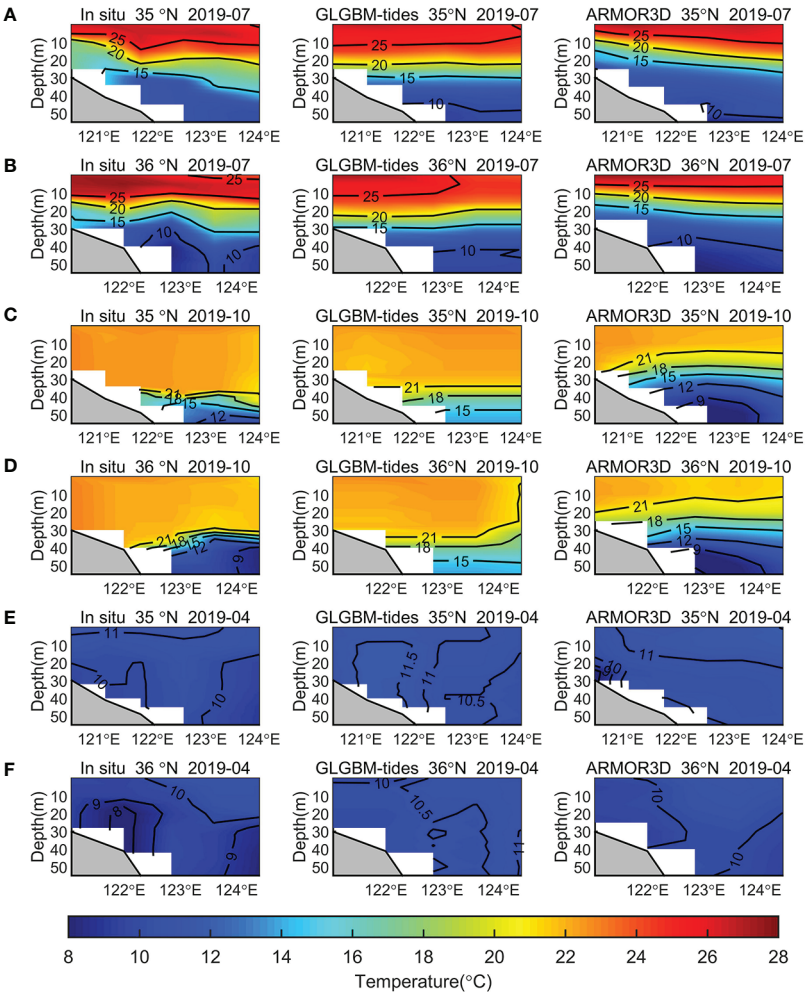


FIGURE 13
Comparison of vertical temperature distributions of *in situ* observations (left), reconstruction from GLGBM-tides (middle) and ARMOR3D (right) in 0-55 m along the 35°N and 36°N section at maturity stage **(A, B)**, declining stage **(C, D)** and disappearance stages **(E, F)** of YSCWM.

good reconstruction of well-mixed temperature structure in the disappearance stage (Figures 13E, F). However, the cold cores in S36 could be observed (Figures 13B, D, F) by cruise data but this special structure is difficult to reproduce. Figure 14 shows the spatial distribution of RMSE in three stages. The accuracy of the proposed method is good from 121°E to 123.5°E. From Figure 14A, the RMSE increases from the center (location of M1) along longitude towards the sides, but with larger differences in farther regions, which may stem from the sparseness of the offshore observations. On the contrary, the RMSE of ARMOR3D decreases gradually from the center to the outside but is similar on the west side of the study area. However, the GLGBM-tides and ARMOR3D have close overall R^2, which are 0.927 and 0.884, respectively. Generally, our reconstruction results are reliable through comparison with ARMOR3D data.

5 Conclusion

This paper proposed the offshore temperature reconstruction method coupled TPXO tide model based on LGBM, using sea surface parameters (ADT, SST, SSS, SSW, tides). The performance of model incorporating tides is quantitatively analyzed. In addition, the temperature estimation is applied spatially and compare with other ARMOR3D. The primary significance of this study is as follows:

(1) The SYS is a typical offshore sea with a huge tidal range, resulting in the difficulty of temperature prediction by classic machine learning method. We coupled the tide model by feeding the estimated tidal elevation and tidal currents by the tide model into a lightweight ensemble learning approach to retrieve SYS thermal structure using small data. The method can generate continuous 3D temperature field at 0-55 m in the SYS at daily

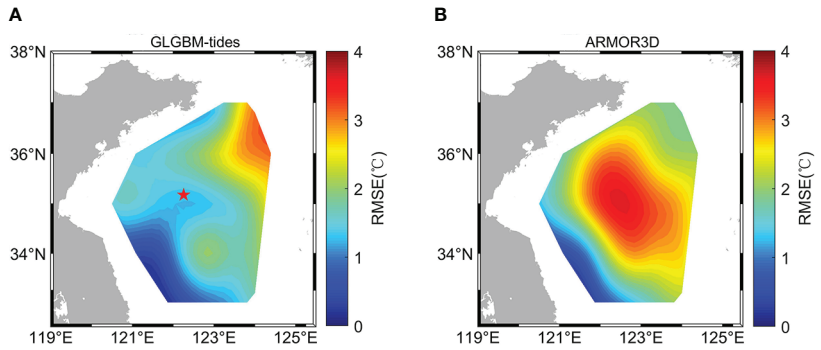


FIGURE 14
Spatial distribution of the overall RMSE by depths from GLGBM-tides **(A)** and ARMOR3D **(B)**. The red star indicates the location of the M1 and H1.

and 0.25° × 0.25° resolution. Experiments demonstrate that proposed method increases the R^2 by 12%, compared to GLGBM and the model tide data mainly improves the accuracy below thermocline in the maturity stage of YSCWM. It has significance for the depth prediction of the YSCWM. Meanwhile, the contribution of tides is comparable with SST in the temperature reconstruction model. The proposed method provides a new explorable direction for reconstructing the offshore thermal structure.

(2) The proposed method is also compared with other machine learning approaches and ARMOR3D dataset. Time series experiments show that the proposed method is superior to SVR and ANN with the RMSE of 0.803°C, 1.506°C, and 1.22°C, respectively. Compared with the cruise data, the method has good and stable results in the three stages of YSCWM. Around the location of M1, the RMSE and R^2 have a good performance in our experiments so our method is effective in the SYS. Furthermore, the temperature reconstruction is comparable to observation-based ARMOR3D dataset, with close R^2 although their RMSE differed in spatial distribution.

Due to the small samples, important oceanic phenomena at longer time scales and larger spatial scales may not be well represented in the reconstructed temperature fields. With sufficient data, better accuracy will be obtained on larger spatial and temporal scale. Therefore, extending the data over longer time and more space to improve the prediction performance of the model is a priority for future work.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Materials. Further inquiries can be directed to the corresponding author.

Author contributions

All authors conceived the research question. FY and JL conducted the analysis on the datasets of the *in situ* observations. FY and GC led the design of the inversion model. FS performed the run of the model. FY and FS wrote the first draft and all authors reviewed and edited the final manuscript. All authors contributed to the article and approved the submitted version.

Funding

This research was jointly supported by the following programs: (1) the Laoshan Laboratory science and technology innovation projects (No. LSKJ202204304); and (2) the Key Laboratory of Marine Science and Numerical Modeling, Ministry of Natural Resources (Grant No. 2021-ZD-01) and (3) the National Natural Science Foundation of China (Grant No. 41806190).

Acknowledgments

The study is benefited from the cruise dataset collected onboard of R/V Lanhai 101 implementing the open research cruise NORC2020-01 supported by NSFC Shiptime Sharing Project.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ali, M. M., Swain, D., and Weller, R. A. (2004). Estimation of ocean subsurface thermal structure from surface parameters: A neural network approach. *Geophys. Res. Lett.* 31, L20308. doi: 10.1029/2004GL021192

Askari, S. (2021). Fuzzy c-means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development. *Expert Syst. Appl.* 165, 113856. doi: 10.1016/j.eswa.2020.113856

Atlas, R., Hoffman, R. N., Ardizzone, J., Leidner, S. M., Jusem, J. C., Smith, D. K., et al. (2011). A cross-calibrated, multiplatform ocean surface wind velocity product for meteorological and oceanographic applications. *B. Am. Meteorol. Soc* 92, ES4–ES8. doi: 10.1175/2010BAMS2946.2

Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L., and Amirat, Y. (2015). Physical human activity recognition using wearable sensors. *Sensors-Basel* 15, 31314–31338. doi: 10.3390/s151229858

Banzon, V., Smith, T. M., Chin, T. M., Liu, C., and Hankins, W. (2016). A long-term record of blended satellite and *in situ* sea-surface temperature for climate monitoring, modeling and environmental studies. *Earth System Sci. Data* 8, 165–176. doi: 10.5194/essd-8-165-2016

Bi, C., Yao, Z., Bao, X., Zhang, C., Ding, Y. A., Liu, X., et al. (2021). The sensitivity of numerical simulation to vertical mixing parameterization schemes: a case study for the yellow Sea cold water mass. *J. Oceanol. Limnol.* 39, 64–78. doi: 10.1007/s00343-019-9262-y

Boutin, J., Vergely, J. L., Marchand, S., D'Amico, F., Hasson, A., Kolodziejczyk, N., et al. (2018). New SMOS Sea surface salinity with reduced systematic errors and improved variability. *Remote Sens. Environ.* 214, 115–134. doi: 10.1016/j.rse.2018.05.022

Chu, P. C., Fralick, C. R., Haeger, S. D., and Carron, M. J. (1997). A parametric model for the yellow Sea thermal variability. *J. Geophys. Res.* 102, 10499–10507. doi: 10.1029/97JC00444

Diao, X., Si, G., Wei, C., and Yu, F. (2022). Structure and formation of the south yellow Sea water mass in the spring of 2007. *J. Oceanol. Limnol.* 40, 55–65. doi: 10.1007/s00343-021-0206-y

Dong, L., Qi, J., Yin, B., Zhi, H., Li, D., Yang, S., et al. (2022). Reconstruction of subsurface salinity structure in the south China Sea using satellite observations: A LightGBM-based deep forest method. *Remote Sens. Basel* 14, 3494. doi: 10.3390/rs14143494

Guinehut, S., Dhomps, A. L., Larnicol, G., and Le Traon, P. Y. (2012). High resolution 3-d temperature and salinity fields derived from *in situ* and satellite observations. *Ocean Sci.* 8, 845–857. doi: 10.5194/os-8-845-2012

Guo, Y., Mo, D., and Hou, Y. (2021). Interannual to interdecadal variability of the southern yellow Sea cold water mass and establishment of "Forcing mechanism bridge". *J. Mar. Sci. Eng.* 9, 1316. doi: 10.3390/jmse9121316

Huang, B. Y., Liu, C. Y., Banzon, V., Freeman, E., Graham, G., Hankins, B., et al. (2021). Improvements of the daily optimum interpolation Sea surface temperature (DOISST) version 2.1. *J. Climate* 34, 2923–2939. doi: 10.1175/JCLI-D-20-0166.1

Hwang, J. H., Van, S. P., Choi, B. J., Chang, Y. S., and Kim, Y. H. (2014). The physical processes in the yellow Sea. *Ocean Coast. Manage.* 102, 449–457. doi: 10.1016/j.ocecoaman.2014.03.026

Klemas, V., and Yan, X. (2014). Subsurface and deeper ocean remote sensing from satellites: An overview and new results. *Prog. Oceanogr.* 122, 1–9. doi: 10.1016/j.pocean.2013.11.010

Landschutzer, P., Gruber, N., Bakker, D., Schuster, U., Nakaoka, S., Payne, M. R., et al. (2013). A neural network-based estimate of the seasonal to inter-annual variability of the Atlantic ocean carbon sink. *Biogeosciences* 10, 7793–7815. doi: 10.5194/bg-10-7793-2013

Lee, J. H., Pang, I. C., and Moon, J. H. (2016). Contribution of the yellow Sea bottom cold water to the abnormal cooling of sea surface temperature in the summer of 2011. *J. Geophys. Res.* 121, 3777–3789. doi: 10.1002/2016JC011658

Li, J., Jiang, F., Wu, R., Zhang, C., Tian, Y. J., Sun, P., et al. (2021). Tidally induced temporal variations in growth of young-of-the-Year pacific cod in the yellow Sea. *J. Geophys. Res.* 126, e2020JC016696. doi: 10.1029/2020JC016696

Li, J., Li, G., Xu, J., Dong, P., Qiao, L. L., Liu, S. D., et al. (2016). Seasonal evolution of the yellow Sea cold water mass and its interactions with ambient hydrodynamic system. *J. Geophys. Res.* 121, 6779–6792. doi: 10.1002/2016JC012186

Lin, F., Asplin, L., and Wei, H. (2021). Summertime M2 internal tides in the northern yellow Sea. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.798504

Liu, X., Huang, B., Huang, Q., Wang, L., Ni, X. B., Tang, Q. S., et al. (2015). Seasonal phytoplankton response to physical processes in the southern yellow Sea. *J. Sea Res.* 95, 45–55. doi: 10.1016/j.seares.2014.10.017

Li, A., Yu, F., Si, G., and Wei, C. (2017a). Long-term temperature variation of the southern yellow Sea cold water mass from 1976 to 2006. *Chin. J. Oceanol. Limn.* 35, 1032–1044. doi: 10.1007/s00343-017-6037-1

Li, A., Yu, F., Si, G. C., and Wei, C. J. (2017b). Long-term variation in the salinity of the southern yellow Sea cold water mass 1976-2006. *J. Oceanogr.* 73, 321–331. doi: 10.1007/s10872-016-0405-x

Lü, X., Qiao, F., Xia, C., Wang, G., and Yuan, Y. (2010). Upwelling and surface cold patches in the yellow Sea in summer: Effects of tidal mixing on the vertical circulation. *Cont. Shelf. Res.* 30, 620–632. doi: 10.1016/j.csr.2009.09.002

Lu, W. F., Su, H., Yang, X., and Yan, X. H. (2019). Subsurface temperature estimation from remote sensing data using a clustering-neural network method. *Remote Sens. Environ.* 229, 213–222. doi: 10.1016/j.rse.2019.04.009

Maze, G., Mercier, H., Fablet, R., Tandeo, P., Radcenco, M. L., Lenca, P., et al. (2017). Coherent heat patterns revealed by unsupervised classification of argo temperature profiles in the north Atlantic ocean. *Prog. Oceanogr.* 151, 275–292. doi: 10.1016/j.pocean.2016.12.008

Meng, L. S., Yan, C., Zhuang, W., Zhang, W. W., Geng, X. P., and Yan, X. H. (2022). Reconstructing high-resolution ocean subsurface and interior temperature and salinity anomalies from satellite observations. *IEEE T. Geosci. Remote* 60, 1–14. doi: 10.1109/TGRS.2021.3109979

Nieves, V., Wang, J., and Willis, J. K. (2014). A conceptual model of ocean freshwater flux derived from sea surface salinity. *Geophys Res. Lett.* 41 (18), 6452–6458. doi: 10.1002/2014GL061365

Parard, G., Charantonis, A. A., and Rutgerson, A. (2015). Remote sensing the sea surface CO_2 of the Baltic Sea using the SOMLO methodology. *Biogeosciences* 12, 3369–3384. doi: 10.5194/bg-12-3369-2015

Qiao, F., Ma, J., Yang, Y., and Yuan, Y. (2004a). Simulation of the temperature and salinity along 36°N in the yellow Sea with a wave-current coupled model. *J. Korean Soc Oceanogr.* 39, 35–45. Available at: https://koreascience.kr/article/JAKO200411922304273.page.

Qiao, F., Xia, C., Shi, J., Ma, J., Ge, R., and Yuan, Y. (2004b). Seasonal variability of thermocline in the yellow Sea. *Chin. J. Oceanol. Limn.* 22, 299–305. doi: 10.1007/BF02842563

Sambe, F., and Suga, T. (2022). Unsupervised clustering of argo temperature and salinity profiles in the mid-latitude Northwest pacific ocean and revealed influence of the kuroshio extension variability on the vertical structure distribution. *J. Geophys. Res.* 127, e2021JC018138. doi: 10.1029/2021JC018138

Sun, F., Yu, F., Si, G., Wang, J., Xu, A., Pan, J., et al. (2022). Characteristics and influencing factors of frontal upwelling in the yellow Sea in summer. *Acta Oceanol. Sin.* 41, 84–96. doi: 10.1007/s13131-021-1967-z

Su, H., Wang, A., Zhang, T. Y., Qin, T., Du, X. P., and Yan, X. H. (2021). Super-resolution of subsurface temperature field from remote sensing observations based on machine learning. *Int. J. Appl. Earth Obs.* 102, 102440. doi: 10.1016/j.jag.2021.102440

Su, H., Wu, X. B., Yan, X. H., and Kidwell, A. (2015). Estimation of subsurface temperature anomaly in the Indian ocean during recent global surface warming hiatus from satellite measurements: A support vector machine approach. *Remote Sens. Environ.* 160, 63–71. doi: 10.1016/j.rse.2015.01.001

Su, H., Yang, X., Lu, W., and Yan, X. (2019). Estimating subsurface thermohaline structure of the global ocean using surface remote sensing observations. *Remote Sens. Basel* 11, 1598. doi: 10.3390/rs11131598

Taburet, G., Sanchez-Roman, A., Ballarotta, M., Pujol, M., Legeais, J., Fournier, F., et al. (2019). DUACS DT2018: 25 years of reprocessed sea level altimetry products. *Ocean Sci.* 15, 1207–1224. doi: 10.5194/os-15-1207-2019

Wang, S., Azzari, G., and Lobell, D. B. (2019). Crop type mapping without field-level labels: Random forest transfer and unsupervised clustering techniques. *Remote Sens. Environ.* 222, 303–317. doi: 10.1016/j.rse.2018.12.026

Wang, B., Hirose, N., Kang, B., and Takayama, K. (2014). Seasonal migration of the yellow Sea bottom cold water. *J. Geophys. Res.* 119, 4430–4443. doi: 10.1002/2014JC009873

Wan, X., Liu, S., and Ma, W. (2022). Numerical simulation of double warm tongues related to the bifurcation of the yellow Sea warm current. *Cont. Shelf. Res.* 236, 104680. doi: 10.1016/j.csr.2022.104680

Wu, X., Yan, X., Jo, Y., and Liu, W. T. (2012). Estimation of subsurface temperature anomaly in the north Atlantic using a self-organizing map neural network. *J. Atmos. Ocean. Tech.* 29, 1675–1688. doi: 10.1175/JTECH-D-12-00013.1

Xin, M., Ma, D., and Wang, B. (2015). Chemicohydrographic characteristics of the yellow Sea cold water mass. *Acta Oceanol. Sin.* 34, 5–11. doi: 10.1007/s13131-015-0681-0

Yang, Y., Li, K. P., Du, J. T., Liu, Y. L., Liu, L., Wang, H. W., et al. (2019). Revealing the subsurface yellow Sea cold water mass from satellite data associated with typhoon muifa. *J. Geophys. Res.* 124, 7135–7152. doi: 10.1029/2018JC014727

Yao, Z., He, R., Bao, X., Wu, D., and Song, J. (2012). M2 tidal dynamics in bohai and yellow seas: a hybrid data assimilative modeling study. *Ocean Dynam.* 62, 753–769. doi: 10.1007/s10236-011-0517-1

Yu, X., Guo, X., and Takeoka, H. (2016). Fortnightly variation in the bottom thermal front and associated circulation in a semienclosed Sea. *J. Phys. Oceanogr.* 46, 159–177. doi: 10.1175/JPO-D-15-0071.1

Yu, F., Ren, Q., Diao, X., Wei, C., and Hu, Y. (2022). The sandwich structure of the southern yellow Sea cold water mass and yellow Sea warm current. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.767850

Zhang, S., Wang, Q., Lü, Y., Cui, H., and Yuan, Y. (2008). Observation of the seasonal evolution of the yellow Sea cold water mass in 1996-1998. *Cont. Shelf. Res.* 28, 442–457. doi: 10.1016/j.csr.2007.10.002

Zhu, J., Shi, J., Guo, X., Gao, H., and Yao, X. (2018). Air-sea heat flux control on the yellow Sea cold water mass intensity and implications for its prediction. *Cont. Shelf. Res.* 152, 14–26. doi: 10.1016/j.csr.2017.10.006

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 22 December 2022
DOI 10.3389/fmars.2022.1093623

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Hao Zhang,
Henan Agricultural University, China
Yong Zhang,
Sun Yat-sen University, China

*CORRESPONDENCE
Bo Wang
wb@ouc.edu.cn
Jingjie Hu
hujingjie@ouc.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 09 November 2022
ACCEPTED 09 December 2022
PUBLISHED 22 December 2022

CITATION
Wang Y, Xin C, Zhu B, Wang M, Wang T, Ni P, Song S, Liu M, Wang B, Bao Z and Hu J (2022) A new non-invasive tagging method for leopard coral grouper (*Plectropomus leopardus*) using deep convolutional neural networks with PDE-based image decomposition. *Front. Mar. Sci.* 9:1093623. doi: 10.3389/fmars.2022.1093623

COPYRIGHT
© 2022 Wang, Xin, Zhu, Wang, Wang, Ni, Song, Liu, Wang, Bao and Hu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

A new non-invasive tagging method for leopard coral grouper (*Plectropomus leopardus*) using deep convolutional neural networks with PDE-based image decomposition

Yangfan Wang[1,2], Chun Xin[1], Boyu Zhu[3], Mengqiu Wang[1], Tong Wang[1], Ping Ni[1], Siqi Song[2], Mengran Liu[1,2], Bo Wang[1,2]*, Zhenmin Bao[1,2] and Jingjie Hu[1,2]*

[1]Ministry of Education Key Laboratory of Marine Genetics and Breeding, College of Marine Life Science, Ocean University of China, Qingdao, China, [2]Key Laboratory of Tropical Aquatic Germplasm of Hainan Province, Sanya Oceanographic Institution, Ocean University of China, Sanya, China, [3]Faculty of Arts and Science, University of Toronto, Toronto, ON, Canada

External tagging methods can aid in the research of leopard coral grouper (*Plectropomus leopardus*) in terms of its spatio-temporal behavior at population and individual scales. However, due to the strong exclusion ability and the damage to the body wall of *P. leopardus*, the retention rate of traditional invasive tagging methods is low. To develop a non-invasive identification method for *P. leopardus*, we adopted a multiscale image processing method based on matched filters with Gaussian kernels and partial differential equation (PDE) multiscale hierarchical decomposition with the deep convolutional neural network (CNN) models VGG19 and ResNet50 to extract shape and texture image features of individuals. Then based on image features, we used three classifiers Random forest (RF), support vector machine (SVM), and multilayer perceptron (MLP)) for individual recognition on sequential images of *P. leopardus* captured for 50 days. The PDE, ResNet50 and MLP combination obtained a maximum accuracy of 0.985 $\pm$ 0.045 on the test set. For individual temporal tracking recognition, feature extraction and model training were performed using images taken in 1-20 days. The classifier could achieve an accuracy of 0.960 $\pm$ 0.049 on the test set consisting of images collected in the periods of 20-50 days. The results show that CNNs with the PDE decomposition can effectively and accurately identify *P. leopardus*.

KEYWORDS

***Plectropomus leopardus*, non-invasive tagging method, convolutional neural networks, PDE-based image decomposition, complex trait**

1 Introduction

P. leopardus represents one of the most economically significant chordate and is mainly distributed in the Western Pacific Ocean along the coasts of China, Vietnam, and Thailand (Yang et al., 2020). *P. leopardus* has a high economic value in the international market due to its high nutritional profile and plays a vital role in marine ecosystems (Xia et al., 2020). However, the *P. leopardus* industry has encountered many challenges in recent years, including devastating diseases and environmental stresses, which caused a large amount of economic loss and hampered the healthy and sustainable development of the *P. leopardus* industry (Rimmer and Glamuzina, 2019). Therefore, it is urgent to advance the scientific culture of *P. leopardus* and to select and breed new species with superior characteristics. Designing effective external tagging methods for long-term and stable tracking identification of *P. leopardus* is not only essential for successful breeding but also a concern for ecologists conducting population dynamics studies (Williams et al., 2002; Zhuang et al., 2013), as well as revealing the ecological significance of fish endotherms (Watanabe et al., 2015), and studying the life history of fish such as foraging, migration and reproduction (Quinn et al., 1989; Ogura and Ishida, 1995; Yano et al., 1996; Hinch et al., 2002; Welch et al., 2004; Sulak et al., 2009; Døving et al., 2011).

Traditionally, individual recognition has been accomplished by capturing animals and placing visible and unique marks on them. The traditional marking methods include implanting acoustic markers inside the abdominal cavity of fish (Shi et al., 2022), and then using the positioning system to track the acoustic markers. The individual unique electric field generated by electric fish discharges was used for recording and tracking (Raab et al., 2022). Due to the strong exclusion ability and the damage to the body of *P. leopardus*, the retention rate of traditional invasive tagging methods is low (Bolger et al., 2012). Besides, the infection rate and mortality rate of implanted marker fish are relatively high (Shi et al., 2022), and the marker will also affect the original normal life of fishes in the water, and with the extensive use of individual markers, it is also a hazard to the environment (Šmejkal et al., 2020), while individual electric field tracking is only applicable to fish that can generate electricity. This makes it difficult for breeders to manage good individuals, which is not conducive to the implementation of accurate breeding by tracking the growth of individuals. Recently, molecular genetic markers such as RFLP (restriction fragment length polymorphism), RAPD (random amplified polymorphism DNA), SSR (simple sequence repeat), and SNP (single nucleotide polymorphism) have also been widely used to study the population and individual recognition (Reed et al., 1997; Wang, 2016). However, these methods are not suitable for a larger population because of inconsistency, inconvenience, and higher cost, among others. Currently, photographic mark-recapture has gained popularity because of the advances in digital photography and image processing software. The abundance of species with variable natural marking patterns makes this an attractive method for many researchers. The image mark method has been employed particularly in the studies of populations of marine mammals and mammalian terrestrial predators (Karanth and Nichols, 1998; Forcada and Aguilar, 2000; Langtimm et al., 2004; Fearnbach et al., 2012). Some image processing methods have been used to extract, store, and compare pattern information from digital images (Bolger et al., 2012). With the development of computer vision, deep learning (DL) methods, such as convolutional neural networks (CNNs) are emerging as possibly powerful tools for individual recognition and long-term tracking (He et al., 2016; Redmon et al., 2016). Numerous broad models of convolutional neural networks, such as AlexNet, Inception, VGG19, ResNet50, etc., have been presented (Kamilaris and Prenafeta-Boldú, 2018). These models are trained using public datasets (e.g., CIFAR-10, ImageNet datasets, etc.) and used to perform Multi-Category tasks for particular items. Considering the unique body shape and texture patterns of different *P. leopardus* individuals, it is a promising technical route to extract and identify the body surface features using CNN as an alternative method against traditional invasive tagging methods.

In this study, we used a novel multiscale image processing method based on matched filters with Gaussian kernels and partial differential equation (PDE) multiscale hierarchical decomposition (Wang et al., 2013) to segment the shape features of *P. leopardus* images. Two deep CNN models, VGG19 and ResNet50, were implemented to extract the texture features. Then based on the shape and texture features, three classifiers (Random forest (RF) (Kamilaris and Prenafeta-Boldú, 2018), support vector machine (SVM) (Cortes and Vapnik, 1995), and multilayer perceptron (MLP) (LeCun et al., 2015) were compared for individual recognition on sequential images of *P. leopardus* captured over the course of 50 days. Finally, we found that the combination of PDE and CNN methods could achieve the best accurate recognition of *P. leopardus*. This is the first time, to our knowledge, that image recognition analysis has been applied to the tracking of *P. leopardus*. Our results will provide a new vision for using non-invasive tagging of *P. leopardus*.

2 Materials and methods

2.1 Data acquisition

P. leopardus used in this study were obtained from Sanya, Hainan Province. 50 individuals were randomly selected from a breeding population of 10,000 *P. leopardus*, and reared under laboratory conditions. The numbered clapboards were added to

the rearing pool to facilitate individual identification. In the 50 days from September 3, 2022, to October 23 2022, each individual was taken out from the rearing pool daily and placed on a smooth white foam plastic plate. The *P. leopardus* were anesthetized by immersion in seawater which containing MS222 (tricaine methanesulfonate) with a concentration of 100 mg/L and kept in the solution for 3 min after loss of body posture (Savson et al., 2022). After its body was fully stretched, photos were taken directly for each individual using a mobile device. Then they were placed back in the pond immediately. At the end of the experiment, 50 images were taken for each individual. So, we obtained a total of 2500 images for all individuals.

2.2 Image feature extraction

2.2.1 PDE-based feature extraction

We used a PDE-based multiscale decomposition method to extract the shape features of *P. leopardus* images. For the shape detection, we used matched filtering with Gaussian kernel (MFGK) ker(x,y; a,b)=−exp(−a^{-1}(x−b)^2/2σ^2) (Chaudhuri et al., 1989), and the computed MFGK response image was as follows:

$$M_{ker}(x, y; a, b) = \max_{\theta}(r_{\theta}(ker(x, y; a, b)) * Img(x, y)) \quad (1)$$

where Img, (x, y), σ, ker, a, and b denoted an image, a two-dimensional pixel position, the standard deviation of image gray value in Gaussian convolution kernel, two-dimensional Gaussian functions, the dilation parameter (also known as scaling parameter), and the translation parameter, respectively. r_θ rotated the kernel function with an angle θ, and * represented the convolution operation in variables (x and y).

The normalized response image was defined as follows:

$$f = (M_{ker}(x, y; a, b) - \mu)/s$$

where μ and s were the mean and standard deviation of the enhanced MFGK image M_{ker}(x,y;a,b) The multiscale hierarchical decomposition of an image f was defined as follows (Wang et al., 2013). Given an initial scale parameter λ_0 and the PDE-based total variation (TV) function (Rudin et al., 1992)

$$J(f, \lambda) = \lambda || v_\lambda ||^2_{L2} + || u_\lambda ||_{BV}$$

where BV stood for the homogenous bounded total variation space equipped with the norm of total variation

$$|| \cdot ||_{BV} = || \cdot ||_{L^1} = \int_\Omega \sqrt{(u_\lambda)^2_x + (u_\lambda)^2_y}$$

$$f = u_0 + v_0, \ \ where \ [u_0, v_0] := argmin|_{u+v=f} \ J(f, \lambda_0)$$

$$v_k = u_{k+1} + v_{k+1}, \ \ k = 0, 1, \ldots, \ \lambda_k = \lambda_0 2^{k+1}$$

where $[u_{k+1}, v_{k+1}$:=argmin$|_{u+v=v_k} J(v_k, \lambda_0 2^{k+1})$].

Based on the above enhancement with MFGK and multiscale hierarchical decomposition, many line maps u_λ were generated at varying image resolutions, representing different levels of line details to avoid the possible failure of feature extraction caused by a single-scale segmentation. The initial scaling parameter was λ_0 = 0.01 in the multiscale hierarchical decomposition.

The binarization is performed as follows:

$$out(x, y) = \begin{cases} 1 & \bar{u}(x, y) \ \le u(x, y) \\ 0 & otherwise \end{cases}$$

where out stands for the finally segmented binary mask of the *P. leopardus* image.

2.2.2 CNN-based feature extraction

With the development of deep learning algorithms, many general models of convolutional neural networks have been proposed, such as AlexNet, Inception, VGGNet, ResNet, etc. (Kamilaris and Prenafeta-Boldú, 2018). These models have been trained on large public datasets (e.g., CIFAR-10, ImageNet datasets, etc.) (Lecun et al., 1998) to achieve the goal of multiple-classification tasks for specific items. After training, the deep layers and convolutional kernels in these models can explore the visual characteristics of images. For other classification tasks, new characteristics can be extracted with the help of the pre-trained convolutional layers and used as input for many classifiers. This method of applying the "knowledge" gained from training on a specific dataset to a new domain is also known as migration learning (Yoshua, 2011). In this study, the VGG19 and ResNet50 of CNN models were used for image feature extraction. The weights of each convolutional layer of VGG19 or ResNet50 were frozen and fed into a new CNN. The output of the last pooling layer of the new CNN was then taken as the extracted image features. After feature extraction using VGG19 or ResNet50, a 4096-1D or 2048-1D vector of features was obtained, respectively.

LeNet-5 Convolutional Neural Network (Lecun et al., 1998), as a classic CNN, has only two convolution layers and a simple structure, which is suitable for preliminary evaluation of the complexity of the dataset. The structure of the model is as follows. Input layer: single input is a 224*224*3 RGB three-channel image without feature extraction; convolutional layer 1, containing 6 convolutional kernels with the size of 5*5 pixels using activation function ReLU; batch normalization layer 1; maximum pooling layer 1, with the pooling size of 2*2; convolutional layer 2, containing 16 convolutional kernels with the size of 5*5 pixels using activation function ReLU; batch normalization layer 2; Maximum pooling layer 2, with the pooling size of 2*2; fully connected layer 1, containing 120 neurons using activation function ReLU; batch normalization layer 3; fully connected layer 2, containing 84 neurons using activation function ReLU; batch normalization layer 4; output layer, outputting 20 classes using activation function softmax. The loss function is

cross entropy and the optimizer is Adam. When training on the raw dataset, batch_size is 30 and epoch is 50.

VGG is a type of CNN model developed by the Oxford Robotics Institute (Simonyan and Zisserman, 2015). VGG19 uses an architecture of very small (3x3) convolution filters and pushes the depth to 19 weight layers. There are five building blocks in VGG19, consisting of 16 convolutional layers and 3 fully connected layers. The first and second building blocks have two convolutional layers and one pooling layer, respectively, and four convolutional layers and one pooling layer exist in the third and fourth building blocks. The last building block contains four convolutional layers.

The architecture of the residual network consists of 50 layers named ResNet50 (He et al., 2016). There is an extra identity in ResNet50 where the ResNet model predicts the delta needed in the final prediction from one layer to the next. ResNet50 provides alternate paths to allow gradient flow which helps to solve the problem of gradient disappearance. The ResNet model uses identity mapping to bypass the weight layer of the CNN when the current layer is not required. This model solves the overfitting problem of the training set with the presence of 50 layers in the feature extraction of ResNet50 (Stateczny et al., 2022).

In this study, the PDE-based multiscale decomposition and the Convolutional Neural Network models, VGG19 and ResNet50, were used to extract shape and texture features on the original image datasets. A total of five combined datasets are generated, which are called: PDE+ raw dataset, VGG19+ raw dataset, ResNet50+ raw dataset, PDE+VGG19+ raw dataset, and PDE+ResNet50+ raw dataset. After feature extraction, the image features obtained from each feature extraction method are visualized using the t-SNE algorithm (Linderman et al., 2019) to visually examine the effectiveness of several feature extraction methods.

2.3 Training of classifiers based on extracted features

The feature-extracted dataset is used as input to train Random Forest (RF), Support Vector Machine (SVM), and Multi-layer Perceptron (MLP), models, respectively. The RF models were trained using default parameters. The SVM models were trained with RBF kernel using default parameters.

The structure of the multi-layer perceptron was: input layer, where the number of neurons contained depends on the length of the features used (2048 for PDE features, 4096 for VGG19 features, 2048 for ResNet50 features); fully connected layer, containing 1024 neurons using activation function ReLU (LeCun et al., 2015); batch normalization layer; output layer, outputting 50 classes using activation function softmax. The loss function was cross entropy and the optimizer was Adam (LeCun et al., 2015). When training on the raw dataset, batch_size is 30 and epoch is 50.

The essential architecture of our method for fully automated segmentation and identification of *P. leopardus* is shown in Figure 1.

2.4 Model assessment indicators

In a multi-classification task, there are differences in the predicting ability of the model for different categories, and there may be category imbalance in the predicting results. Since the accuracy rate simply calculates the ratio of the number of correctly predicted samples to the total number of samples, ignoring the predicting ability of the model for different categories, it is hard to objectively measure the predicting effect of the model. In order to

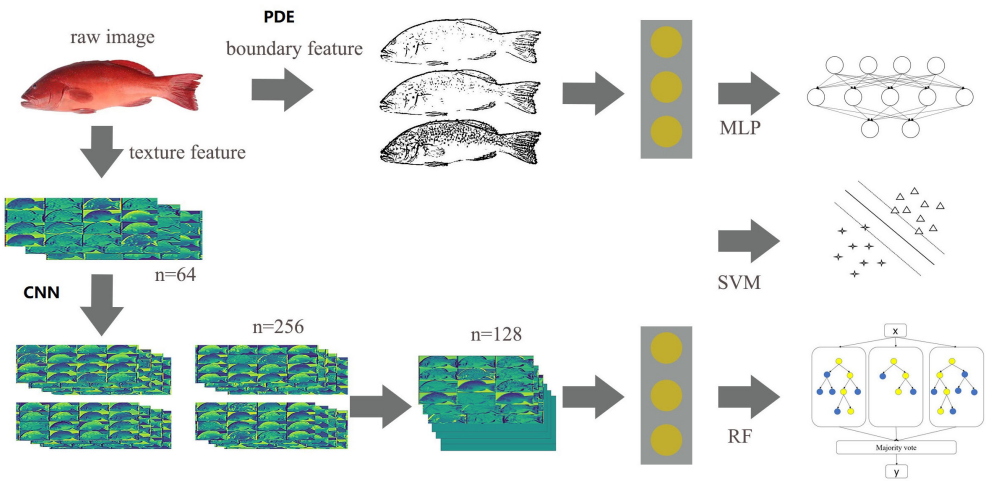


FIGURE 1
The essential architecture of our method for fully automated segmentation and identification of *P. leopardus*.

measure the model's comprehensive predicting ability for each category, the accuracy for each category should be taken into account, so the Precise, Recall and Macro-F1 Score are selected as evaluation indicators (Zhou et al., 2021). The calculation method is as follows.

True Positives (TP): all cases where we have predicted YES and the actual result was YES. True Negatives (TN): all cases where we have predicted NO and the actual result was NO. False Positives (FP): all cases where prediction was YES, but the actual result was NO ('Type I error'). False Negatives (FN): all cases where prediction was NO, but the actual result was YES ('Type II error').

Precision is the proportion of positive samples that are correctly predicted out of all samples that are predicted to be positive:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall is the proportion of positive samples that are correctly predicted out of all actual positive samples (including the positive samples that were predicted incorrectly).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{TN}}$$

F1-Score is the harmonic mean of precision and recall.

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Macro-F1 is the mean of F1-Score for each category, where N is the total number of categories.

$$\text{Macro F1} = \frac{\sum_{N}^{i} \text{F1}_i}{\text{N}}$$

2.5 Software and hardware environment

In this study, the Python 3.8.10 environment was used with the scikit-image library for feature extraction, the scikit-learn 0.24.0 library for principal component analysis and the construction of random forest and support vector machine models, and the tensorflow 2.3.1 library for CNN-based feature extraction and the training of multilayer perceptrons. The tsne library was used to accomplish the t-SNE downscaling and visualization in the R 4.1.1 environment.

3 Results

3.1 PDE-based feature extraction

The results of the illustrative segmentation of *P. leopardus* using the PDE multiscale decomposition method with different scale parameters are shown in Figure 2. Obviously, the camera image can be used for good segmentation with the selection of more growth rings of body shape. Meanwhile, the segmentation of shape contours in the image can be still detected even though the original image was degraded by some body color; hence, our segmentation method was robust in noise and color.

3.2 CNN-based feature extraction

As shown in Figure 2, the PDE method can obtain more details of the shape of *P. leopardus* compared with the ResNet50 model of CNNs. By visualizing several convolutional layers in the ResNet50 model (Figure 1), we found that some kernels in different layers could distinguish smaller tubular and periodic structures in *P. leopardus* images, which made ResNet50 more effective in the extraction of texture details.

The shape and texture features obtained by PDE-based and CNN-based methods were visualized using the tSNE software (Figure 3). For the shape features obtained by PDE, the points of different categories overlapped each other and were difficult to distinguish (Figure 3A). While we found that the CNN-based texture features of the same individuals were gathered into a cluster, reflecting the intra-category consistency and inter-category dissimilarity, for example, individuals of 4, 5, 14, 15, 16, 17, 18, 19 in ResNet50 features (Figure 3B) and individuals of 5, 15, 19 in VGG19 features (Figure 3C). Features of the same individuals using the ResNet50 model were more likely to gather into clusters than the VGG19 features, suggesting that the ResNet50 feature may extract more small texture information from images than VGG19 features.

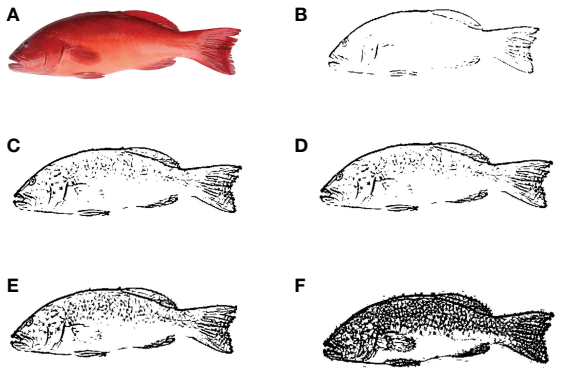


FIGURE 2
Segmentation results produced by multiscale hierarchical decomposition using PDE with $\lambda 0 = 0.01$ and $\lambda i = \lambda 02^i$ **(A)** original image; **(B–F)** segmented image at scaling parameters $\lambda 1$, $\lambda 2$, $\lambda 3$, $\lambda 4$, and $\lambda 5$, respectively.

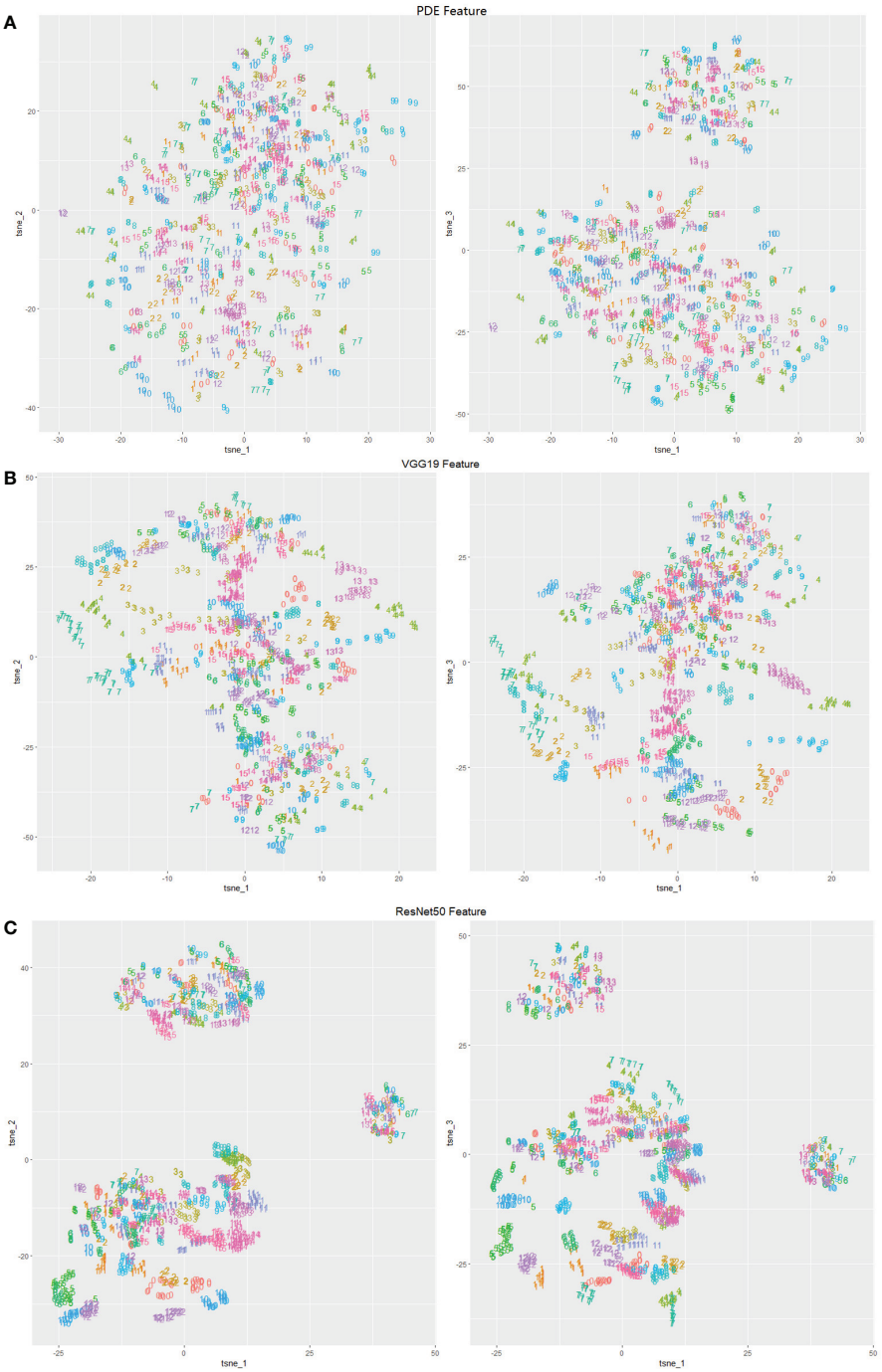


FIGURE 3
Visualization of feature-extraction methods (number labels in the range of 1~20 denote 20 individuals randomly sampled from all the *P. leopardus*) **(A)** PDE feature; **(B)** CNN ResNet50 feature; **(C)** CNN VGG19 feature.

3.3 Prediction performance of combinations with different features and classifiers

In this section, five-fold Cross-validation (5-fold CV) was used to assess the prediction performance of the different methods in the *P. leopardus* data set. For 5-fold CV, the data set was divided into five mutually exclusive subsets; four of five formed the estimation set (ES) for fitting input feature effects and the fifth subset was used as a test set (TS). Three methods (RF, SVM and MLP) were trained on the feature-extracted (PDE, VGG19 and ResNet50) datasets, and the traditional LeNet-5 convolutional neural network dataset of the 224*224-pixel images from the raw dataset, respectively (Table 1).

Among the classifiers trained on the only PDE features for the dataset, PDE+ MLP achieved the best prediction (Macro-F1 Score 0.748 ± 0.066), followed by PDE + SVM (Macro-F1 Score 0.717 ± 0.076). The predicting performance of RF was poor with Macro-F1 score of only 0.681 ± 0.117. Compared with classifiers trained on PDE features, the simple CNN LeNet-5 with a simple structure had a significant improvement in the predicting effect with Macro-F1 score of 0.861 ± 0.069. For the deep CNN VGG19 features, VGG19 + MLP achieved the best prediction (Macro-F1 Score 0.872 ± 0.068) followed by VGG19 + SVM (Macro-F1 Score 0.849 ± 0.071) and VGG19 + RF (Macro-F1 Score 0.813 ± 0.079). Only VGG19 + MLP outperformed the simple LeNet-5 model (Macro-F1 Score 0.861 ± 0.069) with a Macro-F1 score increased about 0.011. After training on ResNet50 texture features, any classifier can achieve better predictions than any other combinations on VGG19 texture features. ResNet50 + MLP achieves the best prediction (Macro-F1 Score 0.927 ± 0.043) followed by ResNet50 + SVM (Macro-F1 Score 0.925 ± 0.048). It is interesting that SVM can also achieve similar performance on ResNet50-extracted features.

If we combined PDE shape features with ResNet50 or VGG19 text features to form a new feature set, any classifier can achieve better predictions than the feature set of PDE, VGG19, or ResNet50. In the PDE+ResNet50 dataset, the maximum accuracy was Macro-F1 Score 0.985 ± 0.045 for MLP. In the PDE+VGG19 dataset, the maximum accuracy was Macro-F1 Score 0.949 ± 0.069 for MLP. We, therefore, decided to take PDE+ResNet50+MLP and PDE+ResNet50 +SVM as the experimental model to identify individuals in the following analyses.

3.4 Predictions effect of the model on training sets of different sizes

Due to the constraint of time and labor costs in actual application scenarios, it is often difficult to obtain large datasets.

TABLE 1 Predictive accuracies obtained with different combination of features and classifiers by 5-fold CV.

Input feature	Classifiers	Metrics		
		Precision	Recall	Macro-F1 score
LeNet-5		0.851 ± 0.078	0.869 ± 0.061	0.861 ± 0.069
ResNet50	RF	0.881 ± 0.082	0.892 ± 0.073	0.889 ± 0.079
	SVM	0.923 ± 0.054	0.929 ± 0.035	0.925 ± 0.048
	MLP	0.925 ± 0.046	0.931 ± 0.037	0.927 ± 0.043
VGG19	RF	0.811 ± 0.084	0.827 ± 0.102	0.813 ± 0.079
	SVM	0.847 ± 0.045	0.843 ± 0.062	0.849 ± 0.071
	MLP	0.862 ± 0.049	0.879 ± 0.059	0.872 ± 0.068
PDE	RF	0.693 ± 0.115	0.715 ± 0.108	0.681 ± 0.117
	SVM	0.724 ± 0.078	0.734 ± 0.070	0.717 ± 0.076
	MLP	0.736 ± 0.071	0.753 ± 0.062	0.748 ± 0.066
PDE + ResNet50	RF	0.927 ± 0.091	0.932 ± 0.083	0.924 ± 0.074
	SVM	0.981 ± 0.063	0.977 ± 0.072	0.981 ± 0.059
	MLP	0.984 ± 0.051	0.981 ± 0.067	0.985 ± 0.045
PDE+VGG19	RF	0.919 ± 0.101	0.920 ± 0.105	0.911 ± 0.112
	SVM	0.922 ± 0.062	0.935 ± 0.054	0.928 ± 0.071
	MLP	0.941 ± 0.061	0.955 ± 0.063	0.949 ± 0.069

To refrain from the possible effect of the small dataset, it is necessary to investigate the predicting performance of the classifier on different size training sets to make a trade-off between the cost of dataset size and the predicted effect. The images of days 1-5, 1-10, 1-15, 1-20, 1-25, 1-30, 1-35,1-40 and 1-45 were taken from the ResNet50+ raw dataset and used for training MLP and SVM, respectively. The evaluation has two steps. Firstly, the prediction of these classifiers was estimated on the set of remaining images corresponding to their training set (e.g., for classifiers trained on images of days 1-5, the prediction was performed on images of days 6-50, and so on). Secondly, all classifiers trained on different periods of days were used to predict the images of days 46-50 (Figure 4).

As shown in Figure 4, the Macro-F1 Scores of all the classifiers increase with the expansion of the sizes of training sets. When images of days 1-20 were used as the training set, models achieved relative high values of macro-F1 on all test sets with Macro-F1 Scores of 0.960 ± 0.049 for PDE+ResNet50+MLP to identify the individuals in images of the rest days and 0.960 ± 0.104 in images of days 46-50. When the size of the training set continues to enlarge, the curve of predicting effect goes steadily and changes slightly with the expansion of the training set. The highest Macro-F1 Score (0.983 ± 0.047) is achieved by PDE +ResNet50+MLP when using images of days 1-45 as the training set and images of days 46-50 as test sets. Furthermore, the Macro-F1 Score of PDE+ResNet50+MLP was higher than that of PDE+ResNet50+SVM in most sets of experiments except using images of 1-5 days as the training set.

3.5 Temporal tracking recognition of individuals on different time scales

In the actual scenario of breeding work, individuals need to be tracked continuously over a while. To investigate the tracking ability of the PDE+ResNet50 + MLP model, predicting the results of combination of the training set and test set for each individual on every day were extracted and summarized (Figure 5).

When trained on images of days 1-5, 1-10, and 1-15, the size of the training set was small and the model performed poorly on some individuals. For example, when trained on images of days 1-5, the model performed poorly on most of the individuals. As the size of the training set increased, these hard-to-predict individuals were gradually correctly identified by the model. When trained with images of days 1-30, there were few individuals that were difficult to identify, and for some individuals, the model could achieve a 100% recognition rate.

To understand how well each individual was tracked, we treated it as a traceable individual with an error rate of less than or equal to 10%. Then the predicting effects for all individuals were counted according to the above criterion (Table 2). When the size of the training set was small, the number of traceable individuals increased with the increase of the size of the training set. When images of days 1-25 were used as the training set, the number of traceable individuals was 45, accounting for 90% of the total individuals, and the number of individuals that could be identified at a 100% recognition rate was 27. When images of days 1-30 were used as the training set, the proportion of traceable individuals reached 98%, and the number of individuals that could be 100% identified was 33.

4 Discussion

The approach described in this paper using image processing analytical methods, which are widely used in studies on ecology and evolution (Bolger et al., 2012), has demonstrated its powerful application in studies on non-invasive tagging methods for *P. leopardus*. The PDE-based and CNN-based

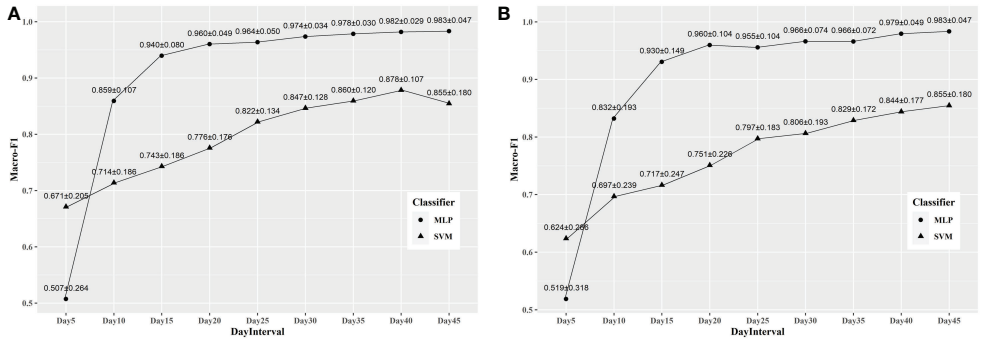


FIGURE 4
The results for prediction using classifiers trained by datasets with different size. **(A)** The results for prediction of the whole images of the rest days; **(B)** The results for prediction of the images of the 46th -50th days.

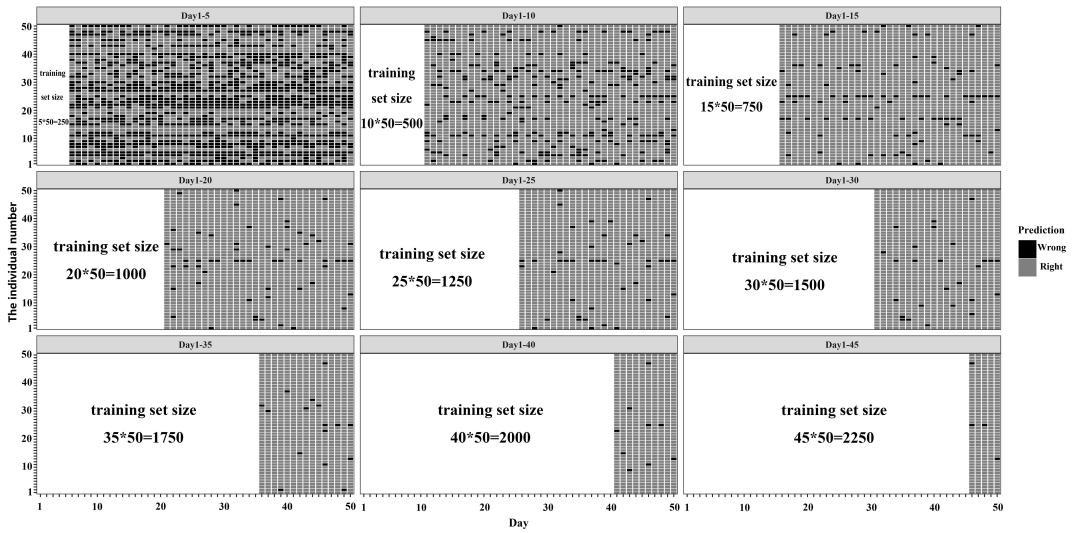


FIGURE 5
The result of tracking recognition of *P. leopardus* on different time scales.

image feature vector of each shape and texture structure, which is invariant against translation, rotation, scaling, and even modest distortion. As long as the feature pattern can be extracted from each image, the individuals can be effectively identified by using the RF, SVM and MLP classification of shape and texture features.

4.1 Advantages of deep convolutional neural networks in recognition of *P. leopardus*

To explore the feature extraction methods and machine learning models suitable for the recognition of *P. leopardus*,

TABLE 2 The statistics for results of tracking recognition on different time scales.

Dataset		Number of 100% identification	Number of an individual misclassified once	Number of individuals with error rate no more than 10%	Percent of trackable individuals
Training set	Test set				
1-5 days	6 - 50 days	6	1	10	20%
1-10 days	11-50 days	11	3	25	50%
1-15 days	16-50 days	21	7	36	72%
1 - 20 days	21-50 days	23	12	44	88%
1-25 days	26-50 days	27	15	45	90%
1-30 days	31-50 days	33	12	49	98%
1-35 days	36-50 days	38	9	47	94%
1-40 days	41-50 days	42	7	49	98%
1-45 days	46-50 days	47	2	47	94%

the PDE-based shape features, and CNN-based texture features were used for feature extraction, and then RF, SVM, and MLP were trained on the extracted features compared to the LeNet-5 model, an early convolutional neural network with fewer layers and a simple structure. The PDE + MLP obtained the best predictability with a Macro-F1 score of 0.748 ± 0.066 compared with PDE+RF and PDE+SVM on the raw dataset, while the ResNet50+MLP model achieved a Macro-F1 score of 0.927 ± 0.043, indicating that compared to the PDE-based image segmentation that had the relatively weak ability of feature extraction, ResNet50 extracted more details of features for individual imaged and achieved better recognition results. Various researchers are addressing the task of individual recognition in different way using traditional machine learning methods (Vaillant et al., 1994; Viola and Jones, 2001; Dollár et al., 2009) such as thresholding (Sivakumar and Murugesh, 2014), region growing (Gómez et al., 2007; Preetha et al., 2012), edge detection (Ma and Manjunath, 1997; Huang and Kuo, 2010; Wang et al., 2013), clustering (Celenk, 1990; Ali et al., 2006; Kavitha and Chellamuthu, 2010; Zheng et al., 2018), super-pixel (Li et al., 2012; Xie et al., 2019), etc. for years. PDE-based image multiscale decomposition belongs to edge detection method. Individual recognition research has also started to use the convolutional neural network (CNN) for better segmentation accuracy. That is why CNN is used successfully for individual recognition.

In this study, the CNN-based texture features included two categories: the features extracted by VGG19 and ResNet50. The VGG19 network has 16 layers of convolution layer (Simonyan and Zisserman, 2015), and the ResNet50 network has 49 layers of convolution layer (Savson et al., 2022). Among the three classifiers (RF, SVM, and MLP) trained with VGG19 features, VGG19 + MLP achieved the highest Macro-F1 score (0.872 ± 0.068), with an improvement of ~0.011 compared to LeNet-5 (0.861 ± 0.069). Our result is consistent with the conclusion in (He et al., 2016) that the accuracy of convolutional neural networks (CNNs) has been continuously improving. For example, the very deep VGG models, which have witnessed great success in a wide range of recognition tasks. In this study, when trained on a small dataset of 50 individuals, VGG19 or ResNet50 can better characterize the variability among individuals than LeNet-5 due to the deeper convolutional layers. Trained on the raw dataset, ResNet50+MLP achieved an improvement of ~0.055 compared to VGG19+MLP, indicating that the depth of the convolution layers in the ResNet50 network is enough for fully extracting the image features of *P. leopardus*. It is generally believed that by stacking multi-layer convolution kernels, the deep convolutional neural network allows the model to capture higher-dimensional and abstract features, including invisible high-frequency features that are traditionally considered noise (Krizhevsky et al., 2012). Thus, we purposed to use the ResNet50 to capture the patterns on the surface of *P. leopardus*.

Combined PDE-based and CNN-based features, PDE +ResNet50+MLP achieved the best prediction and PDE+ ResNet50+SVM got the suboptimal prediction, which are both better than those achieved by ResNet50+MLP and ResNet50 +SVM trained on the same dataset. These results indicated that when the size of the training set was small, the CNN had difficulty in capturing more details of the shape features of *P. leopardus*. The PDE-based features generated by PDE multiscale decomposition contained a series of segmentation results at varying image resolutions of shape pattern details at different levels. This process performed an iterative segmentation at an increasing image resolution in each step, and thus detected much smaller patterns of shape. It was exactly because the PDE-based features added more shape features for the CNN-based features to identify the individuals more effectively. This result also suggested that CNNs with some image segmentation methods may be more well-suited for individual recognition when the size of the dataset is small compared to just using CNNs.

4.2 Prediction at different time scales determine the optimal dataset size

In practical applications, due to the limited time available for collecting image data of the *P. leopardus*, it is usually hard for researchers to obtain enough data, so a trade-off between data volume and predicting effect is needed. Thus, the whole dataset was divided at different ratios to simulate the training set on different time scales, which were used as the training set to train the classifier and the remaining images as the test set for prediction. When using images of days 1-20 (i.e., 20 images per individual, 1000 images in total) as the training set, better results could be obtained (0.960 ± 0.049). Then the curve of the Macro-F1 changed slightly as the size of the training set increased. When trained on images of days 1-45, a remarkable improvement in predicting effect was obtained (0.983 ± 0.047). Since the test set was small, which only had images from days 46-50 when using images from days 1-45 as the training dataset, the model may have a higher recognition rate for some specific individuals coincidently.

After fixing the test set to images of 46-50 days, the predicting effect of the classifiers trained on a series of image subsets of 1-45 days, compared with the image set of 1-45 days. The results showed that the average Macro-F1 score increased with the increasing subset size for the models. It then plateaued when using images of days 1-20 for training and more selected days. The predicting effect slight increased training with images of days 1-40 and days 1-45, which may be a serendipitous result caused by the small test set. In addition, because the images faithfully reflect a continuous morphological change of *P. leopardus* over time, the images of days 1-45 were temporal continuity with the test set of days 46-50,

which might be another reason for the models to achieve the above best prediction.

4.3 Reliability of CNN-based recognition methods in long-term tracking

In the breeding work, breeders require individuals to be traceable for a long time using tagging methods, so it is necessary to ensure that the CNN-based method can achieve a comparatively high correction identification ratio of individuals for a continuous period. In our tracking experiments, we found that the performance of predicting effects showed large differences for some individuals. For example, the CNN-based method had a poor predicting effect on some individuals using small-size training sets, probably because the shape and texture features of these individuals were more similar to each other. If we expanded the training set, the model performed highly accurate recognition for these hard-to-identify individuals, showing that the CNN-based method needs large numbers of training images to obtain temporal-stable features for individual long-term tracking.

Most of the traditional tagging methods involve puncturing and destroying the body wall of *P. leopardus*, which can easily make them die due to wound ulceration. Meanwhile, the retention rate of the label fluctuates greatly due to the choice of the labeling tool, the experimental individual, and the operation methods. Generally speaking, the retention rate for one month is between 50% and 80%. The above two types of problems make it difficult to apply traditional tagging methods to the tagging work of aquatic animal breeding (Jepsen et al., 2015). Our method can also save time and cost less in comparison with molecular methods for the individual tracking, especially in a large population. For 100 individual samples, it would take approximately 14 days for good identification with the traditional molecular methods (Wang, 2016). In addition, these methods are generally laborious and time-consuming and sometimes require invasive operations that need a relatively large amount of sample materials, which would require the sacrifice of animals under study to ensure a sufficient amount of DNA for individual recognition (Mao et al., 2013). However, our method can achieve a high-throughput operation with aid of an ordinary digital camera, and even mobile phones and can reduce the workload to just less than 1 hrs. Therefore, we would propose that the use of CNN-based image recognition method has a great applying potential in the tagging work for *P. leopardus*.

4.4 Possible improving directions of model

In this study, the CNNs were trained on images of 50 days, which were randomly selected in the period. The sample size was relatively sufficient for training. However, in the actual breeding work, there are often more individuals. It is necessary to increase individuals in the subsequent study to explore the upper limit of the individuals that can be classified by the CNN method to meet the actual needs. Fortunately, many multiclassification models are now available, and perform well. Although the CNN approach outlined above has great potential, there are several outstanding challenges with applying CNNs to a wider spectrum of problems. One important obstacle is the large amount of training data required by CNNs. This challenge includes both the generation of large labeled training examples and time- and memory-efficient training with these large examples given limited computational resources. Fortunately, continued improvements in simulation speed and the efficiency of CNN training (Chilimbi et al., 2014; Urs et al., 2017) are mitigating this problem.

Another challenge with the application of CNNs is that their performance can be sensitive to network architecture (Szegedy et al., 2015). There is no underlying theory for selecting optimal network architecture, though improved architectures are sure to continue to arise, and automated methods exist for optimizing the many hyperparameters of a given architecture (Snoek et al., 2012). Though we uncover some promising CNN architectures for the recognition of *P. leopardus*, we suspect that substantial improvements can still be made. Meanwhile, length calibrators (e.g., rulers) can be added to the field of view for photograph, so that the difference in relative size among individuals can be involved in the dataset, which may improve the performance of model in the temporal tracking task. Furthermore, if more lightweight network architectures such as MobileNets (Li et al., 2012) are used, it is promising to deploy the recognition systems on mobile device as applications to enable mobile and real-time recognition of *P. leopardus*.

5 Conclusion

In this study, a dataset involving images of 50 P. *leopardus* individuals was obtained by continuous photography in 50 consecutive days. Then we performed prediction using different classifiers with different feature extraction methods and compare the predicting effect on the dataset. The results shows that the feature extraction method based on deep CNN model ResNet50 with PDE-based multiscale decomposition segmentation method performed well in the recognition task of *P. leopardus*. The prediction results on training sets of different sizes show that the model achieves satisfactory prediction results when the number of images per individuals in training set reaches 20. Temporal tracking recognition experiments on different time scales showed that the deep CNN model ResNet50 with PDE-based segmentation method can recognize individuals over a longer time span with better

accuracy than other invasive tagging methods. The results of this study will provide an important reference for the development of non-invasive tagging methods based on deep learning and the characterization of complex traits of *P. leopardus*. In the future, we will increase the population to further verify our conclusion.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Ethics statement

The animal study was reviewed and approved by Institutional Animal Care and Use Committee of Ocean University of China.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work, and approved it for publication.

Funding

We acknowledged the National Key Research and Development Program of China (2022YFD2400501), the Key R&D Project of Hainan Province (ZDYF2021XDNY133), National Natural Science Foundation of China (32072976), Sanya Yazhou Bay Science and Technology City (SKJCKJ-2019KY01), and the China Postdoctoral Science Foundation (2021703030).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The handling editor HZ declared a shared affiliation with the authors at the time of review.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ali, M. A., Dooley, L. S., and Karmakar, G. C. (2006). "Object based image segmentation using fuzzy clustering," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse, France: IEEE. doi: 10.1109/ICASSP.2006.1660290

Bolger, D. T., Morrison, T. A., Vance, B., Lee, D., and Farid, H. (2012). A computer-assisted system for photographic mark–recapture analysis. *Methods Ecol. Evol.* 3 (5), 813–822. doi: 10.1111/j.2041-210X.2012.00212.x

Celenk, M. (1990). A color clustering technique for image segmentation. *Comput. Vis. Graph. Image Process* 52 (2), 145–170. doi: 10.1016/0734-189X(90)90052-W

Chaudhuri, S., Chatterjee, S., Katz, N., Nelson, M., and Goldbaum, M. (1989). Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE. Trans. Med. Imaging* 8 (3), 263–269. doi: 10.1109/42.34715

Chilimbi, T., Suzue, Y., Apacible, J., and Kalyanaraman, K. (2014). "Project adam: Building an efficient and scalable deep learning training system," in *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)* (Berkeley, CA, USA: USENIX Association) 571–582. doi: 10.1108/01439911111122716

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20 (3), 273–297. doi: 10.1007/BF00994018

Døving, K., Westerberg, H., and Johnsen, P. (2011). Role of olfaction in the behavioral and neuronal responses of Atlantic salmon, *Salmo salar*, to hydrographic stratification. *Can. J. Fish. Aquat. Sci.* 42, 1658–1667. doi: 10.1139/f85-207

Dollár, P., Tu, Z., Perona, P., and Belongie, S. (2009). "Integral channel features," in *British Machine Vision Conference*. (London, UK:BMVC Press) 91.1–91.11. doi: 10.5244/C.23.91

Fearnbach, H., Durban, J., Parsons, K., and Claridge, D. (2012). Photographic mark-recapture analysis of local dynamics within an open population of dolphins. *Ecol. Appl.* 22 (5), 1689–1700. doi: 10.1890/12-0021.1

Forcada, J., and Aguilar, A. (2000). Use of photographic identification in capture-recapture studies of mediterranean monk seals. *Mar. Mammal. Sci.* 16 (4), 767–793. doi: 10.1111/j.1748-7692.2000.tb00971.x

Gómez, O., González, J. A., and Morales, E. F. (2007). "Image segmentation using automatic seeded region growing and instance-based learning," in *Progress in pattern recognition, image analysis and applications*. Eds. L. Rueda, D. Mery and J. Kittler (Berlin Heidelberg: Springer), 192–201. doi: 10.1007/978-3-540-76725-1_21

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE. 770–778. doi: 10.1109/CVPR.2016.90

Hinch, S. G., Standen, E. M., Healey, M. C., and Farrell, A. P. (2002). Swimming patterns and behaviour of upriver-migrating adult pink (*Oncorhynchus gorbuscha*) and sockeye (*O. nerka*) salmon as assessed by EMG telemetry in the Fraser river, British Columbia, Canada. *Hydrobiologia* 483 (1), 147–160. doi: 10.1023/A:1021327511881

Huang, Y. R., and Kuo, C. M. (2010). "Image segmentation using edge detection and region distribution," in *2010 3rd International Congress on Image and Signal Processing*, Yantai, China: IEEE. 1410–1414. doi: 10.1109/CISP.2010.5646352

Jepsen, N., Thorstad, E. B., Havn, T., and Lucas, M. C. (2015). The use of external electronic tags on fish: an evaluation of tag retention and tagging effects. *Anim. Biotelemetry.* 3 (1), 49. doi: 10.1186/s40317-015-0086-z

Kamilaris, A., and Prenafeta-Boldú, F. X. (2018). A review of the use of convolutional neural networks in agriculture. *J. Agric. Sci.* 156 (3), 312–322. doi: 10.1017/S0021859618000436

Karanth, U., and Nichols, J. D. (1998). Estimation of tiger densities in India using photographic captures and recaptures. *Ecology* 79 (8), 2852–2862. doi: 10.1890/0012-9658(1998)079[2852:EOTDII]2.0.CO;2

Kavitha, A. R., and Chellamuthu, C. (2010). Implementation of gray-level clustering algorithm for image segmentation. *Procedia. Comput. Sci.* 2, 314–320. doi: 10.1016/j.procs.2010.11.041

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "ImageNet classification with deep convolutional neural networks," in *NIPS'12: Proceedings of the 25th International Conference on Neural Information Processing Systems*, New York, United States: Curran Associates Inc. 1097–1105. doi: 10.1145/3065386

Langtimm, C. A., Beck, C. A., Edwards, H. H., Fick-Child, K. J., Ackerman, B. B., Barton, S. L., et al. (2004). Survival estimates for Florida manatees from the photo-identification of individuals. *Mar. Mammal. Sci.* 20 (3), 438–463. doi: 10.1111/j.1748-7692.2004.tb01171.x

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521 (7553), 436–444. doi: 10.1038/nature14539

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. IEEE.* 86 (11), 2278–2324. doi: 10.1109/5.726791

Linderman, G. C., Rachh, M., Hoskins, J. G., Steinerberger, S., and Kluger, Y. (2019). Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data. *Nat. Methods* 16 (3), 243–245. doi: 10.1038/s41592-018-0308-4

Li, Z., Wu, X. M., and Chang, S. F. (2012). "Segmentation using superpixels: A bipartite graph partitioning approach," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, USA: IEEE. doi: 10.1109/CVPR.2012.6247750

Ma, W. Y., and Manjunath, B. S. (1997). "Edge flow: A framework of boundary detection and image segmentation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, PR, USA: IEEE. 744–749. doi: 10.1109/CVPR.1997.609409

Mao, J., Lv, J., Miao, Y., Sun, C., Hu, L., Zhang, R., et al. (2013). Development of a rapid and efficient method for non-lethal DNA sampling and genotyping in scallops. *PLoS. One* 8 (7), e68096. doi: 10.1371/journal.pone.0068096

Ogura, M., and Ishida, Y. (1995). Homing behavior and vertical movements of four species of pacific salmon (*Oncorhynchus* spp.) in the central Bering Sea. *Can. J. Fish. Aquat. Sci.* 52 (3), 532–540. doi: 10.1139/f95-054

Preetha, M. M. S. J., Suresh, L. P., and Bosco, M. J. (2012). "Image segmentation using seeded region growing"," in *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, Nagercoil, India: IEEE. 576–583. doi: 10.1109/ICCEET.2012.6203897

Quinn, T. P., Terhart, B. A., and Groot, C. (1989). Migratory orientation and vertical movements of homing adult sockeye salmon, *Oncorhynchus nerka*, in coastal waters. *Anim. Behav.* 37, 587–599. doi: 10.1016/0003-3472(89)90038-9

Raab, T., Madhav, M. S., Jayakumar, R. P., Henninger, J., Cowan, N. J., and Benda, J. (2022). Advances in non-invasive tracking of wave-type electric fish in natural and laboratory settings. *Front. Integr. Neurosci.* 16. doi: 10.3389/fnint.2022.965211

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE. 779–788. doi: 10.1109/CVPR.2016.91

Reed, J. Z., Tollit, D. J., Thompson, P. M., and Amos, W. (1997). Molecular scatology: the use of molecular genetic analysis to assign species, sex and individual identity to seal faeces. *Mol. Ecol.* 6 (3), 225–234. doi: 10.1046/j.1365-294x.1997.00175.x

Rimmer, M. A., and Glamuzina, B. (2019). A review of grouper (Family serranidae: Subfamily epinephelinae) aquaculture from a sustainability science perspective. *Rev. Aquac.* 11 (1), 58–87. doi: 10.1111/raq.12226

Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Phys. D.* 60 (1), 259–268. doi: 10.1016/0167-2789(92)90242-F

Savson, D. J., Zenilman, S. S., Smith, C. R., Daugherity, E. K., Singh, B., and Getchell, R. G. (2022). Comparison of alfaxalone and tricaine methanesulfonate immersion anesthesia and alfaxalone residue clearance in rainbow trout (*Oncorhynchus mykiss*). *Comp. Med.* 72 (3), 181–194. doi: 10.30802/aalas-cm-22-000052

Shi, L., Ye, S. W., Zhu, H., Ji, X., Wang, J. C., Liu, C. A., et al. (2022). The spatial-temporal distribution of fish in lake using acoustic tagging and tracking method. *Acta Hydrobiol. Sin.* 46 (5), 611–620. doi: 10.7541/2022.2021.004

Simonyan, K., and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, arXiv:1409.1556. doi: 10.48550/arXiv.1409.1556

Sivakumar, V., and Murugesh, V. (2014). "A brief study of image segmentation using thresholding technique on a noisy image"," in *International Conference on Information Communication and Embedded Systems (ICICES2014)*, Chennai, India: IEEE. 1–6. doi: 10.1109/ICICES.2014.7034056

Šmejkal, M., Bartoň, D., Děd, V., Souza, A. T., Blabolil, P., Vejřík, L., et al. (2020). Negative feedback concept in tagging: Ghost tags imperil the long-term monitoring of fishes. *PLoS. One* 15 (3), e0229350. doi: 10.1371/journal.pone.0229350

Snoek, J., Larochelle, H., and Adams, R. P. (2012). "Practical Bayesian optimization of machine learning algorithms," in *NIPS'12: Proceedings of the 25th International Conference on Neural Information Processing Systems*, New York, United States: Curran Associates Inc. 2951–2959. doi: 10.48550/arXiv.1206.2944

Stateczny, A., Uday Kiran, G., Bindu, G., Ravi Chythanya, K., and Ayyappa Swamy, K. (2022). Spiral search grasshopper features selection with VGG19-ResNet50 for remote sensing object detection. *Remote. Sens.* 14(21), 5398. doi: 10.3390/rs14215398

Sulak, K. J., Randall, M. T., Edwards, R. E., Summers, T. M., Luke, K. E., Smith, W. T., et al. (2009). Defining winter trophic habitat of juvenile gulf sturgeon in the suwannee and Apalachicola rivermouth estuaries, acoustic telemetry investigations. *J. Appl. Ichthyol.* 25 (5), 505–515. doi: 10.1111/j.1439-0426.2009.01333.x

Szegedy, C., Wei, L., Yangqing, J., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions *in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (Boston, MA, USA: IEEE). 1–9. doi: 10.1109/CVPR.2015.7298594

Urs, K., Webb, T. J., Wang, X., Nassar, M., Arjun, K., Bansal,, et al. (2017). "Flexpoint: an adaptive numerical format for efficient training of deep neural networks," in *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, New York, United States: Curran Associates Inc. doi: 10.48550/arXiv.1711.02213

Vaillant, R., Monrocq, C., and Lecun, Y. (1994). Original approach for the localization of objects in images. *IEE. P-VIS. Image. Sign.* 141 (4), 245–250. doi: 10.1049/ip-vis:19941301

Viola, P., and Jones, M. (2001). "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, HI, USA: IEEE. I–I. doi: 10.1109/CVPR.2001.990517

Wang, J. (2016). Individual identification from genetic marker data: developments and accuracy comparisons of methods. *Mol. Ecol. Resour.* 16 (1), 163–175. doi: 10.1111/1755-0998.12452

Wang, Y., Ji, G., Lin, P., and Trucco, E. (2013). Retinal vessel segmentation using multiwavelet kernels and multiscale hierarchical decomposition. *Pattern. Recognit.* 46 (8), 2117–2133. doi: 10.1016/j.patcog.2012.12.014

Watanabe, Y. Y., Goldman, K. J., Caselle, J. E., Chapman, D. D., and Papastamatiou, Y. P. (2015). Comparative analyses of animal-tracking data reveal ecological significance of endothermy in fishes. *Proc. Natl. Acad. Sci.* 112 (19), 6104–6109. doi: 10.1073/pnas.1500316112

Welch, D. W., Ward, B. R., and Batten, S. D. (2004). Early ocean survival and marine movements of hatchery and wild steelhead trout (*Oncorhynchus mykiss*) determined by an acoustic array: Queen Charlotte strait, British Columbia. *Deep. Sea. Res. Part II.* 51 (6), 897–909. doi: 10.1016/j.dsr2.2004.05.010

Williams, B. K., Nichols, J. D., and Conroy, M. J. (2002). *Analysis and management of animal populations: modeling, estimation and decision making* (San Diego, CA: Academic Press).

Xia, S., Sun, J., Li, M., Zhao, W., Zhang, D., You, H., et al. (2020). Influence of dietary protein level on growth performance, digestibility and activity of immunity-related enzymes of leopard coral grouper, *Plectropomus leopardus* (Lacépède 1802). *Aquacult. Nutr.* 26 (2), 242–247. doi: 10.1111/anu.12985

Xie, X., Xie, G., Xu, X., Cui, L., and Ren, J. (2019). Automatic image segmentation with superpixels and image-level labels. *IEEE. Access.* 7, 10999–11009. doi: 10.1109/ACCESS.2019.2891941

Yang, Y., Wu, L. N., Chen, J. F., Wu, X., Xia, J. H., Meng, Z. N., et al. (2020). Whole-genome sequencing of leopard coral grouper (*Plectropomus leopardus*) and exploration of regulation mechanism of skin color and adaptive evolution. *Zool. Res.* 41 (3), 328–340. doi: 10.24272/j.issn.2095-8137.2020.038

Yano, A., Ogura, M., Sato, A., Sakaki, Y., Ban, M., and Nagasawa, K. (1996). Development of ultrasonic telemetry technique for investigating the magnetic of salmonids. *Fish. Sci.* 62 (5), 698–704. doi: 10.2331/fishsci.62.698

Yoshua, B. (2011). "Deep learning of resentations for unsupervised and transfer learning," in *UTLW'11: Proceedings of the 2011 International Conference on Unsupervised and Transfer Learning workshop*, (Washington, USA: JMLR.org) 17–37.

Zheng, X., Lei, Q., Yao, R., Gong, Y., and Yin, Q. (2018). Image segmentation based on adaptive K-means algorithm. *Eurasip. J. Image. Video. Process* 2018 (1), 68. doi: 10.1186/s13640-018-0309-3

Zhou, L., Zheng, X., Yang, D., Wang, Y., Bai, X., and Ye, X. (2021). Application of multi-label classification models for the diagnosis of diabetic complications. *BMC. Med. Inform. Decis. Mak.* 21 (1), 182. doi: 10.1186/s12911-021-01525-7

Zhuang, X., Qu, M., Zhang, X., and Ding, S. (2013). A comprehensive description and evolutionary analysis of 22 grouper (perciformes, epinephelidae) mitochondrial genomes with emphasis on two novel genome organizations. *PLoS. One* 8 (8), e73561. doi: 10.1371/journal.pone.0073561

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 23 January 2023
DOI 10.3389/fmars.2022.1058401

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Carlos Pérez-Collazo,
University of Vigo, Spain
Ning Wang,
Dalian Maritime University, China

*CORRESPONDENCE
Jiucai Jin
jinjiucai@fio.org.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 30 September 2022
ACCEPTED 28 November 2022
PUBLISHED 23 January 2023

CITATION
Zhang J, Jin J, Ma Y and Ren P (2023) Lightweight object detection algorithm based on YOLOv5 for unmanned surface vehicles.
Front. Mar. Sci. 9:1058401.
doi: 10.3389/fmars.2022.1058401

COPYRIGHT
© 2023 Zhang, Jin, Ma and Ren. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Lightweight object detection algorithm based on YOLOv5 for unmanned surface vehicles

Jialin Zhang[1,2], Jiucai Jin[1*], Yi Ma[1] and Peng Ren[2]

[1]First Institute of Oceanography, Ministry of Natural Resources, Qingdao, China, [2]College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao, China

Visual detection technology is essential for an unmanned surface vehicle (USV) to perceive the surrounding environment; it can determine the spatial position and category of the object, which provides important environmental information for path planning and collision prevention of the USV. During a close-in reconnaissance mission, it is necessary for a USV to swiftly navigate in a complex maritime environment. Therefore, an object detection algorithm used in USVs should have high detection s peed and accuracy. In this paper, a YOLOv5 lightweight object detection algorithm using a Ghost module and Transformer is proposed for USVs. Firstly, in the backbone network, the original convolution operation in YOLOv5 is upgraded by convolution stacking with depth-wise convolution in the Ghost module. Secondly, to exalt feature extraction without deepening the network depth, we propose integrating the Transformer at the end of the backbone network and Feature Pyramid Network structure in the YOLOv5, which can improve the ability of feature expression. Lastly, the proposed algorithm and six other deep learning algorithms were tested on ship datasets. The results show that the average accuracy of the proposed algorithm is higher than that of the other six algorithms. In particular, in comparison with the original YOLOv5 model, the model size of the proposed algorithm is reduced to 12.24 M, the frames per second reached 138, the detection accuracy was improved by 1.3%, and the mean of average precision (0.5) reached 96.6% (from 95.3%). In the verification experiment, the proposed algorithm was tested on the ship video collected by the "JiuHang 750" USV under different marine environments. The test results show that the proposed algorithm has a significantly improved detection accuracy compared with other lightweight detection algorithms.

KEYWORDS

object detection, USV, ghost model, lightweight, YOLO

1 Introduction

In recent years, unmanned surface vehicle (USV) technology has developed rapidly, and USVs are widely used in maritime safety tasks, such as orderly and complex patrols, reconnaissance, and detection and tracking of specific objects. Traditional ship detection and tracking systems typically employ radar or AIS (Vesecky et al., 2009; Dzvonkovskaya and Rohling, 2010; Vesecky et al., 2010; Sermi et al., 2013). However, the radar has a relatively long scanning period and slow detection speed. It cannot distinguish between specific types of objects, and hence false and missed detections easily occur. Information collected by AIS can be intentionally turned off by ships, which sometimes results in AIS unreliability. The existing ship detection methods are based on vision; they not only have a long detection range but also have high resolution and object detailing. The traditional detection methods based on vision are mainly Mean-shift (Liu et al., 2013) and HOG-SVM (Xu and Liu, 2016). Their characteristic is that they mainly rely on a single shallow feature to complete the ship detection task. However, these features are easily affected by the ship's appearance, shape, and complex environment, resulting in poor robustness. With the rapid development of the visual field, visual object detection based on deep learning has become a popular research topic. Object detection algorithms based on deep learning have broad application prospects in the marine environment (Chen et al., 2021; Wang et al., 2022); nevertheless, their applications have not been fully valued until now (Mittal et al., 2022). For example, object detection can be used to perceive the surrounding environment. The object's orientation and image information plays an important role in path planning, collision avoidance, and object monitoring of a USV. At present, an object detection algorithm based on deep learning can more accurately classify and detect object positions. However, it has high requirements for the vision-based processing system of the USV; moreover, speed and accuracy of the object detection algorithm are also major challenges.

In this study, we propose a lightweight object detection network based on the You-Only-Look-Once-v5 (YOLOv5) to obtain fast detection speed and high accuracy for USVs. The object detection performance in a complex environment has been improved. The proposed network has reduced detection time and improvements in terms of anchor boxes, backbone, and feature pyramid network (FPN) structure. We obtained a set of anchor boxes through the K-means clustering method to adopt to the ship's characteristics. The Ghost module upgraded the convolution (Conv) in the backbone to reduce the network detection time. The Transformer is integrated into the cross stage partial network (CSPNet) of the backbone and FPN structure to achieve more useful feature extraction. The proposed network is composed of these simple but effective modules, thus balancing detection speed and accuracy well. Figure 1 shows the detailed flowchart of our training model. Lastly, the experimental results demonstrate its excellent performance on the task of detecting ship objects.

The contributions of this study include the following:

- We obtain a new set of anchor boxes to adapt to the structural characteristics; i.e., the width of the ship is longer than the height used by the K-means clustering algorithm on the ship dataset.
- A combination of Conv stacking with depth-wise Conv in the Ghost module was adopted to structure the backbone feature extraction in YOLOv5. In comparison with the original Conv, the Ghost module has better computing efficiency, which not only reduces the model training and detection times but also improves accuracy.
- We integrated the Transformer into the end of the backbone and FPN structure in the YOLOv5 network, which can improve the feature expression ability and enhance the detection accuracy without deepening the network depth.
- The proposed algorithm has achieved a good balance between detection accuracy and speed. In the actual marine environment testing process, our algorithm obtains a high accuracy rate and is found to be robust in the sea fog environment.

The remainder of this paper is organized as follows. In *Section 2*, we show the data augmentation and related work. We describe our approach in *Section 3*. The experimental results performance and discussion are presented in *Section 4*. In *Section 5*, we summarize this work.

2 Related work

2.1 Data augmentation

The purpose of data augmentation is to generate more training samples based on existing datasets. The method of data augmentation is to randomly transform the local or global features of the images, and its role is to improve the robustness and generalization ability of our trained model. In certain special circumstances, highlighting, blurring, and occlusion were encountered in the future detection process of our model. Therefore, the hue, saturation, and value have been adjusted in the model training process. With regard to the geometric distortion of the image, certain operations are performed, i.e., rotation, horizontal and vertical translation, scaling, and shearing of the image. In addition, there are some special data enhancement methods, such as Mixup (Zhang et al., 2017) and Mosaic (Bochkovskiy et al., 2020). In the Mixup data

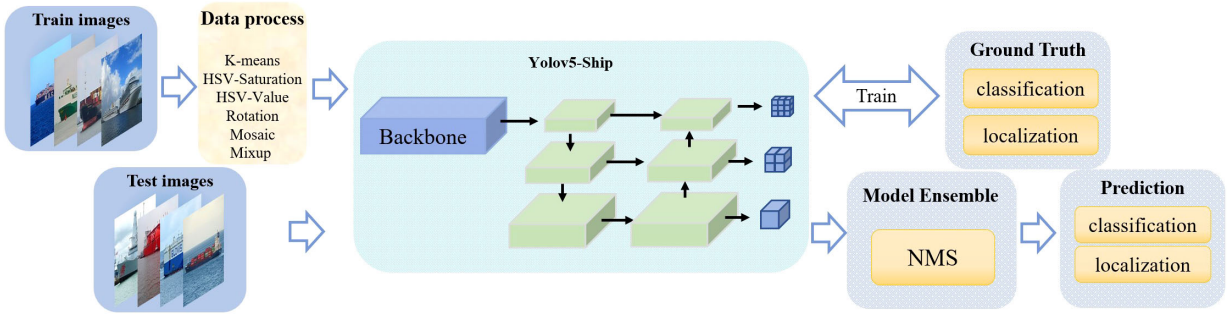


FIGURE 1
Detailed flowchart of the proposed model training.

enhancement method, new sample-label data are generated by adding two image sample-label data pairs in proportion. In the Mosaic data enhancement method, a new picture is generated using four pictures through random reduction, cropping, and arrangement. In this paper, we used a combination of Mixup, Mosaic, and traditional data augmentation methods.

2.2 Visual object detection based on deep learning

In recent years, visual detection technology has made great progress, particularly detection methods that are based on deep learning. The deep learning-based object detection algorithms are mainly divided into two types—two-stage and one-stage. The first step of a two-stage object detection algorithm is to generate a position box by generating a region proposal that can extract features; then, the second step is to perform category prediction. It has high accuracy but slow speed; thus, it is not suitable for real-time object detection like Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2015). A one-stage object detection algorithm performs classification and bounding box regression while generating candidate boxes and has fast speed but less accuracy; hence, it is suitable for real-time object detection like SSD (Liu et al., 2016) and YOLOv3 (Redmon and Farhadi, 2018). High object detection speed is essential for a USV platform; therefore, one-stage object detection algorithms are more suitable.

In the case of maritime object detection, many scholars have investigated from sea–skyline detection to ship detection. Bai et al. (2021) proposed a sea–skyline detection method based on local Otsu segmentation and Hough transform. Later, the monopole object detection method was introduced for ship detection, which reduces a certain amount of interference and calculations, and it optimizes the accuracy and speed of ship detection. Chen et al. (2021) proposed an integrated ship detection framework based on an image segmentation method for edge detection. The Canny edge detector and Gaussian filter are used to detect the edges of ships in the image, suppress the edges related to the background, and, finally, connect them to form the outline of the ship; the method achieved an effect of 32 fps. In ship detection methods based on deep learning, Gupta et al. (2021) proposed a classification method for ship detection based on support vector machines (SVMs) and convolutional neural networks (CNNs). First, the feature package is used to deal with diverse features of different types of ships, and then the CNN is used for feature extraction. Finally, 2,700 images are used for training, and the accuracy rate of their model reaches 91.04%. Zou et al. (2019) improved a maritime object detection method based on Faster R-CNN. The ResNet-50 network is replaced by the VGG16 network. The results show that the recognition and detection effect of small ships was significantly improved. Zou et al. (2020) proposed an improved SSD algorithm based on the MobileNetV2 CNN that is used in ship detection and identification. The results show that the SSD_MobileNetV2 algorithm has better performance for ship images. Shi and Suo (2018) proposed a ship detection algorithm based on an improved visual attention model. Firstly, the wavelet transform (WT) is used for feature extraction; secondly, the improved Gabor filter and deep multifaceted transformers (DMT) algorithm are used to obtain the directional and edge texture features of the image. The final test demonstrated high detection accuracy and good real-time performance. For the existing ship detection algorithms based on deep learning, it is difficult to simultaneously obtain good detection accuracy and real-time performance.

2.3 Ship detection based on YOLO

Since the YOLO algorithm was published, it has been widely studied because of its good computational efficiency and detection accuracy. Lee et al. (2018) applied the YOLOv2 algorithm to ship detection and classification. In comparison with other machine learning algorithms, their model has better robustness and scalability. Li and Qiao (2021) proposed a ship

detection and tracking algorithm based on YOLOv3. They used a graph matching algorithm and Kalman filter to achieve object matching and tracking, which solves the problems of object occlusion and label switching. Jie et al. (2021) improved YOLOv3 for ship detection and tracking in inland waterways; the K-means clustering algorithm was used to improve the anchor boxes, and it was improved by taking the single softmax classifier and introducing the Soft-NMS algorithm. Their algorithm could enhance the safety of inland navigation and prevent collisions and accidents. Zhang et al. (2020) improved a maritime object detection algorithm based on YOLOv3. They proposed an E-CIoU loss function for bounding box regression, and the improved method accelerated the convergence speed and improved the detection accuracy. Liu and Li, (2021) studied ship statistics in waterway videos. To realize automatic detection and tracking by YOLOv3, they designed a self-correcting network combining regression-based direction judgment and object counting method with variable time window. The results show that their algorithm can achieve automatic analysis and statistical data extraction in waterways videos. Sun et al. (2021) optimized the backbone network CSPDarkNet of YOLOv4 for application in an auxiliary intelligent ship navigation system. They added a receptive field block module, and the FPN of YOLOv4 was improved by combining the Transformer mechanism. Their algorithm improves the inference speed and detection accuracy. Liu et al. (2021) improved the USV maritime environment perception ability using an improved YOLOv4 object detection algorithm. The reverse depth-wise separable convolution (RDSC) was applied to the backbone and FPN structures of YOLOv4, which reduced the number of parameters of the network and improved the accuracy by 1.78% compared with the original model. Thus, the algorithm has a small network size and better performance in terms of detection speed.

In summary, the ship detection methods are mostly difficult to apply on USVs because of limited computing resources and detection speed. Thus far, the problems of accuracy and speed of maritime object detection have not been resolved. In comparison with traditional object detection algorithms, the deep learning-based object detection algorithm has good accuracy rate, but slow detection speed. Therefore, this study focuses on improving an object detection algorithm based on YOLOv5 to solve the problems of real-time performance and accuracy of the maritime ship detection algorithm applied to the USV platform.

3 Methods

The maritime object detection includes two tasks, i.e., classification and positioning of ships. A robust object detection algorithm should not only consider the detection speed, but also consider the complex environmental scenarios. In the field of object detection, the YOLO object detection algorithm performs well in various environments, such as changes in illumination in a complex sea environment, and recognition of distant small targets in the sea. The fifth version YOLO object detection algorithm has been developed, and its efficiency is very good.

YOLOv5 has high performance in terms of detection speed and accuracy. According to the depth and width of the network, it is divided into four versions: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The basic network of the four versions is similar. The structure of YOLOv5 is mainly composed of the input, backbone, Neck, and Prediction. At the input, we perform data augmentation operations, such as Mixup and Mosaic, which can enrich the ship dataset and improve the detection efficiency of small objects. Feature maps of different scales are extracted at the backbone network. The FPN and path aggregation network (PANet) at the Neck strengthen the feature fusion ability. The FPN transfers high-level semantic features in a top-down manner, and the PANet transfers low-level strong localization features in a bottom-up manner after the FPN. The final output is the prediction of the network, and the prediction uses the non-maximum suppression (NMS) algorithm to filter the object boxes. Then, we make predictions on the image features, generate bounding boxes and predict classes.

In this study, we examine the ability of the USV to detect and classify an object quickly. We used YOLOv5 as the base network and improved it. The architecture of the improved YOLOv5 is shown in Figure 2.

3.1 Anchor box calculation

In object detection tasks, choosing suitable anchor boxes can significantly improve the speed and accuracy of object detection. Anchor boxes are boxes presented by a fixed aspect ratio in YOLO, which is used to predict the category and position offset of the bounding box. The default anchor boxes are generated in the MS COCO and VOC datasets. The COCO and VOC datasets have 80 and 20 classes, respectively, but ships are only one of their classes. Therefore, the default anchor boxes are not fully applicable to the objects in the ship dataset. To adapt the structural characteristics of the width of the ship being longer than the height of the ship, we used the K-means clustering algorithm on the ship dataset to obtain a set of anchor boxes. The clustering results for the ship dataset labels are shown in Figure 3. The steps to implement the Algorithm 1 are described as follows.

Input:
A ground truth label dataset: S ={x_1, x_2, x_3 … x_m}
The number of cluster centers: k
Output:
A group of anchor boxes: {c_1, c_2, c_3 … c_k}

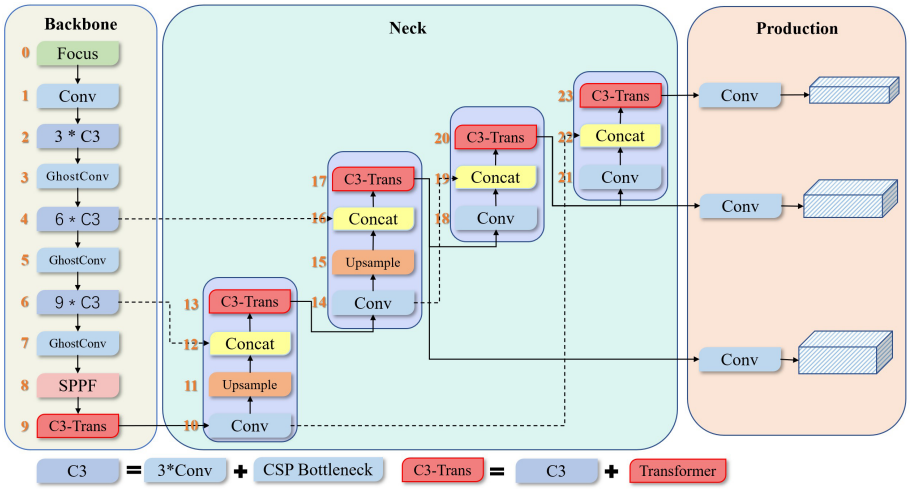


FIGURE 2
Improved YOLOv5 network structure proposed in this paper.

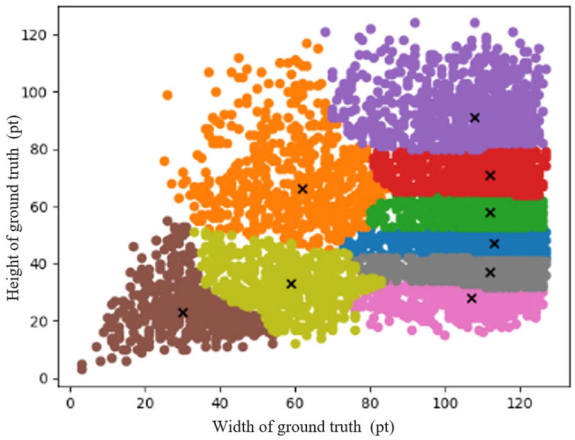


FIGURE 3
Result of ship dataset using K-means clustering. The *x*-coordinate is the width of the ground truth bounding box and the *y*-coordinate is the height of the ground truth bounding box.

Procedure:
First, select randomly nine boxes of ground truth labels from the ship dataset as the cluster centers;
for $i = 1, 2 \ldots k$ **do**
REPEAT
for $j = 1, 2, 3 \ldots m$ **do**
Calculate the distance between x_j and each cluster center $\{c_1, c_2, c_3 \ldots c_k\}$ $d_{ji} = ||x_i - c_i||_2$;
Return each label x_j to cluster centers c_i with the closest distance; Update the cluster center c_i for each class in each cluster $c_i = \Sigma_{x \in c_i} \frac{x_i}{|c_i|}$;
end for
UNTIL Cluster centers no longer change.

ALGORITHM 1
Pseudocode of K-means clustering algorithm for anchor boxes.

Finally, nine sets of adaptive anchor boxes are generated using the K-means clustering algorithm, i.e., (29,23), (58,31), (109,30), (62,60), (112,39), (114,50), (78,89), (112,65), and (112, 87). The anchor boxes of the clustering algorithm can effectively accelerate the convergence speed of the network and effectively improve the gradient descent problem in the training process.

3.2 Ghost model

There are limitations regarding the memory and computing resources of embedded industrial computers in USVs; therefore, the key to ship detection on an USV is to find a lightweight detection model that can balance detection accuracy and computational complexity. CNNs are usually composed of many convolution kernel operations, which will result in large computational cost. During model training, many redundant feature maps will be generated, as shown in Figure 4. Redundant feature maps not only have high similarity but also greatly increase computational complexity. To reduce the computational load of the model and raise the detection speed, an efficient architecture and high-performance GhostNet (Han et al., 2020) structure are adopted.

The detailed structure of the Conv and Ghost model is shown in Figure 5. Figure 5A shows the Conv operaton. A given input is

defined as $X \in \mathbb{R}^{c \times h \times w}$, where c is the number of channels of the input; h and w are the height and width of the input data, respectively. The n feature maps are generated through ordinary convolution that can be expressed as $Y = X * f + b$where $Y \in \mathbb{R}^{c \times k \times k \times n}$ is the output feature map with n channels, and * is the convolution operation; f denotes the convolution filter of this layer, b is the bias term, and $k \times k$ is the size of the convolution kernel f. The value of the floating point of operations (FLOPs) can be expressed as $n \cdot h \cdot w \cdot c \cdot k \cdot k$. Owing to the large values of n and c, the usual parameters of the model are very large. The Ghost model comprises Conv and depth-wise Conv with less parameters and computations. The Ghost model first obtains the necessary feature map of half channel of the input features through Conv. These necessary feature maps are used to perform the depth-wise Conv that can obtain similar feature maps of the necessary feature maps. Finally, the two parts of the feature maps from Conv and depth-wise Conv are spliced. The schematic diagram of the Ghost module is shown in Figure 5B. Specifically, we used the primary convolution $Y' = X * f'$ generate m feature maps $Y' \in \mathbb{R}^{h' \times w' \times m}$. To obtain the required n feature maps, the following cheap operations are used for each intrinsic feature in Y':

$$y_{ij} = \Phi_{i,j}(y'_i), \forall\ i = 1, 2, \ldots m, j = 1, 2, \ldots \ldots s \qquad (1)$$

where y'_i is the ith intrinsic feature map in Y' and $\Phi_{i,j}$ is the depth-wise Conv operation to generate the jth (except the last one) Ghost feature map y_{ij}; y'_i can obtain one or more feature maps. The last $\Phi_{i,s}$ is the identity mapping to preserve the intrinsic feature map as shown in Figure 5B. We can obtain $n = m \cdot s$ feature maps for $Y = [y_{11}, y_{12} \ldots y_{ms}]$, which are taken as the output of the Ghost module. The value of the Ghost module FLOPs can be expressed as $\frac{n}{s} \cdot h \cdot w \cdot c \cdot k \cdot k + \frac{n}{s}(s-1) \cdot h \cdot w \cdot k \cdot k$. The operations $\Phi_{i,j}$ are convoluted on one channel. One convolution kernel of ordinary convolution is convoluted on every channel. The computational cost of the depth-wise Conv operation is much lower than that of the ordinary convolution.

The original convolution operation in the YOLOv5 backbone network is upgraded to Conv stacking with depth-wise Conv in the Ghost module, which can raise the operation speed and reduce the number of parameters of the model.

3.3 Transformer encoder block

In the case of ship detection, the classification result of the model can be affected because of the high similarity of ship features. Generally, an image contains rich visual information, such as the object and background information. The key is to fully mine the information in the sample and solve the problem of low accuracy. The Transformer's (Vaswani et al., 2017; Zhu et al., 2020) self-attention mechanism is used to learn the association between the foreground and background in the sample, so that the model can focus on the key areas for detection. The Transformer can improve the detection accuracy of objects. First, the Transformer constructed the sample features into sequence form and added positional encoding. Then, the self-attention mechanism of the Transformer model was used to learn the association between each feature block and assigned different attention to each feature block. Lastly, the original feature sequences are fused, and each feature block in the sequence can contain useful

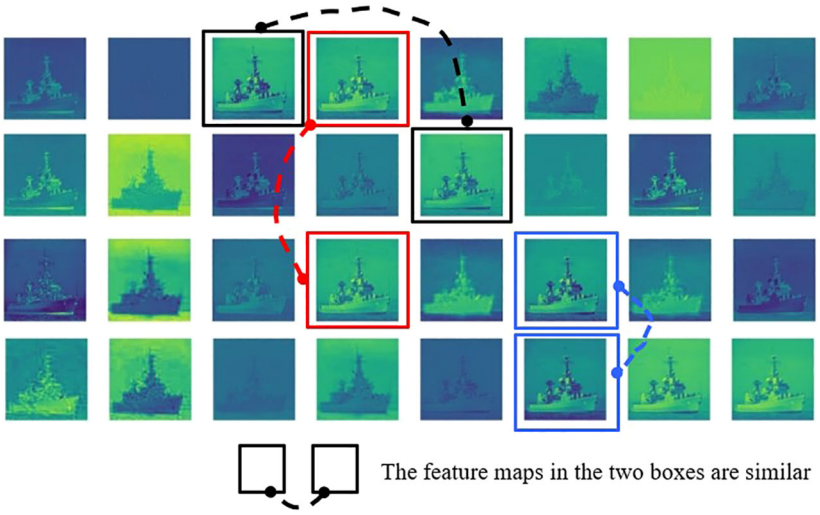


FIGURE 4
Redundant feature maps generated by original convolution.

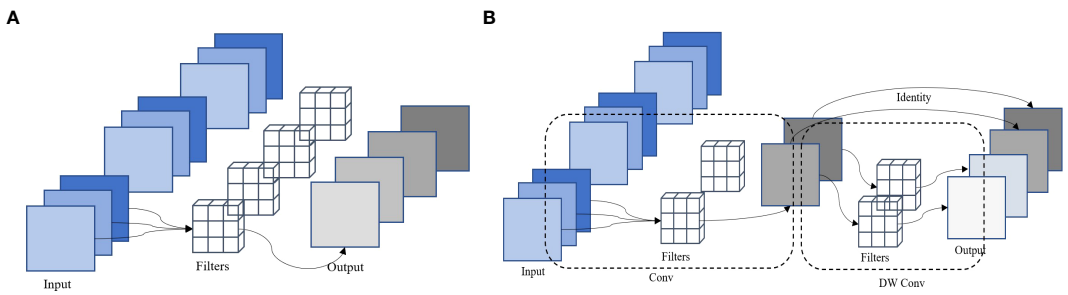


FIGURE 5
Conv and Ghost module structure diagrams **(A)** The Conv operaton, **(B)** The Ghost module.

information for detection in other feature blocks. These operations can enhance the feature expression ability of training samples and improve the accuracy of classification and detection.

The Transformer encoder comprises L layers of alternating Multihead Self-Attention (MSA) and Multilayer Perceptron (MLP) modules. The model structure of Transformer is shown in Figure 6. Therefore, the output $Z_l$if layer l based on the Transformer encoder is:

$$Z'_l = MSA(LN\ (Z_{l-1})) + Z_{l-1} \quad (2)$$

$$Z_l = MLP\ (LN\ (Z'_l)) + Z'_l \quad (3)$$

where $l = \{1, 2,\ldots, L\}$ represents the number of layers, $LN(\cdot)$ presents the layer normalization operation, and Z'_l represents the output of the Lth layer of the MSA. The final output (hidden feature) of the Transformer encoder is $Z_L \in \mathbb{R}^{N\times P\times P}$.

To improve the detection accuracy of the network without deepening the network depth, we focused on the fusion of multilayer features on the PANet and optimization of the feature transfer on the FPN structure. High-quality feature map upsampling and forward transfer were obtained, and the interference of the underlying feature background was reduced. The Transformer was integrated into YOLOv5, which could improve the feature expression. The Transformer was taken into the end of the backbone structure and CSPnet module of the FPN structure. The spatial areas of low-level features were weighted by the salient target position information contained in the attention map, which highlighted the salient regions of the low-level features and suppressed the interference of the background. Thus, it could be more conducive to the identification and classification of ships.

The Transformer could guide the model's attention to reliable and useful channels, while reducing the impact of unreliable and useless background channels. Based on the YOLOv5 model, we integrated the Transformer block at the end of its backbone and Neck networks. Because the resolution of the images at the end of the backbone network was relatively low, applying the Transformer module on the low-resolution feature maps could reduce the additional computational cost.

4 Experiment

4.1 Datasets

In marine transportation, there are generally five basic types of vessels, namely, cargo ships, general cargo ships, carrier ships, bulk carriers, and oil tankers. In addition, there are other types of ships, such as ro-ro, reefer, barge, and liquified natural gas carrier. Among them, cargo, carrier, and cruise ships account for 60%–70% of global ships (Electronic Quality Shipping

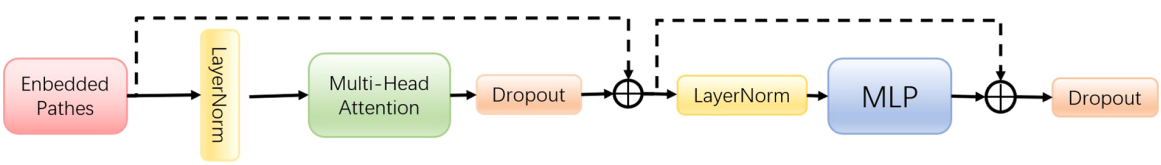


FIGURE 6
Transformer encoder architecture.

Information System, 2020). Therefore, we selected a ship dataset, which can be found on Kaggle (Jain, 2021). It includes five different ship types: cargo, military, carrier, cruise, and oil tanker. Additionally, the dataset comprises 7,604 ship images, including 1,853 cargo ships, 916 warships, 829 transport ships, 1,281 cruise ships, and 1,062 tankers. Figure 7 shows sample images that were randomly selected from ship datasets.

The "JiuHang750" USV is designed and fabricated to detect and trace ships and is used as our research platform. The USV was equipped with the three-light photoelectric platform, which comprises a 30× continuous zoom high-definition visible light camera, an 80-mm uncooled infrared thermal imager, and a 5-km laser rangefinder. The visible light camera can achieve 30× optical zoom and output video images with a 1,920 × 1,080 resolution; the stabilization accuracy of the photoelectric platform reaches 0.5 mrad, the rotation range can reach 360°, and the pitch angle can reach 70° up and down. Based on this optoelectronic platform, the "JiuHang750" USV collected images in the areas of Yellow Sea to test the detection ability of the algorithm in the maritime environment in October and December 2021 and February 2022. The video screenshots are shown in Figure 8.

4.2 Experimental environment and parameters

To ensure experimental consistency, all experiments in this study were carried out under the same hardware platform and software framework. All models used an NVIDIA RTX2080Ti GPU (11 GB) for training and testing. The operating system was CentOS 7, the test framework was PyTorch1.9.0, and the CUDA version 10.2 was the parallel computing framework. The networks were trained for 200 epochs.

4.3 Analysis of results

4.3.1 Comparison with other object detection algorithms

In this section, we evaluate the performance of the proposed improved YOLOv5 algorithm. Multiple evaluation indicators were used to evaluate the performance of the different object detection algorithms, including Average Precision (AP), Precision (P), Recall (R), and F1-score. The mean average

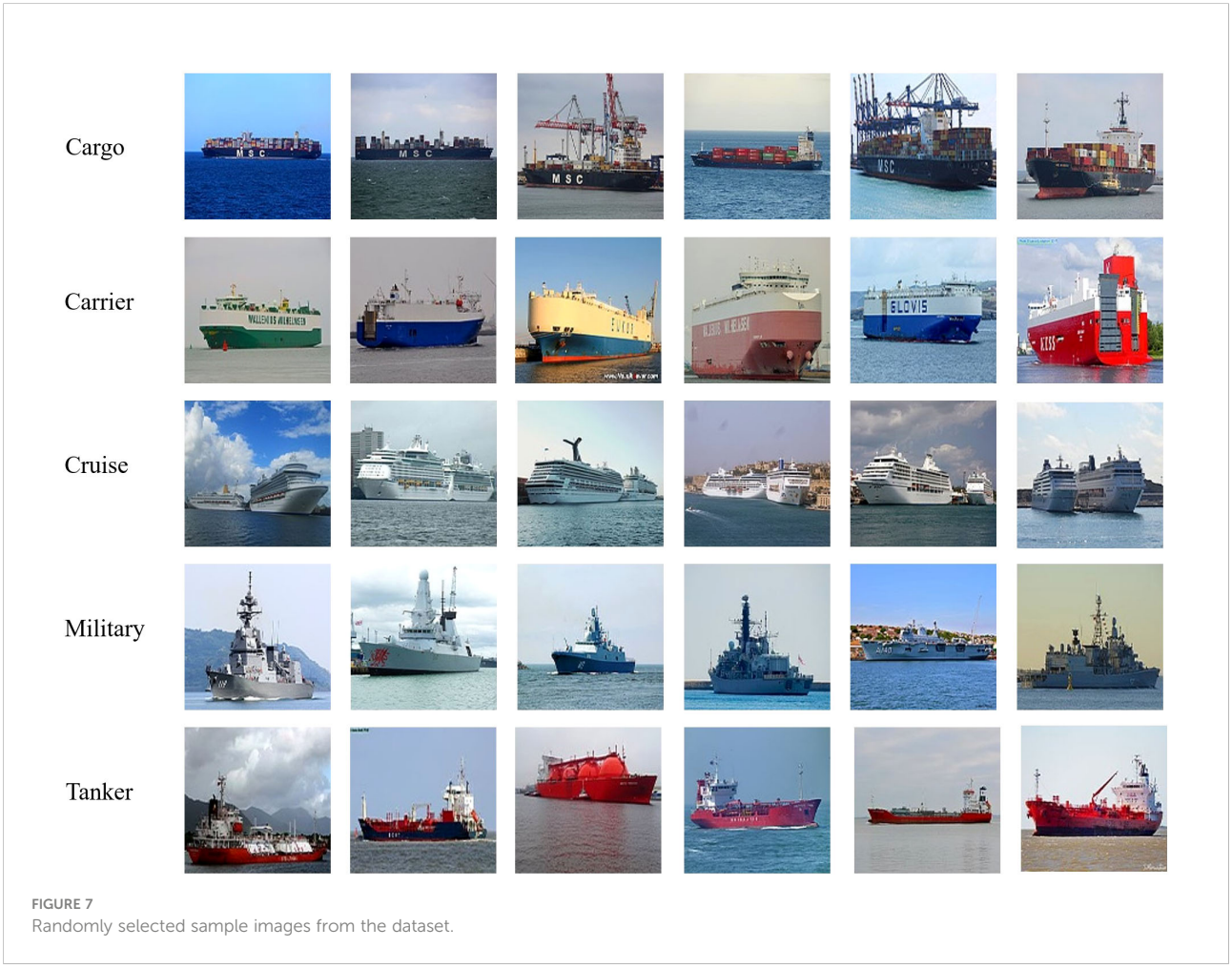


FIGURE 7
Randomly selected sample images from the dataset.

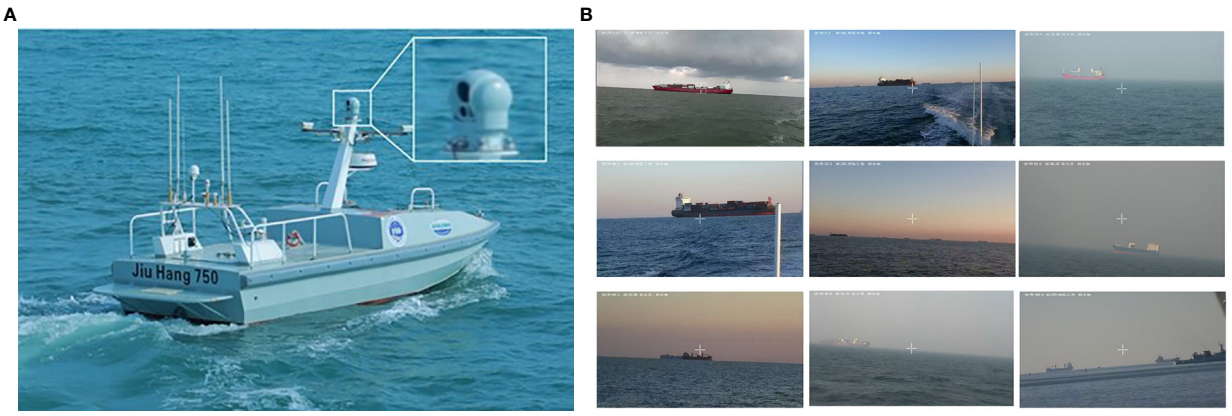


FIGURE 8
(A) "JiuHang750" USV and **(B)** its video images collected under different weather conditions.

precision (mAP) was adopted to evaluate the accuracy of the object detection algorithms. P was adopted to measure the algorithm classification accuracy, and R was used to measure the recall ability of the algorithm detection. The F1-score can consider both P and R. The frames per second (FPS) is an important indicator to evaluate the speed of a target detection algorithm, which indicates the number of frames per second processed by the detection algorithm. The calculation formulas are presented as follows:

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$AP = \int_0^1 PR \cdot dR \tag{6}$$

$$F1 = 2 \times \frac{P \times R}{P + R} \tag{7}$$

$$mAP = \frac{\sum_{i=1}^{n} AP_i}{n} \tag{8}$$

where P represents the precision rate, R represents the recall rate, TP represents the situation where the prediction and label are both ships, and FP represents the situation where the prediction is a ship but the label is the background; FN represents the situation where the prediction is the background but the label is the ship. n represents the number of classes.

Four deep learning and two lightweight algorithms were used to compare with the proposed algorithm, including SSD, YOLOv3, YOLOv4, YOLOv5, YOLOv3-tiny, and YOLOv4-tiny. The specific test results in Table 1 show that the proposed algorithm achieves the best results between detection speed and accuracy, and its detection precision is better than SSD, YOLOv3, YOLOv4, YOLOv3-tiny, and YOLOv4-tiny. The ship detection precision of our study is 0.7% and 1.5% higher than that of YOLOv3 and YOLOv4, respectively, and 28.8% and 43.9% higher than that of YOLOv3-tiny and YOLOv4-tiny, respectively. The FPS value of our algorithm was 138. The detection speed of our algorithm is faster than that of SSD,

TABLE 1 Performance comparison of SSD, YOLOv3, YOLOv3-tiny, YOLOv4, YOLOv4-tiny, YOLOv5 and the proposed algorithm in the ship dataset.

Methods	mAP0.5 (%)	mAP@0.5:0.95 (%)	P (%)	R (%)	F1 (%)	Model size (M)	FPS
SSD	95.2	72.1	81.3	85.7	83.4	92.6M	83
YOLOv3	95.9	77.3	95.1	94.8	94.9	117M	54
YOLOv3-tiny	72.6	31.4	67.0	72.4	69.6	16.6M	149
YOLOv4	93.5	77.5	81.2	**96.4**	88.1	488M	26
YOLOv4-tiny	88.9	63.9	51.9	91.5	66.23	45M	98
YOLOv5	95.3	70.9	**95.8**	94.5	95.1	13.61M	131
Ours	**96.6**	**79.2**	**95.8**	94.7	**95.2**	**12.24M**	138

The bolded areas inside the table represent the best performance.

YOLOv3, YOLOv4, YOLOv4-tiny, and YOLOv5. The results show that the detection algorithm of the proposed algorithm achieves optimal results between speed and accuracy. Therefore, the ship detection algorithm of our study is suitable for application to USVs.

Figure 9 shows the Precision–Recall (P–R) curves of YOLOv3, YOLOv4, YOLOv5, and the proposed algorithm. The P–R curves represent the predictions of the test set samples as positive samples under different thresholds, and different precision and recall rates are obtained. The larger the area enclosed by the P–R curve with the coordinate axis, the better the precision and recall of the detection algorithm. After comparison, it can be seen that the area enclosed by the algorithm in this study is larger than that of other object detection algorithms. Hence, the algorithm in this paper is better than the three algorithms of YOLOv3, YOLOv4, and YOLOv5 in terms of detection performance.

4.4 Comparison of actual test results of USV

To test the detection effect of the proposed algorithm in an actual maritime environment, we conducted several maritime experiments in the Yellow Sea near Qingdao to detect and classify ships. Figure 10 shows the detection results of the proposed algorithm and lightweight models YOLOv3-tiny, YOLOv4-tiny, and YOLOv5 on images collected by the

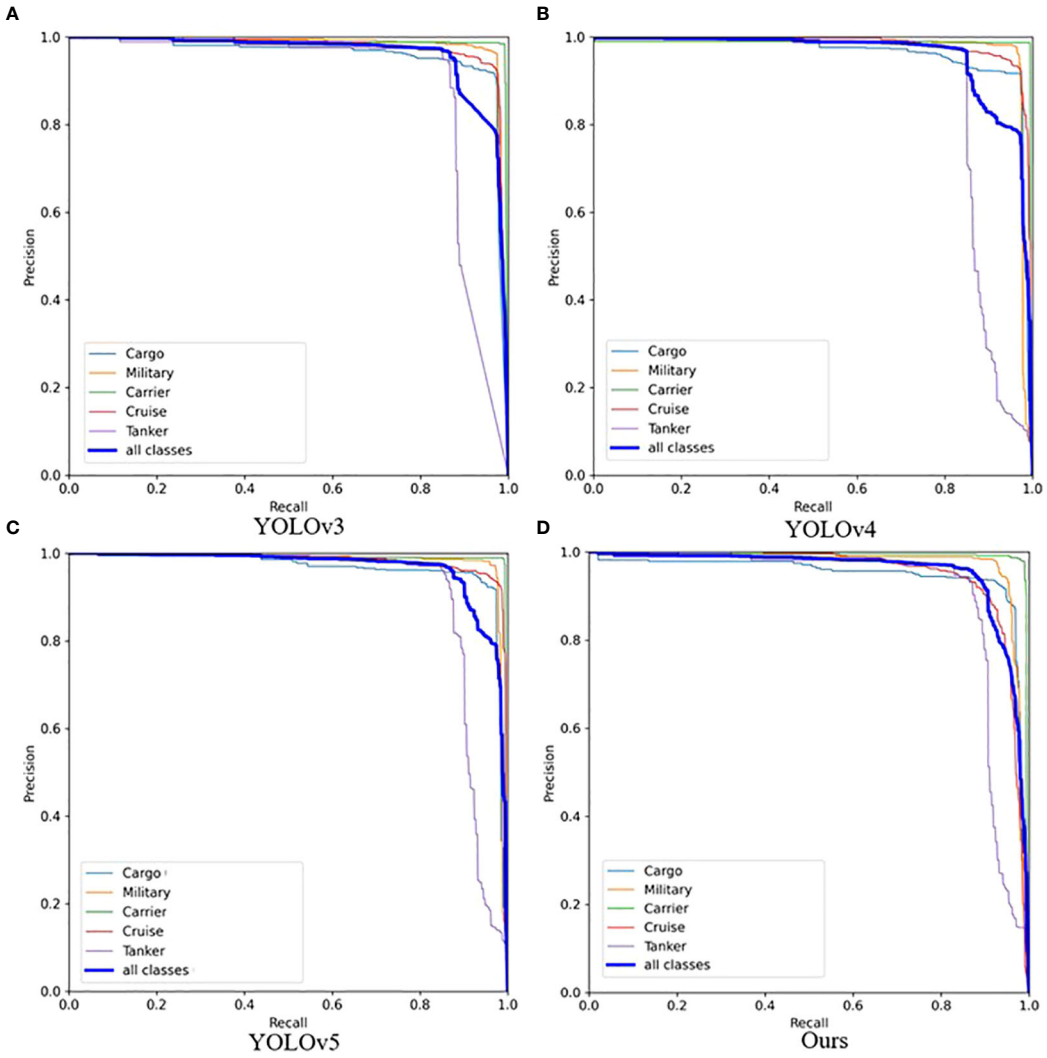


FIGURE 9
Precision-Recall (P-R) curves of different object detection algorithms on the ship dataset (IoU = 0.5). **(A)** is from YOLOv3, **(B)** is from YOLOv4, **(C)** is from YOLOv5 and **(D)** is from our proposed algorithms.

"JiuHang750" USV. The results show that the proposed algorithm has the best detection performance in the actual maritime environment. Each column presents the original image and the detection results of YOLOv3-tiny, YOLOv4-tiny, Yolov5, and the proposed algorithm from left to right. The first row shows a ship clearly. Although YOLOv4-tiny detects the object, the detection box is significantly smaller than the actual position of the ship in the image. In the second row, we show the image of a ship that is far away from the ship and has wake waves. YOLOv4-tiny recognizes the waves as a ship object, and the detection accuracy of the proposed algorithm is significantly higher than that of other detection algorithms. The third row shows the ship image under the swing of the USV. YOLOv3-tiny and YOLOv4-tiny also detect the ship object, but the detection box is inconsistent with the actual position of the ship in the image; additionally, YOLOv5 does not detect the ship object. The fourth row shows the image of the ship under dark clouds; all algorithms detect the ship object, but YOLOv4-tiny splits one ship object into two different objects. Furthermore, the accuracy of the proposed algorithm is significantly higher than that of other detection algorithms. The fifth and sixth rows show the ship image in the case of sea fog. Two images do not detect the ship object of YOLOv3-tiny and YOLOv4-tiny, and the detection accuracy is also low; however, the accuracy rate of the ship object detected by the proposed algorithm is higher.

4.5 Ablation experiments

To further evaluate the effectiveness of the proposed algorithm and each module, ablation experiments were designed, and Table 2 presents the results. Experiment 1 is set as the benchmark, which demonstrates the performance of YOLOv5s without any modification. Then, we replaced the original anchor boxes in experiment 2. In experiment 3, we added the Ghost module to the backbone structure. In experiment 4, we included the attention mechanism in the Neck network structure.

The results show that the mAP increased by 0.11% in experiment 2 after replacing the original anchor boxes. The original Conv operation in the backbone was replaced by Conv stacking with depth-wise Conv in the Ghost module in experiment 3. Compared with the results achieved by YOLOv5s, the mAP increased by 0.14% and the size of the model reduced by 1.45 M. In experiment 4, we integrated the Transformer into the end of the backbone network and FPN structure, and the mAP increased by 0.43%. These results show



FIGURE 10
Detection results of different object detection algorithms in various environments collected by "JiuHang750" USV.

TABLE 2 The results of the ablation experiment.

Experiment	Anchor boxes	Ghost module	Transformer	mAP0.5 (%)	Size (M)	FPS
1				94.80	13.61	131
2	✓			94.91	13.61	131
3		✓		95.06	**12.16**	**140**
4			✓	95.23	13.60	133
5	✓	✓	✓	**96.6**	12.24	138

The bolded areas inside the table represent the best performance.

that the addition of the two modules can improve the detection ability of the algorithm.

5 Conclusions

In this study, an object detection algorithm is improved based on the YOLOv5 model for USVs. First, based on the shape characteristics of ships, the K-means algorithm was used to optimize the initial value of the anchor boxes. Second, the Ghost module was added to the backbone, thus reducing the size of the network and improving detection efficiency. Third, we integrated the Transformer at the end of the backbone and Neck structures in the YOLOv5 network, thereby improving the model's attention to reliable and useful features. Finally, we conducted experiments to verify the accuracy of the proposed algorithm and its effectiveness in real-time detection tasks. In comparison with other deep learning object detection algorithms, the results show that the proposed algorithm achieves a mAP of 96.6%. Our model size is the smallest among all other algorithms used for comparison and only reaches 12.24 M. The detection results in different maritime environments are also significantly better than those of other detection algorithms. Additionally, our algorithm has obtained good detection results in the sea fog environment. Furthermore, the proposed algorithm was applied to the vision system of the "JiuHang750" USV and successfully realized the identification and classification of the surrounding ships of the USV.

Sea images are easily affected by weather and lighting, resulting in unclear objects on images; thus, feature extraction of objects can become difficult. In future research, we can resolve this problem by focusing on the hardware technology for image acquisition, image stabilization, and other aspects. In addition, the dataset used in this study is small in terms of size, and it is necessary to collect more photos of objects on the sea, and especially pictures at different times and light conditions.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

JZ conceived, planned, and performed the designs and drafted this paper. YM and PR provided guidance and reviewed this paper. JJ provided the design ideas and edited this paper. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Key Research and Development Program of China (grant number 2021YFC3101101) and the National Key Research and Development Program of China (grant number 2017YFC1405203).

Acknowledgments

The authors would like to thank China University of Petroleum (East China) for technical support and all the members of our team for their contribution to the sea experiment of the USV.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Bai, Y., Lei, S., and Liu, L. (2021). "The ship object detection based on Sea-Sky-Line," in *2021 6th International Conference on Automation, Control and Robotics Engineering (CACRE)*, IEEE. 456–460. doi: 10.1109/CACRE52464.2021.9501

Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint* arXiv:2004.10934. doi: 10.48550/arXiv.2004.10934

Chen, X., Ling, J., Wang, S., Yang, Y., Luo, L., and Yan, Y. (2021). Ship detection from coastal surveillance videos *via* an ensemble canny-gaussian-morphology framework. *J. Navigation* 74, 1252–1266. doi: 10.1017/S037346332100

Chen, T., Wang, N., Wang, R., Zhao, H., and Zhang, G. (2021). One-stage CNN detector-based benthonic organisms detection with limited training dataset. *Neural Networks* 144, 247–259. doi: 10.1016/j.neunet.2021.08.014

Dzvonkovskaya, A., and Rohling, H. (2010). "Cargo ship RCS estimation based on HF radar measurements," in *11-th International Radar Symposium*. IEEE. 1–4.

Electronic Quality Shipping Information System (2020) *The world merchant fleet in 2020*. Available at: https://www.equasis.org (Accessed May 17, 2000).

Girshick, R. (2015). "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*. IEEE. 1440–1448.

Gupta, V., Gupta, M., and Singla, P. (2021). Ship detection from highly cluttered images using convolutional neural network. *Wireless Pers. Commun.* 121, 287–305. doi: 10.1007/s11277-021-08635-5

Han, K., Wang, Y., Tian, Q., Guo, J., and Xu, C. (2020). *Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. IEEE. 1580–1589.

Jain, A. (2021) *Game of deep learning: Ship datasets*. Available at: https://www.kaggle.com/datasets/arpitjain007/game-of-deep-learning-ship-datasets (Accessed June 24, 2021).

Jie, Y., Leonidas, L., Mumtaz, F., and Ali, M. (2021). Ship detection and tracking in inland waterways using improved YOLOv3 and deep SORT. *Symmetry* 13(2), 308. doi: 10.3390/sym130203

Lee, S.-J., Roh, M.-I., Lee, H.-W., Ha, J.-S., and Woo, I.-G. (2018). "Image-based ship detection and classification for unmanned surface vehicle using real-time object detection neural networks," in *The 28th International Ocean and Polar Engineering Conference (OnePetro)*. 726–730.

Li, G., and Qiao, Y. (2021). A ship object detection and tracking algorithm based on graph matching. In *Journal of Physics: Conference Series (IOP Publishing)*, vol. 1873 (1), 012056. doi: 10.1088/1742-6596/1873/1/012056

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "Ssd: Single shot multibox detector," in *European Conference on computer vision* (Springer), 21–37. doi: 10.1007/978-3-319-46448-0_2

Liu, C., and Li, J. (2021). Self-correction ship tracking and counting with variable time window based on YOLOv3. *Complexity* 2021, 1–9. doi: 10.1155/2021/7428927

Liu, T., Pang, B., Zhang, L., Yang, W., and Sun, X. (2021). Sea Surface object detection algorithm based on YOLO v4 fused with reverse depthwise separable convolution (RDSC) for USV. *J. Mar. Sci. Eng.* 9, 753. doi: 10.3390/jmse9070

Liu, Z., Zhou, F., Bai, X., and Yu, X. (2013). Automatic detection of ship object and motion direction in visual images. *Int. J. Electron.* 100, 94–111. doi: 10.1080/00207217.2012.687188

Mittal, S., Srivastava, S., and Jayanth, J. P. (2022). "A survey of deep learning techniques for underwater image classification," in *IEEE Transactions on Neural Networks and Learning Systems*. IEEE. 1–15. doi: 10.1109/TNNLS.2022.3143887

Redmon, J., and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint* arXiv:1804.02767. doi: 10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* 28.

Sermi, F., Mugnai, C., Cuccoli, F., and Facheris, L. (2013). "Analysis of the radar coverage provided by a maritime radar network of Co-operative vessels based on real AIS data," in *2013 European Radar Conference* IEEE., vol. 2013. 251–254.

Shi, G., and Suo, J. (2018). "Ship target detection based on visual attention," in *2018 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*. IEEE. 1–4.

Sun, X., Liu, T., Yu, X., and Pang, B. (2021). Unmanned surface vessel visual object detection under all-weather conditions with optimized feature fusion network in YOLOv4. *J. Intelligent Robotic Syst.* 103, 1–16. doi: 10.1007/s10846-021-01499-8

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Advances in neural information processing systems*. 5998–6008

Vesecky, J. F., Laws, K. E., and Paduan, J. D. (2009). "Using HF surface wave radar and the ship automatic identification system (AIS) to monitor coastal vessels," in *2009 IEEE International Geoscience and Remote Sensing Symposium* IEEE., Vol. 3. III–761-III-764. doi: 10.1109/IGARSS.2009.5417876

Vesecky, J. F., Laws, K. E., and Paduan, J. D. (2010). "A system trade model for the monitoring of coastal vessels using HF surface wave radar and ship automatic identification systems (AIS)," in *IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 3414–3417. doi: 10.1109/IGARSS.2010.5650279

Wang, N., Wang, Y., and Er, M. J. (2022). Review on deep learning techniques for marine object recognition: Architectures and algorithms. *IEEE. Control Eng. Pract.* 118. doi: 10.1016/j.conengprac.2020.104458

Xu, F., and Liu, J. (2016). Ship detection and extraction using visual saliency and histogram of oriented gradient. *Optoelectronics Lett.* 12, 473–477. doi: 10.1007/s11801-016-6179-y

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2017). Mixup: Beyond empirical risk minimization. *arXiv preprint*. arXiv:1710.09412. doi: 10.48550/arXiv.1710.09412

Zhang, Y., Wu, S., Liu, Z., Yang, Y., Zhu, D., and Chen, Q. (2020). "A real-time detection USV algorithm based on bounding box regression," in *Journal of Physics: Conference Series*, Vol. 1544. 012022 (IOP Publishing). doi: 10.1088/1742-6596/1544/1/012022

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., and Dai, J. (2020). Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint* arXiv:2010.04159. doi: 10.48550/arXiv.2010.04159

Zou, J., Yuan, W., and Yu, M. (2019). "Maritime object detection of intelligent ship based on faster r-CNN," in *2019 Chinese Automation Congress (CAC)*. IEEE. 4113–4117. doi: 10.1109/CAC48633.2019

Zou, Y., Zhao, L., Qin, S., Pan, M., and Li, Z. (2020). "Ship object detection and identification based on SSD_MobilenetV2," in *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*. IEEE. 1676–1680.

frontiers | Frontiers in Marine Science

TYPE Brief Research Report
PUBLISHED 03 February 2023
DOI 10.3389/fmars.2023.1093665

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Shuming Jiao,
Peng Cheng Laboratory, China
Lina Zhou,
Hong Kong Polytechnic University,
Hong Kong SAR, China

*CORRESPONDENCE
Fangjie Yu
yufangjie@ouc.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 09 November 2022
ACCEPTED 19 January 2023
PUBLISHED 03 February 2023

CITATION
Li Z, Chen G and Yu F (2023) ESPC-BCS-Net: A network-based CS method for underwater image compression and reconstruction.
Front. Mar. Sci. 10:1093665.
doi: 10.3389/fmars.2023.1093665

COPYRIGHT
© 2023 Li, Chen and Yu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ESPC-BCS-Net: A network-based CS method for underwater image compression and reconstruction

Zhenyue Li[1], Ge Chen[1,2] and Fangjie Yu[1,2]*

[1]College of Information Science and Engineering, Ocean University of China, Qingdao, China, [2]Laboratory for Regional Oceanography and Numerical Modelling, Qingdao National Laboratory for Marine Science and Technology, Qingdao, China

The Internet of Underwater Things (IoUT) is a typical energy-limited and bandwidth-limited system where the technical bottleneck is the asymmetry between the massive demand for information access and the limited communication bandwidth. Therefore, storing and transmitting high-quality underwater images is a challenging task. The data measured by cameras need to be effectively compressed before transmission to reduce storage and reconstructed with minor errors, which is the best solution. Compressed sensing (CS) theory breaks through the Nyquist sampling theorem and has been widely used to reconstruct sparse signals accurately. For adaptive sampling underwater images and improving the reconstruction performance, we propose the ESPC-BCS-Net by combining the advantages of CS and Deep Learning. The ESPC-BCS-Net consists of three parts: Sampling-Net, ESPC-Net, and BCS-Net. The parameters (e.g. sampling matrix, sparse transforms, shrinkage thresholds, etc.) in ESPC-BCS-Net are learned end-to-end rather than hand-crafted. The Sampling-Net achieves adaptive sampling by replacing the sampling matrix with a convolutional layer. The ESPC-Net implements image upsampling, while the BCS-Net is used to image reconstruction. The efficient sub-pixel layer of ESPC-Net effectively avoids blocking artifacts. The visual and quantitative evaluation of the experimental results shows that the underwater image reconstruction still performs well when the CS ratio is 0.1 and the PSNR of the reconstructed underwater images is above 29.

KEYWORDS

internet of underwater things, underwater image, compressed sensing, deep learning, convolutional neural networks

1 Introduction

The internet of underwater things (IoUT) is an emerging communication ecosystem to facilitate an integrated, reliable, and coordinated communication network (Jahanbakht et al., 2021) that connects different underwater devices in water bodies (rivers, lakes, and oceans) and underwater environments. The underwater devices include underwater vehicles (sea-bots, remotely operated vehicles, underwater trackers) and underwater sensors (Bello and Zeadally, 2022). By connecting more and more devices to the IoUT, the ecosystem generates a huge amount of data, known as Big Data. However, due to the large size of the captured images and the low memory of low-power embedded devices, communication of underwater images becomes very difficult. Furthermore, the traditional big data processing methods (Cao et al., 2018) that rely on statistical properties lack generalization ability. JPEG and other traditional compression algorithms have limitations regarding reconstruction quality, data rate, and compression performance, making them unsuitable for resource-constrained IoUT (Monika et al., 2022b).

Compressed sensing (CS) theory has several names: compressive sampling, compressed sensing, and compressive sensing. CS theory breaks through Nyquist's theorem, and it is a pre-processing technique that exploits the signal's sparsity for sampling the data (Zhang et al., 2022). CS is more hardware-friendly, especially with simultaneous sampling and compression. Some CS-based methods have been proposed to solve underwater data processing. The SPIHT compression algorithm for underwater images was proposed based on embedded coding compression and CS (Cai et al., 2019). Zhang et al. (2021) used CS to overcome underwater image distortions. The CS multiscale entropy feature extraction method to process target radiation noise is efficient and accurate (Lei et al., 2022). Nevertheless, these traditional CS-based methods face the drawbacks of requiring manual parameter adjustment for the signal, time-consuming calculations, and poor generalization.

With the development of CS and Deep Learning, the network-based CS methods have been applied to magnetic resonance imaging (Kilinc et al., 2022), acoustic transmission (Atanackovic et al., 2020), and synthetic aperture radar imaging (Cheng et al., 2022). The network-based CS method allows the reconstruction of images quickly once the network has been trained. Yuan et al. (2020) proposed SARA-GAN based on Generative Adversarial Networks with the Self-Attention mechanism for CS-MRI reconstruction. In addition, a method called LightAMC based on CS and a convolutional neural network was proposed for a non-cooperative communication system (Wang Y et al., 2020). The parameters of these network-based CS methods are trained end-to-end rather than manually tuned, with the advantage of higher generalization and faster reconstruction.

To improve the CS performance of underwater image reconstruction, we propose ESPC-BCS-Net. The following are the particular contributions of the proposed ESPC-BCS-Net:

1. It is a novel network-based CS method where parameters (excluding hyperparameters) are trained end-to-end rather than through manual adjustment (including the sampling matrix and sparse matrix).
2. The ESPC-BCS-Net can be trained in unison, while the Sampling-Net can be used separately for underwater image sampling.
3. The Sampling-Net achieves adaptive sampling by replacing the fixed sampling matrix with a learnable convolutional layer.
4. The ESPC-Net avoids blocking artifacts and improves reconstruction quality.

2 Related works

This section will present related works and briefly introduce CS and CS-based reconstruction methods.

2.1 CS overview

Mathematically, CS reconstruction is to infer the objective signal $x \in \mathbb{R}^N$ from its randomized CS measurements:

$$y = \Phi\Psi s = \Theta s = \Phi x \tag{1}$$

where $\Phi \in \mathbb{R}^{M\times N}$ is the sampling matrix, Θ is the sensing matrix, Ψ is the sparse matrix, s is the sparse coefficient. CS ratio is defined as $\frac{M}{N}$, $M \ll N$. In block compressed sensing (BCS), blocks of images are processed simultaneously rather than the entire image, which reduces the processing time. The image is divided into small blocks of size B×B. The vector y_i can be expressed as:

$$y_i = \Phi_{Bi}\, x_i \tag{2}$$

where x_i presents the vector form of the ith image block and Φ_{bi} is the ith measurement matrix of size B×B. BCS solves the problem of high decoding computational complexity by independently measuring and recovering non-overlapping blocks, but the images can lead to blocking artifacts (Li et al., 2017).

2.2 CS reconstruction methods

We classify the existing CS into three categories: iteration-based method, optimization-based CS method, and network-based CS method. The general iteration-based method for CS reconstruction is:

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{\Phi x - y}||_2^2 + \lambda\mathcal{R}(\mathbf{x}) \tag{3}$$

where the first term $\frac{1}{2}\| \Phi x - y \|_2^2$ is the data fitting term, $\lambda > 0$ is the weighting parameter, $\mathcal{R}(\cdot)$ is the regularization term that requires reconstructed data satisfies the priori information. The optimization-based method for CS reconstruction is to solve the following optimization problem:

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{\Phi x - y}||_2^2 + \lambda||\mathbf{\Psi x}||_1 \tag{4}$$

where the sparsity of the vector Ψx encouraged by the l_1 norm (Qin, 2020). In addition, the common idea of network-based CS method is to replace the operators in traditional CS methods with neural networks (Liu et al., 2021).

3 Proposed ESPC-BCS-Net

This section will briefly introduce the proposed method and then explain the novel ESPC-BCS-Net. As shown in Figure 1, the proposed ESPC-BCS-Net contains Sampling-Net, ESPC-Net, and BCS-Net. We will describe the design of these three networks in the following sub-sections.

3.1 Problem formulation

We divided CS reconstruction into two steps:

$$\mathbf{r}^{(k)}=\mathbf{x}_i^{(k-1)}-\rho*\nabla\frac{1}{2}||\Phi\mathbf{x}_i^{(k-1)}-\mathbf{y}||_2^2 \tag{5}$$

$$\mathbf{x}^{(k)}=\arg\min_{\mathbf{x}_i}\frac{1}{2}||\Phi\mathbf{x}_i-\mathbf{r}^{(k)}||_2^2+\lambda||\mathbf{F}(\mathbf{x}_i)||_1 \tag{6}$$

Where ρ is the step length of the gradient, ∇ express gradient operations, λ is the regularization parameter, $F(\cdot)$ is the transform function to sparse images, x_i is the image block. Inspired by a data-driven adaptively learned matrix (Hong and Zhu, 2018), we improve **Equation (6)** to learn sampling matrix Φ follow **Equation (7)**:

$$\mathbf{x}^{(k)}=\arg\min_{\mathbf{x}_i,\Phi,\mathbf{F}}\frac{1}{2}||\Phi\mathbf{x}_i-\mathbf{r}^{(k)}||_2^2+\lambda||\mathbf{F}(\mathbf{x}_i)||_1 \tag{7}$$

3.2 Architecture of ESPC-BCS-Net

3.2.1 Sampling-Net

The traditional sampling matrix, such as the random Gaussian matrix, is computationally complex and takes up a lot of memory, so we design a learnable sampling matrix. Sampling-Net implements adaptive sampling, which is a learnable convolutional layer used to replace a fixed random matrix $\Phi \in \mathbb{R}^{M\times N}$. The convolutional layer uses M filters of size $\sqrt{N}\times\sqrt{N}$ to sample the image block x_i of size $\sqrt{N}\times\sqrt{N}$. After the sampling network, we get the result $y_i=\Phi_{Bi}x_i$ with size $1\times1\times M$ which easily compresses the underwater image. After the ESPC-BCS-Net network has been trained in unison, Sampling-Net can be used as a compression network. Compared to traditional compression algorithms, Sampling-Net is more suitable for low-power embedded devices as it compresses data through a simple convolution layer.

3.2.2 ESPC-Net

Inspired by the image super-resolution network (Shi et al., 2016), we designed the ESPC-Net (efficient sub-pixel convolutional neural network) for underwater image upsampling and reconstruction. The convolutional layer uses N filters of size 1×1 to replace the $(\Phi_{Bi})^{\mathrm{T}}y_i=(\Phi_{Bi})^{\mathrm{T}}\Phi_{Bi}x_i$. After the convolutional layer, we get the result $(\Phi_{Bi})^{\mathrm{T}}y_i$ with size $1\times1\times N$. Furthermore, the efficient sub-pixel operation is depicted in Figure 1. In the end, we obtained image blocks of the size $\sqrt{N}\times\sqrt{N}$ and used them as input to the BCS-Net.

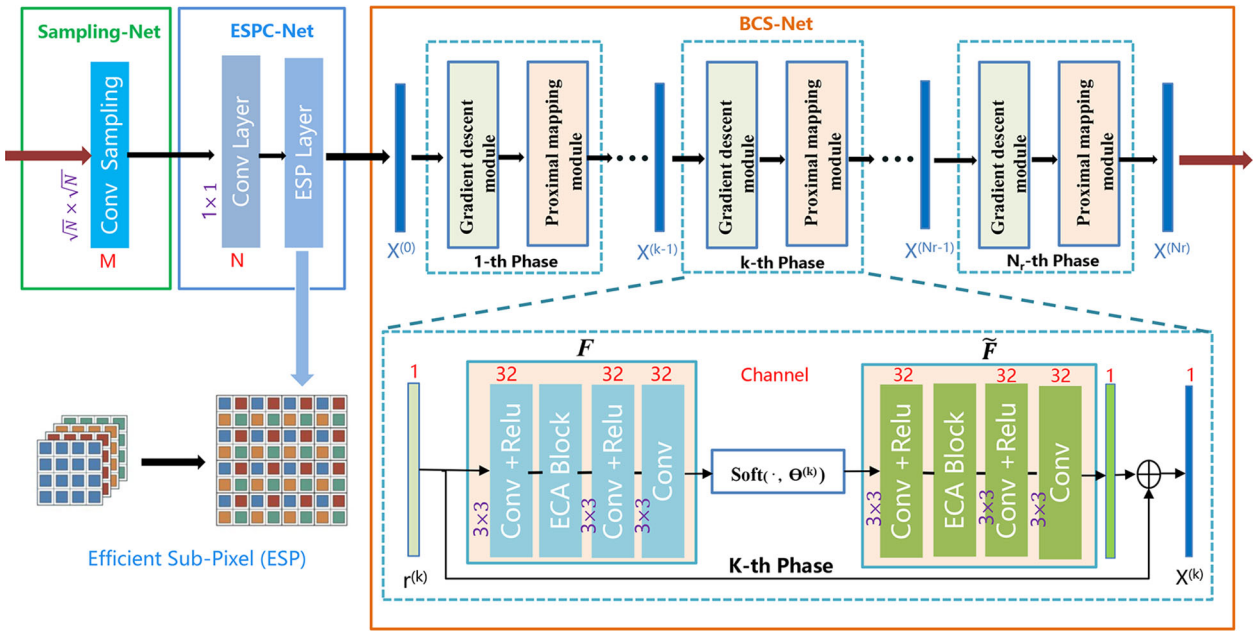


FIGURE 1
The schematic diagram of the proposed ESPC-BCS-Net consists of Sampling-Net, ESPC-Net, and BCS-Net.

3.2.3 BCS-Net

The BCS-Net (block compressed sensing network) is used for underwater image reconstruction and consists of N_r layers network, each containing a gradient module and a proximal module. In particular, the BCS-Net can be trained and used independently as a network for underwater image reconstruction.

Gradient module: corresponds to **Equation (5)**, which is used to generate the $r^{(k)}$. In **Equation (8)**, we omit the calculation process for this $\nabla \frac{1}{2} \| \Phi x^{(k-1)} - y \|_2^2 = \Phi^T(\Phi x^{(k-1)} - y)$. Φ^T is the transpose matrix of Φ.

$$r^{(k)} = x^{(k-1)} - \rho^{(k)} * \Phi^T(\Phi x^{(k-1)} - y) \tag{8}$$

Proximal module: corresponds to **Equation (7)**, which is used to generate the reconstruction result $x^{(k)}$. The soft thresholding function $Soft(\cdot, \theta^{(k)})$ is used to reduce image noise.

$$F^{(k)}(x^{(k)}) = Soft(F^{(k)}(r^{(k)}), \theta^{(k)}) \tag{9}$$

We design the BCS-Net as a residual network structure and $x^{(k)}$ is calculated by **Equation (10)**. $F^{(k)}$ and $\tilde{F}^{(k)}$ have same structures, with an efficient channel attention (ECA) block (Wang Q et al., 2020) in each unit.

$$x^{(k)} = r^{(k)} + \tilde{F}^{(k)}(F^{(k)}(x^{(k)})) \tag{10}$$

3.3 Loss function

The loss function consists of three components, $\mathcal{L}_{constraint}$, $\mathcal{L}_{sparse}$, and $\mathcal{L}_{orth}$. The $\mathcal{L}_{constraint}$ is for network accuracy and the $\mathcal{L}_{sparse}$ is for signal sparsity. The $\mathcal{L}_{orth}$ is an orthogonal constraint for the sampling matrix Φ. The end-to-end loss function for ESPC-BCS-Net as follows:

$$\mathcal{L}_{\mathbf{total}} = \mathcal{L}_{\mathbf{constraint}} + \lambda_1 \mathcal{L}_{\mathbf{sparse}} + \lambda_2 \mathcal{L}_{\mathbf{orth}} \tag{11}$$

with:

$$\mathcal{L}_{\mathbf{constraint}} = \frac{1}{N_b N_x} \sum_{i=1}^{N_x} \sum_{k=1}^{N_r} \| \tilde{\mathbf{F}}^{(k)}(\mathbf{F}^{(k)}(\mathbf{x_i})) - \mathbf{x_i} \|_2^2 \tag{12}$$

$$\mathcal{L}_{\mathbf{sparse}} = \sum_{k=1}^{N_r} \| \mathbf{F}^{(k)}(\mathbf{r}^{(k)}) \|_1 \tag{13}$$

$$\mathcal{L}_{\mathbf{orth}} = \frac{1}{M^2} \| \mathbf{\Phi}^{\mathbf{T}} \mathbf{\Phi} - \mathbf{I} \|_2^2 \tag{14}$$

where the fixed hyperparameters $\lambda_1 = 0.01$, $\lambda_2 = 0.01$, the N_r is the total number of the BCS-Net phase, N_x is the total number of training blocks, N_b is the size of each block x_i, M is the size of Φ, I is the identity matrix.

4 Experiment results and discussion

4.1 Experiment setting

To fairly show the advantages of the ESPC-BCS-Net, we used the same training set (91 images) as ReconNet+ (Lohit et al., 2018) rather than thousands of images. All networks are trained on a workstation with Intel Core i9-10900KF CPU and NVIDIA RTX3060 GPU by PyTorch, taking about 22 hours for each CS ratio (0.5, 0.25, 0.1, 0.04, and 0.01). ESPC-BCS-Net parameters $N_r = 10$, $N_x = 88912$, $N_b = 1089$, and used Adam optimization with a learning rate of 0.0001. In training, the image block size $\sqrt{N} \times \sqrt{N}$ is 33×33. We used the ESPC-BCS-Net for our underwater image reconstruction experiments, and all the underwater images used were accessible through Monika et al. (2022a).

4.2 The results of underwater images

We select different underwater images to sample and reconstruct, including fish, turtles, corals, and underwater scenes. The visual quality comparison of the reconstructed underwater images at different CS ratios is shown in Figure 2. The original images contain three high-resolution images and three noisy images. PSNR (Peak Signal-to-Noise Ratio) and SSIM (structural similarity) evaluated the reconstruction quality. ESPC-BCS-Net has provided a relatively lower CS ratio with convincing visual reconstruction quality. When the CS ratio is 0.1, the PSNR is above 29. At a CS ratio below 0.1, underwater image reconstruction is challenging. As shown in Figure 2E, underwater images reconstructed by ESPC-BCS-Net are still distinguishable when the CS ratio is 0.04.

4.3 Compared with BCS-Net

To demonstrate the usefulness of the Sampling-Net and the ESPC-Net, we conducted a comparative experiment using the BCS-Net and ESPC-BCS-Net. The Gaussian random matrix is used as the sampling matrix, and the same training set for ESPC-BCS-Net was then used to train BCS-Net. As shown in Figure 3, the original images contain a high-resolution image and a dark light image. As shown in Figures 3C, I, the image shows very obviously blocking artifacts with a PSNR below 23. Figures 3D–F, J–L show the results of the ESPC-BCS-Net reconstruction, all of which are better than BCS-Net. By comparison with the BCS-Net, the reconstructed underwater image PSNR and SSIM of the ESPC-BCS-Net are improved by approximately 3.5 and 0.14, respectively.

4.4 Compared with other CS-based methods

To compare with other CS-based methods, we choose Set11 (Kulkarni et al., 2016) as the test set. We compare ESPC-BCS-Net with other CS-based methods, including GSR (Zhang et al., 2014), ReconNet+ (Lohit et al., 2018), BCS (Adler et al., 2017), CSNet (Shi et al., 2017), and FISTA-CSNET* (Xin et al., 2022). Note that the traditional CS-based methods enjoy the advantage of interpretability and do not require training but suffer from the disadvantage of manual adjustment of parameters and computational complexity. In addition, we use the average running time to evaluate these CS-based methods. The GSR is a traditional CS algorithm, which takes the longest time, about 4 minutes. Others CS-based methods are

network-based CS methods, and all take less than 0.3 seconds. Table 1 shows each CS ratio's average PSNR and SSIM for different methods. We highlight the best results in bold and underline the second-best results. Some methods were not trained and tested at a certain CS ratio. For example GSR was not evaluated at a CS ratio of 0.5. It is observed that the ESPC-BCS-Net outperforms the other CS-based methods across five different CS ratios. Even at the lowest CS ratio of 0.01, the PSNR of the reconstructed image is higher than 20. Compared with the BCS, ESPC-BCS-Net performance is superior. The proposed method still performs better reconstruction than the state-of-the-art FISTA-CSNet*. These results indicate that the proposed method produces better reconstruction results while maintaining fast runtime.

5 Conclusion

A novel network-based CS method named ESPC-BCS-Net for underwater image compression and reconstruction is proposed. All parameters (e.g. sampling matrix, sparse transforms, shrinkage thresholds, etc.) of the ESPC-BCS-Net are learned end-to-end, and its structure consists of Sampling-Net, ESPC-Net, and BCS-

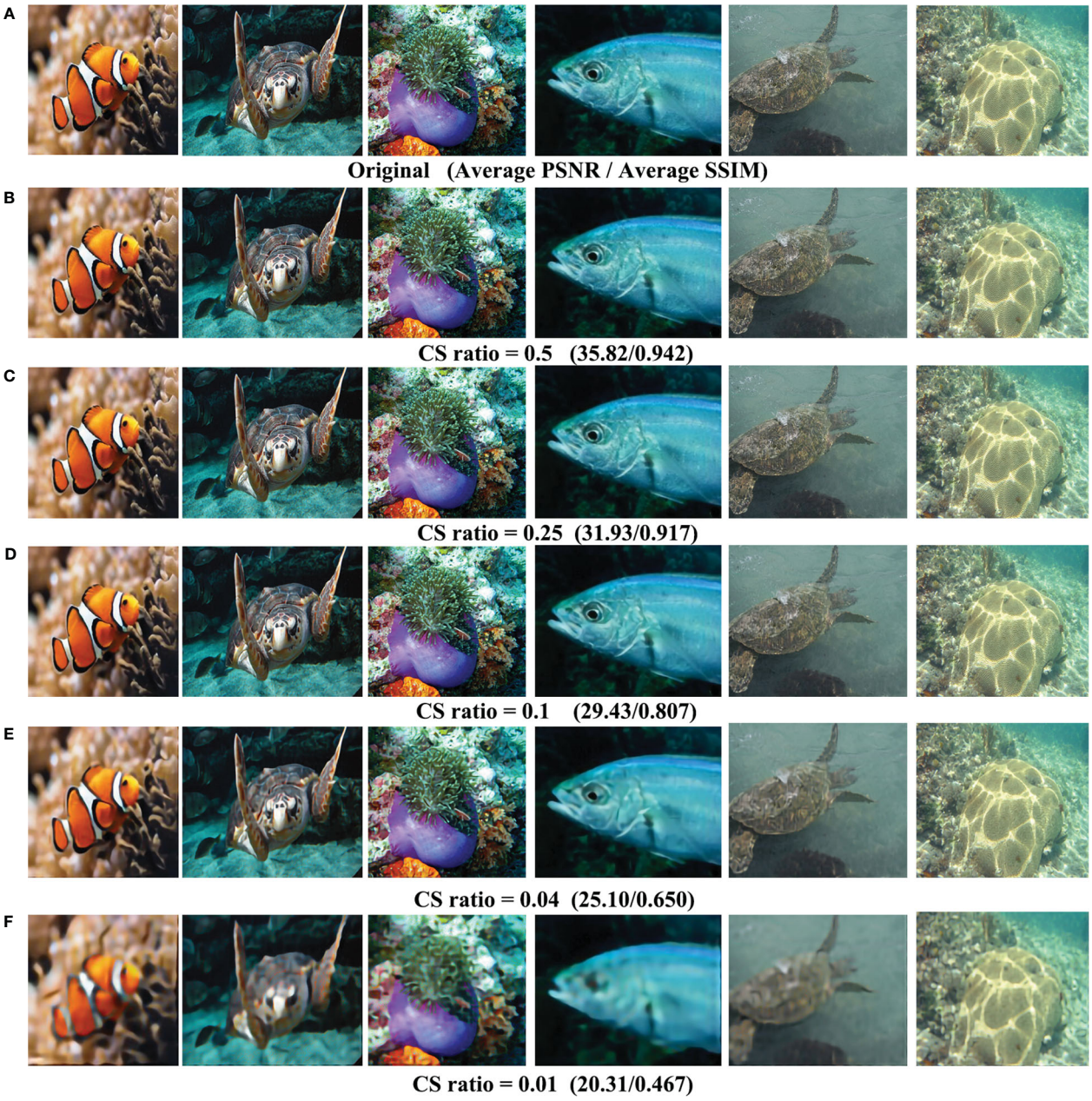


FIGURE 2
Reconstructed underwater images (size of 256×256) by ESPC-BCS-Net at different CS ratios. **(A)** The original underwater images. **(B–F)** Reconstructed underwater images.

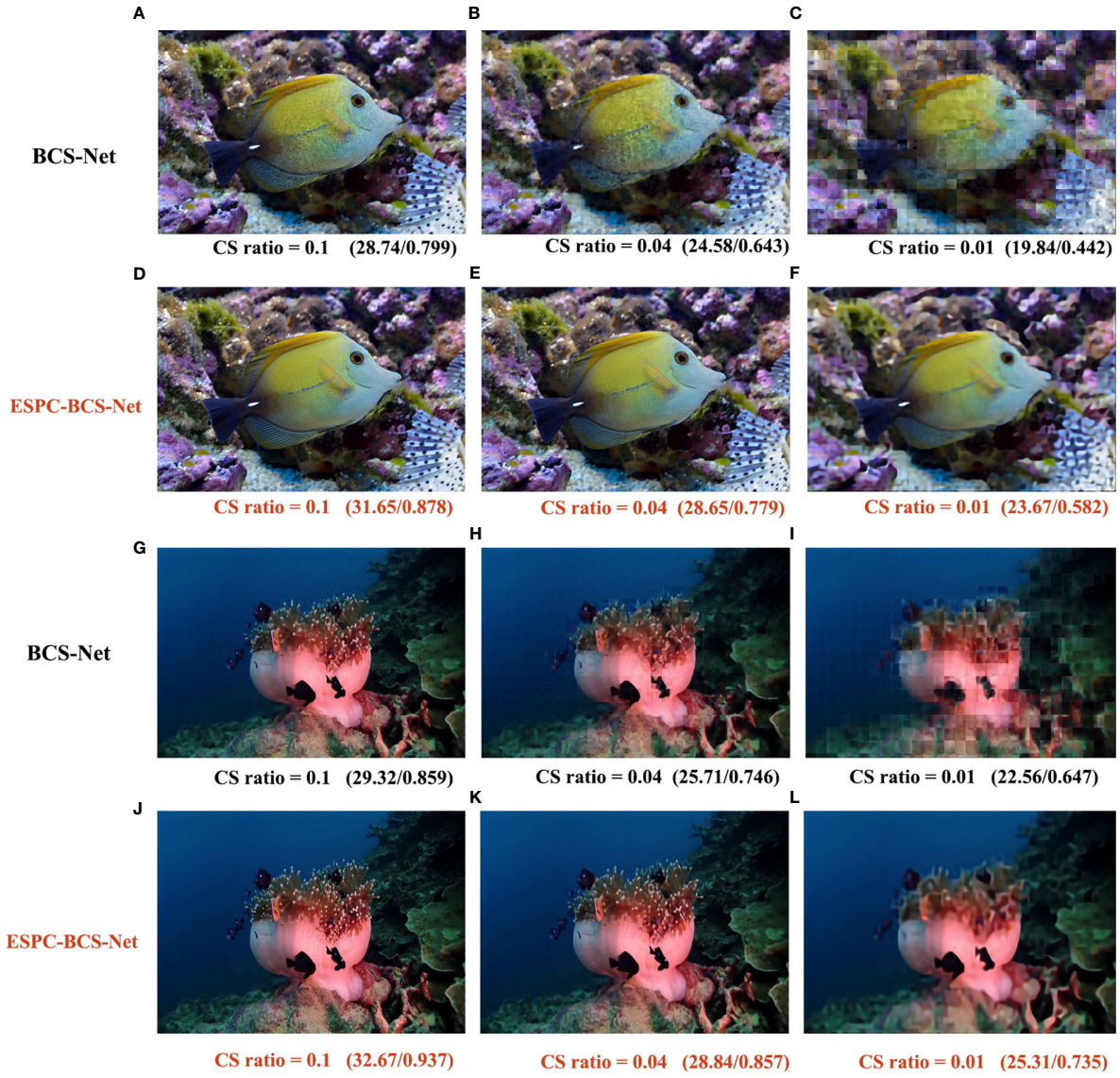


FIGURE 3
Visual comparison of BCS-Net and ESPC-BCS-Net. We evaluate the reconstructed underwater images by PSNR/SSIM. The size of **(A-F)** is 1024×678, and the size of **(G-L)** is 960×540.

TABLE 1 Average PSNR and SSIM of different CS-based methods on Set11 and average running time (in sec) for reconstruction.

CS ratio	Quality	CS-Based Methods					
		GSR	ReconNet+	BCS	CSNet	FISTA-CSNet*	Ours
0.01	PSNR	15.47	16.65	19.15	19.87	**20.65**	20.03
	SSIM	0.368	0.372	0.441	0.497	0.536	**0.536**
0.04	PSNR	19.76	19.64	23.93	23.93	–	**25.52**
	SSIM	0.574	0.535	0.663	0.734	–	**0.789**
0.1	PSNR	26.55	23.39	26.04	27.59	28.53	**29.79**
	SSIM	0.812	0.698	0.797	0.857	0.858	**0.890**
0.25	PSNR	32.26	27.10	29.98	31.70	–	**34.81**
	SSIM	0.924	0.821	0.893	0.927	–	**0.952**

(Continued)

TABLE 1 Continued

CS ratio	Quality	CS-Based Methods					
		GSR	ReconNet+	BCS	CSNet	FISTA-CSNet*	Ours
0.5	PSNR	–	–	34.61	37.19	40.03	**40.18**
	SSIM	–	–	0.943	0.970	0.978	**0.980**
Running Time (s)		235.629	0.019	–	0.025	0.021	**0.018**

Net. The Sampling-Net achieves compressed sampling with only one convolutional layer, which reduces computational costs and is very suitable for resource-constrained IoUT. ESPC-Net and BCS-Net are used for underwater image reconstruction. Furthermore, the ESPC-Net effectively avoids blocking artifacts and improves the reconstruction performance. The results show that ESPC-BCS-Net achieves a PSNR of over 29 for underwater image reconstruction at a CS ratio of 0.1. It can be concluded that ESPC-BCS-Net has effectively improved underwater image compression and reconstruction quality while maintaining fast runtime. The ESPC-BCS-Net mainly focuses on the CS sampling and recovery of underwater images, which can be easily extended to medical images and other fields. The future scope is to implement the proposed method on the hardware platform.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

FY: conceptualization. ZL and FY: designed the experiments. GC and FY: funding acquisition. ZL: methodology. ZL: data processing. ZL: wrote the draft. GC: review and validation. All authors contributed to the article and approved the submitted version.

Funding

The following programs jointly supported this work: (1) the Laoshan Laboratory science and technology innovation projects under Grant No.LSKJ202204304; (2) the Key Laboratory of Marine Science and Numerical Modeling, Ministry of Natural Resources under Grant No.2021-ZD-01.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Adler, A., Boublil, D., and Zibulevsky, M. (2017). "Block-based compressed sensing of images *via* deep learning," in *19th IEEE International Workshop on Multimedia Signal Processing (MMSP)*.

Atanackovic, L., Lampe, L., and Diaman, R. (2020). "Deep-learning based ship-radiated noise suppression for underwater acoustic OFDM systems," in *Global OCEANS Singapore - U.S. Gulf Coast Conference*. doi: 10.1109/IEEECONF38699.2020.9389436

Bello, O., and Zeadally, S. (2022). Internet Of underwater things communication: Architecture, technologies, research challenges and future opportunities. *Ad Hoc Networks*. 135. doi: 10.1016/j.adhoc.2022.102933

Cai, Y.-q., Zou, H.-x., and Yuan, F. (2019). Adaptive compression method for underwater images based on perceived quality estimation. *Front. Inf. Technol. Electronic Engineering*. 20 (5), 716–730. doi: 10.1631/FITEE.1700737

Cao, X., Liu, L., Cheng, Y., and Shen, X. (2018). Towards energy-efficient wireless networking in the big data era: A survey. *IEEE Commun. Surveys Tutorials*. 20 (1), 303–332. doi: 10.1109/COMST.2017.2771534

Cheng, P., Chen, W. Y., Cheng, J. W., Xu, X. M. , and Zhao, J. Q. (2022). A fast ISAR imaging method based on strategy weighted CAMP algorithm. *IEEE Sensors J.* 22 (17), 17022–17030. doi: 10.1109/JSEN.2022.3192534

Hong, T., and Zhu, Z. (2018). Online learning sensing matrix and sparsifying dictionary simultaneously for compressive sensing. *Signal Processing*. 153, 188–196. doi: 10.1016/j.sigpro.2018.05.021

Jahanbakht, M., Xiang, W., Hanzo, L., and Azghadi, M. R. (2021). Internet Of underwater things and big marine data analytics-a comprehensive survey. *IEEE Commun. Surveys Tutorials*. 23 (2), 904–956. doi: 10.1109/COMST.2021.3053118

Kilinc, O., Chu, S., Baraboo, J., Weiss, E. K., Engel, J., Maroun, A., et al. (2022). Hemodynamic evaluation of type b aortic dissection using compressed sensing accelerated 4D flow MRI. *J. Magnet Res. Imag.* doi: 10.1002/jmri.28432

Kulkarni, K., Lohit, S., Turaga, P., Kerviche, R., Ashok, A.Ieee (2016). "ReconNet: Non-iterative reconstruction of images from compressively sensed measurements," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 449–458. doi: 10.1109/CVPR.2016.55

Lei, Z., Lei, X., Zhou, C., Qing, L., and Zhang, Q. (2022). Compressed sensing multiscale sample entropy feature extraction method for underwater target radiation noise. *IEEE Access*. 10, 77688–77694. doi: 10.1109/ACCESS.2022.3193129

Li, R., Duan, X., Guo, X., He, W., and Lv, Y. (2017). Adaptive compressive sensing of images using spatial entropy. *Comput. Intell. Neurosci*. doi: 10.1155/2017/9059204

Liu, J., Zhang, R., Han, G., Sun, N., and Kwong, S. (2021). Video action recognition with visual privacy protection based on compressed sensing. *J. Syst. Architecture*. 113. doi: 10.1016/j.sysarc.2020.101882

Lohit, S., Kulkarni, K., Kerviche, R., Turaga, P., and Ashok, A. (2018). Convolutional neural networks for noniterative reconstruction of compressively sensed images. *IEEE Trans. Comput. Imag* 4 (3), 326–340. doi: 10.1109/TCI.2018.2846413

Monika, R., Dhanalakshmi, S., Kumar, R., and Narayanamoorthi, R. (2022a). Coefficient permuted adaptive block compressed sensing for camera enabled underwater wireless sensor nodes. *IEEE Sensors J.* 22 (1), 776–784. doi: 10.1109/JSEN.2021.3130947

Monika, R., Dhanalakshmi, S., Kumar, R., Narayanamoorthi, R., and Lai, K. W. (2022b). An efficient adaptive compressive sensing technique for underwater image compression in IoUT. *Wireless Networks*. doi: 10.1007/s11276-022-02921-1

Qin, S. (2020). Simple algorithm for L1-norm regularisation-based compressed sensing and image restoration. *Iet Image Processing.* 14 (14), 3405–3413. doi: 10.1049/iet-ipr.2020.0194

Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A. P., Bishop, R., et al. (2016). Real-time single image and video super-resolution using an efficient Sub-pixel convolutional neural network. *IEEE Conf. Comput. Vision Pattern Recognit.* pp, 1874–1883. doi: 10.1109/CVPR.2016.207

Shi, W., Jiang, F., Zhang, S., Zhao, D.Ieee (2017). "DEEP NETWORKS FOR COMPRESSED IMAGE SENSING," in *IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong. 877–882. doi: 10.48550/arXiv.1707.07119

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu., Q. (2020). "ECA-net: Efficient channel attention for deep convolutional neural networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Wang, Y., Yang, J., Liu, M., and Gui, G. (2020). LightAMC: Lightweight automatic modulation classification *via* deep learning and compressive sensing. *IEEE Trans. Vehicul. Technol0* 69 (3), 3491–3495. doi: 10.1109/TVT.2020.2971001

Xin, L., Wang, D., and Shi, W. (2022). FISTA-CSNet: a deep compressed sensing network by unrolling iterative optimization algorithm. *Visual Comput.* doi: 10.1007/s00371-022-02583-2

Yuan, Z., Jiang, M., Wang, Y., Wei, B., Li, Y., Wang, P., et al. (2020). SARA-GAN: Self-attention and relative average discriminator based generative adversarial networks for fast compressed sensing MRI reconstruction. *Front. Neuroinformat.* 14. doi: 10.3389/fninf.2020.611666

Zhang, H., Ni, J., Xiong, S., Luo, Y., and Zhang, Q. (2022). SR-ISTA-Net: Sparse representation-based deep learning approach for SAR imaging. *IEEE Geosci. Remote Sens. Letters.* 19. doi: 10.1109/LGRS.2022.3202557

Zhang, Z., Tang, Y. G., and Yang, K. (2021). A two-stage restoration of distorted underwater images using compressive sensing and image registration. *Adv. Manufact.* 9 (2), 273–285. doi: 10.1007/s40436-020-00340-z

Zhang, J., Zhao, D., and Gao, W. (2014). Group-based sparse representation for image restoration. *IEEE Trans. Image Processing.* 23 (8), 3336–3351. doi: 10.1109/TIP.2014.2323127

frontiers | Frontiers in Marine Science

TYPE Methods
PUBLISHED 23 February 2023
DOI 10.3389/fmars.2023.1132640

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Linlin Wang,
Tsinghua University, China
Changhoon Lee,
Yonsei University, Republic of Korea

*CORRESPONDENCE
Jie Nie
niejie@ouc.edu.cn
Zhiqiang Wei
weizhiqiang@ouc.edu.cn

†These authors have contributed equally to this work and share first authorship

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 27 December 2022
ACCEPTED 02 February 2023
PUBLISHED 23 February 2023

CITATION
Song N, Tian H, Nie J, Geng H, Shi J, Yuan Y and Wei Z (2023) TSI-SD: A time-sequence-involved space discretization neural network for passive scalar advection in a two-dimensional unsteady flow. *Front. Mar. Sci.* 10:1132640. doi: 10.3389/fmars.2023.1132640

COPYRIGHT
© 2023 Song, Tian, Nie, Geng, Shi, Yuan and Wei. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

TSI-SD: A time-sequence-involved space discretization neural network for passive scalar advection in a two-dimensional unsteady flow

Ning Song[1†], Hao Tian[2†], Jie Nie[1*], Haoran Geng[1], Jinjin Shi[1], Yuchen Yuan[1] and Zhiqiang Wei[1*]

[1]College of Information Science and Engineering, Ocean University of China, Qingdao, China, [2]College of Mathematical Science, Ocean University of China, Qingdao, China

Numerical simulation of fluid is a great challenge as it contains extremely complicated variations with a high Reynolds number. Usually, very high-resolution grids are required to capture the very fine changes during the physical process of the fluid to achieve accurate simulation, which will result in a vast number of computations. This issue will continue to be a bottleneck problem until a deep-learning solution is proposed to utilize large-scale grids with adaptively adjusted coefficients during the spatial discretization procedure—instead of traditional methods that adopt small grids with fixed coefficients—so that the computation cost is dramatically reduced and accuracy is preserved. This breakthrough will represent a significant improvement in the numerical simulation of fluid. However, previously proposed deep-learning-based methods always predict the coefficients considering only the spatial correlation among grids, which provides relatively limited context and thus cannot sufficiently describe patterns along the temporal dimension, implying that the spatiotemporal correlation of coefficients is not well learned. We propose the time-sequence-involved space discretization neural network (TSI-SD) to extract grid correlations from spatial and temporal views together to address this problem. This novel deep neural network is transformed from a classic CONV-LSTM backbone with careful modification by adding temporal information into two-dimensional spatial grids along the x-axis and y-axis separately at the first step and then fusing them through a post-fusion neural network. After that, we combine the TSI-SD with the finite volume format as an advection solver for passive scalar advection in a two-dimensional unsteady flow. Compared with previous methods that only consider spatial context, our method can achieve higher simulation accuracy, while computation is also decreased as we find that after adding temporal data, one of the input features, the concentration field, is redundant and should no longer be adopted during the spatial discretization procedure, which results in a sharp decrease of parameter scale and achieves high efficiency. Comprehensive experiments, including a comparison with SOTA methods and sufficient ablation studies, were carried out

to verify the accurate and efficient performance and highlight the advantages of the proposed method.

KEYWORDS

unsteady flow, spatiotemporal feature, CONV-LSTM, passive scalar advection, spatial discretization, discretization acceleration

1 Introduction

Fluid is an indispensable component in the atmosphere and ocean. Additionally, It is of great importance to meteorological services, which attempt to identify safe aerospace and shipping routes. Fluid research is mainly based on numerical simulation by solving partial differential equations Lumley (1979). Mainstream methods include the finite difference method Rai and Moin (1991) and the finite volume method Leschziner (1989) 34. Owing to the rapid variations with a high Reynolds number Kraichnan (1959), the numerical solution requires high-resolution spatial grids to ensure the accuracy of the simulation. In addition, when the Reynolds number folds by ten, the computation load will fold by 1,000. Although current high-performance computing can provide powerful computation ability for these extremely complicated variations, as real-time simulation is always required for emergent forecasting, improving efficiency only in computation power will always be limited and insufficient. Efforts should be made to optimize from the perspective of algorithm architecture.

A scale of previous works has been carried out to reduce the computation load from the perspective of decreasing the resolution of the grids. As early as 1982, Brown et al. Brown (1982) applied a multigrid method to accelerate the numerical solution process of the three-dimensional transonic potential flow. The multigrid method was considered a classic method to reduce computational costs in the traditional numerical solution process because it uses different mesh divisions for different regions instead of high-resolution mesh modeling. Inspired by this thought, Mazhukin et al. Mazhukin et al. (1993) proposed a dynamically adaptive grid method based on a time-dependent coordinate transformation from the physical to a computational space for solving partial differential equations. Additionally, Jin et al. Jin et al. (2014) proposed the application of a coarse grid projection scheme. This method solved the momentum equation on the fine grid level and the pressure equation on the coarse grid level. Therefore, a satisfactory numerical solution should not only retain the simulating accuracy but also improve the computation's efficiency.

This tradeoff issue has been a bottleneck problem for a long period and will remain until a deep-learning solution that utilizes a neural network to take the place of the classic numerical methods module during the spatial discretization procedure is proposed. We use the central difference RUMSEY and VATSA (1993) as an example of traditional numerical methods for spatial discretization and illustrate its basic idea in Figure 1A. To calculate the value of point x at time t, generally, we use neighborhood grid points around x at time t-1,

$$SD = \sum \alpha V(x_{neigborhood}, t-1) \tag{1}$$

where SD is the calculated spatial derivative, and V is a template composed of values at points around x within a certain distance at time t-1 α are fixed coefficients with regard to the corresponding truncation error Lantz (1971). Here, to capture the very subtle variations that occur in the physical movement of unsteady flow, traditional methods usually adopt grids with very high resolution, which leads to an extremely large computation cost. However, the deep-learning method addresses this problem by adopting large-scale grids with adaptively adjusted coefficients instead of traditional methods that adopt small grids with fixed coefficients, as shown in Figure 1B.

$$SD = \sum f_\theta(x_{t-1}) V(x_{neigborhood}, t-1) \tag{2}$$

However, these previously proposed deep-learning-based methods predicted the coefficients only considering spatial correlation among grids, which provided relatively limited context and thus could not describe patterns along the temporal dimension sufficiently, implying that the spatiotemporal correlation of coefficients was not well learned. We propose a novel algorithm to extract grid correlations from spatial and temporal views together to address this problem. We simply illustrate our algorithm in Figure 1C. In our neural network, we added temporal neighborhoods to help predict grid coefficients:

$$SD = \sum f_\theta(x_{t-1}, x_{t-2}, \dots) V(x_{neigborhood}, t-1) \tag{3}$$

where $\{x_{t-n},\dots,x_{t-1}\}$ denotes grid values along the time dimension within a certain range. By adding temporal consideration, we can learn a better mapping function to predict the spatial grid coefficients and achieve a more accurate simulation result. Moreover, we also find that the concentration field, which was used as one of the inputs of the neural network, turns out to be redundant after we add temporal data. Thus, we optimized our method and produced a more efficient neural network with fewer parameters and better accuracy.

Thus, in this paper, we propose a novel time-sequence-involved space discretization neural network (TSI-SD) by taking temporal influence into consideration, which achieves an accurate and efficient simulation result of unsteady flow. Specifically, we produced the proposed neural network based on a classic CONV-LSTM backbone with careful modification by adding temporal

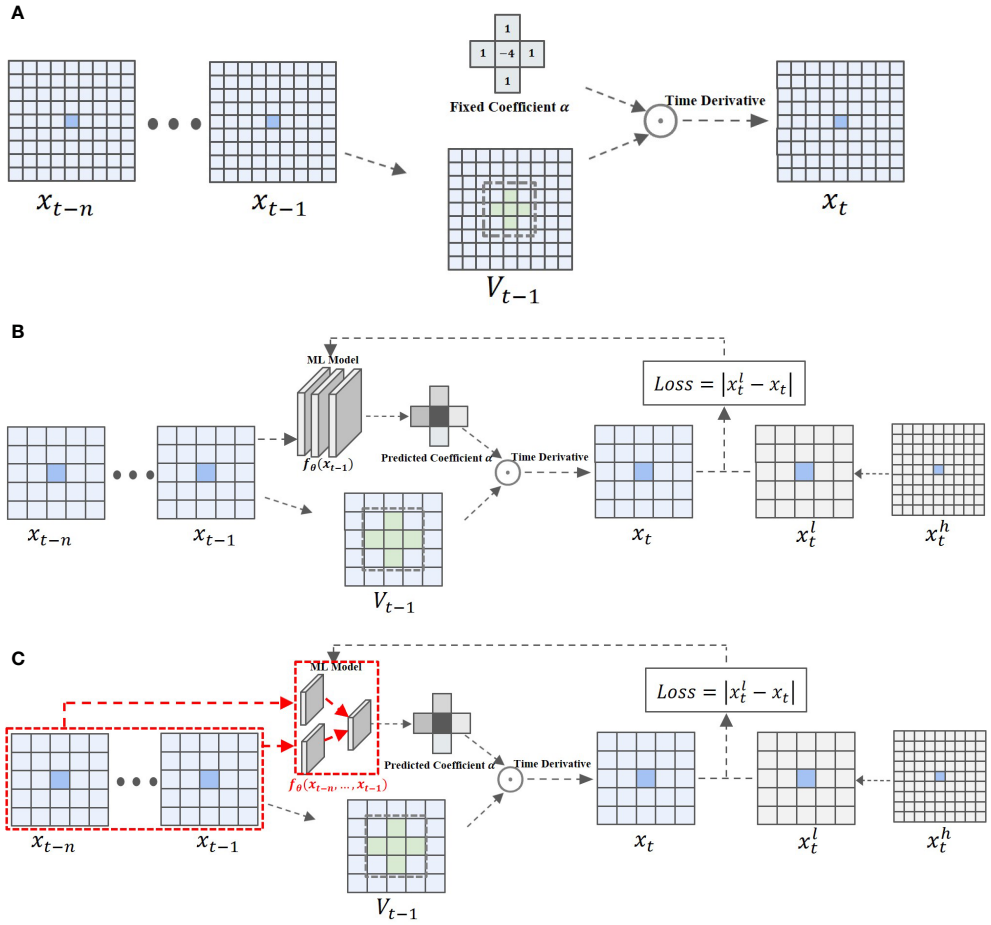


FIGURE 1
This figure shows three methods used to solve the spatial derivative during spatial discretization. **(A)** Figure 1(a) shows a traditional numerical method. **(B)** Figure 1(b) shows the deep-learning-based method. **(C)** Figure 1(c) shows our method. **(A)** The traditional numerical method: in the spatial discretization part, the central difference method is used to calculate the spatial derivative with a fixed spatial discretization coefficient, and then the temporal derivative is calculated in the temporal discretization process to obtain the numerical solution. **(B)**The Deep-learning-based method: In the spatial discretization part, predict the spatial discretization coefficient and calculate the spatial derivative based on the deep learning algorithm and the grid value at time t-1, and then calculate the temporal derivative in the temporal discretization process to obtain a numerical solution. **(C)**. Our method: In the spatial discretization part, predict the spatial discretization coefficient and calculate the spatial derivative based on the deep learning algorithm and the grid value of the time series {t − n, ..., t − 1}, and then calculate the time derivative in the time discretization process to obtain a numerical solution.

information into two-dimensional spatial grids along the x-axis and y-axis separately at the first step and then fusing them together through a post-fusion neural network. After that, we combined the TSI-SD with the finite volume format as an advection solver for passive scalar advection in a two-dimensional unsteady flow. Compared with previous methods that only consider spatial context, our method can achieve higher simulation accuracy, while computation is also decreased after redundant input is removed.

Finally, we highlight the contribution of this paper as follows:

- We optimized the framework of the deep-learning-based numerical simulation methods of unsteady flow. As far as we are aware, we are the first to utilize the temporal relationship to help predict spatial coefficients. Moreover, we also simplified the neural networks by means of decreasing the parameter's scale. Quite simply, our method achieved better accuracy and efficiency compared with existing methods.
- We designed a novel neural network TSI-SD and produced an effective spatial coefficients prediction method that takes both temporal and spatial perspectives into consideration. Our novel framework modeled spatial correlations and temporal correlations and then combined the two aspects properly with a well-designed post-fusion neural network.
- Comprehensive comparisons and ablation studies were carried out with three public datasets, i.e., the numerical solution datasets of the advection equation based on the Vanleer format under the random velocity field, deformed flow velocity field, and the constant velocity field. Sufficient results and explanations were provided and discussed to verify the improvement in both the accuracy and efficiency of the proposed idea.

2 Related work

2.1 Traditional discretization methods of fluid flow simulation

Many researchers have made outstanding contributions in the field of traditional discretization methods of fluid flow simulation Bristeau et al. (1985); Ferziger et al. (2002); Peyret and Taylor (2012); Fletcher (2012); Toro (2013). Based on these theories, Molenkamp et al. Molenkamp (1968) calculated the numerical solution of the convection equation using various finite-difference approximations, and determined that only the Roberts–Weiss approximation convected the initial distribution correctly, but required a huge computational cost. Mikula et al. Mikula et al. (2014) proposed an inflow implicit/outflow explicit finite volume method based on finite volume space discretization and semi-implicit time discretization to solve advection equations. The basic idea is that outflows from cells are handled explicitly, and inflows are handled implicitly. The method achieved outstanding results in terms of stability and computational accuracy. Zhao et al. Zhao et al. (2019) proposed a new improved finite volume method for solving one-dimensional advection equations under the framework of the second-order finite volume method. The method first applied the scalar conservation law to the elements in the finite volume method (FVM) to ensure its conservation in time and space and to ensure advection (i.e., conservation of transport physical quantities); then the time integral values of adjacent grid boundaries are equalized; finally, the equation is established to obtain a numerical solution. Experiments showed that this method has better stability and fewer disspation than the traditional FVM and can maintain the accuracy of the solution. Akitoshi Takayasu et al. Takayasu et al. (2019) proposed a verification calculation method for one-dimensional advection equations with variable coefficients, which was based on spectral methods and semigroup theory. They mainly provided a method for verification calculation using the C_0 semigroup on the complex sequence space l^2,which comes from the solution of the Fourier series. Experiments showed that the given strict error proved the correctness of the exact solution, and the solution has high precision and fast solution speed. Although traditional discretization method shave achieved high solution accuracy, they have the problem of high computational cost if outstanding solution accuracy is desired.

2.2 Traditional discretization acceleration techniques for fluid flow simulation

To solve the problem of high computational cost while calculating high-precision solutions in traditional discretization methods, researchers have proposed acceleration techniques to speed up the numerical discretization solution. Multigrid technology stood out among various approaches Dwyer et al. (1982); Brown (1982); Berger and Oliger (1984); Phillips and Schmidt (1984); Phillips and Schmidt (1985); Zhang (1997); Mazhukin et al. (1993); Jin et al. (2014). Among them, Brown et al. Brown (1982) used the multigrid mesh-embedding technique to solve three-dimensional transonic potential flow. They used small grids to model regions of large local gradients and large-scale grids to model regions with relatively small gradients. Their method improved the speed of solving equation discretization schemes. Phillips et al. Phillips and Schmidt (1984) proposed a multilevel multigrid method combined with a Taylor series interpolation scheme as the best discretization acceleration scheme after comparing the use of simple multigrid and multilevel multigrid methods. Based on the previous method, Phillips Phillips and Schmidt (1985) used multigrid combined with multilevel acceleration technology to realize the accelerated solution of scalar conservation equations. In addition, they proposed a fast finite difference solution to the passive scalar advection-diffusion equation. Although these acceleration methods reduced the computational cost while maintaining high accuracy, high computational cost remained a problem due to the need to retain high solution grid modeling in some complex fluid regions.

2.3 Discretization methods and acceleration techniques combining deep-learning with traditional numerical methods

In recent years, machine learning has been used in the numerical solution of partial differential equations, which have made enormous progress. The combination of machine learning and traditional discretization methods improved the accuracy of the solution and accelerated the numerical calculation Raissi et al. (2019); Ji et al. (2021); Vinuesa and Brunton (2021); Patel et al. (2021); Eliasof et al. (2021); Cai et al. (2022). Based on these methods, O. Obiols-Sales et al. Obiols-Sales et al. (2020) proposed a coupled deep learning and physics simulation framework (CFDNet) to accelerate the convergence of Reynolds-averaged Navier–Stokes simulations. CFDNet was designed to use a single convolutional neural network at its core to predict the main physical properties of fluids, including velocity, pressure, and eddy viscosity. In this paper, CFDNet was evaluated for various use cases, and the results showed that CFDNet significantly speeded up the numerical solution and proved that CFDNet generalized well. Vadyala Shashank Reddy et al. Vadyala et al. (2022) determined the numerical solution of the one-dimensional advection equation using different finite-difference approximations and physical informatic neural networks (PINNs). They trained a neural network to solve supervised learning tasks that obeyed any given laws of physics described by general non-linear partial differential equations. The PINNs approximation was compared with other schemes through experiments, and the results showed that the prediction results obtained by the PINNs approximation were the most accurate. Pathak et al. Pathak et al. (2020) proposed a hybrid ML-PDE solver that combined machine learning and traditional solving methods of the partial differential equation. It can obtain meaningful high-resolution solution trajectories while solving system PDEs at lower

resolutions. The ML part of the solver extracted spatial features by using u-net as the model structure to predict the error accumulated in the short time interval between the evolution of the coarse grid and the solution of the system at a higher resolution. The predicted error can optimize the solution generated by the coarse grid to obtain a solution close to that generated by the fine grid, enabling high-precision solutions at low accuracy. Y. Bar-Sinai Bar-Sinai et al. (2019) designed a data-driven discretization scheme using a deep-learning algorithm. They used neural networks to estimate spatial derivatives that were optimized end-to-end to best satisfy equations on low-resolution grids. The resulting numerical method was very accurate, eventually achieving the same computational accuracy as the standard finite difference method at 4 to 8 times coarser resolution than the standard finite difference method. Zhuang [38] improved the model structure and loss function based on Y. Bar-Sinai and applied it to passive scalar advection in a two-dimensional unsteady flow. They used a convolutional neural network to learn spatial discretization coefficients to calculate spatial derivatives. Then, they combined them with traditional numerical methods to calculate time derivatives to obtain the numerical solution of partial differential equations. This method achieved a high-precision solution with a low computational cost. Ranade et al., 2021 developed DiscretizationNet, a machine learning-based PDE solver that combined essential features of existing PDE solvers with ML techniques. They used a discretization-based scheme to approximate spatiotemporal partial derivatives and a CNN-based generative encoder-decoder model with PDE variables as input and output features for iteratively generating equation solutions. Although these methods addressed the problem of traditional methods, their solution accuracy was limited due to the problems of ignoring spatiotemporal characteristics and input redundancy.

3 Proposed method

3.1 Problem description

If the velocity field is divergence-free, the advective form of the scalar concentration field $C(\vec{x}, t)$ for a given velocity field $\vec{u}(\vec{x}, t)$ is as follows Zhuang et al. (2021):

$$\frac{\partial C}{\partial t} + \vec{u} \cdot \nabla C = 0 \tag{4}$$

The objective of the numerical solution for the passive scalar advection in 2-D unsteady flow is to predict the concentration field distribution at each time step in the future under the influence of the randomly changing velocity field given the initial concentration field. In this paper, we predict the concentration field distribution results in the 32 time steps to demonstrate the ability of our model to make multi-step predictions. We employ a rolling forecasting scheme in which we input multiple velocity fields between t_0 and t_1 into the prediction model and combine the concentration field distribution at t_0 to predict the concentration field distribution at t_1 .Then, we input multiple velocity fields between t_1 and t_2 into the model and combine the concentration field distribution at t_1, predicted by our model to predict the concentration field distribution at t_2. According to this calculation rule, we use multiple velocity fields between t_n and t_{n+1} and the concentration field distribution predicted at time t_n to predict the concentration field distribution at t_{n+1}.By repeating this process, we can get the passive scalar advection solution at each time in the future. Therefore, the key to our multi-step prediction method is to recursively predict the concentration field distribution at a single step, i.e., the numerical solution of passive scalar advection at the next time step. We propose the time-sequence-involved space discretization neural network (TSI-SD) to predict the space discretization coefficient for the space derivative and then combine the finite volume method to calculate the numerical solution of the next time step.

3.2 Main framework of TSI-SD

The framework of the proposed method is shown in Figure 2. This is a fusion framework of deep learning (TSI-SD) and a traditional numerical method (FVM) for end-to-end numerical solutions of passive scalar advection equations. It consists of three modules: the spatial discretization coefficient prediction module (SDCPM), the concentration template extraction module (CTEM), and the concentration solver module based on finite volume numerical format (CSM). For the set of multiple velocity fields between the time steps t_n and t_{n+1}, wedecompose each velocity field into two sub-velocity fields in the horizontal and vertical directions (along the x-axis and y-axis) to obtain the velocity field set in the two directions. In the next step, we build the time-sequence-involved space discretization neural network (TSI-SD) in the SDCPM. TSI-SD extracts the spatiotemporal features from the decomposed velocity field sets in the two directions separately and then fuses them to obtain the spatial discretization coefficient of each grid point. After that, we input the coefficients into the CSM and combine them with the surrounding point concentration template of each grid point obtained by the CTEM to calculate the spatial derivative. Finally, we could calculate the concentration of each grid point at the next moment t_{n+1},, that is, the concentration field of t_{n+1} by the FVM in the CSM.

The equation-solving process can be roughly described in the following three steps:

1. Extract spatiotemporal features from the input velocity fields and predict spatial discretization coefficients;
2. Extract the surrounding point concentration template for each grid point; and
3. Fuse the predicted spatial discretization coefficient and the concentration template to obtain the spatial derivative, which is used to calculate the distribution of the concentration field, i.e., the numerical solution of the equation at the next time step by the finite volume method.

Next, we will provide details of our proposed framework for end-to-end numerical solutions of passive scalar advection equations. First, we introduce the SDCPM and the TSI-SD in the

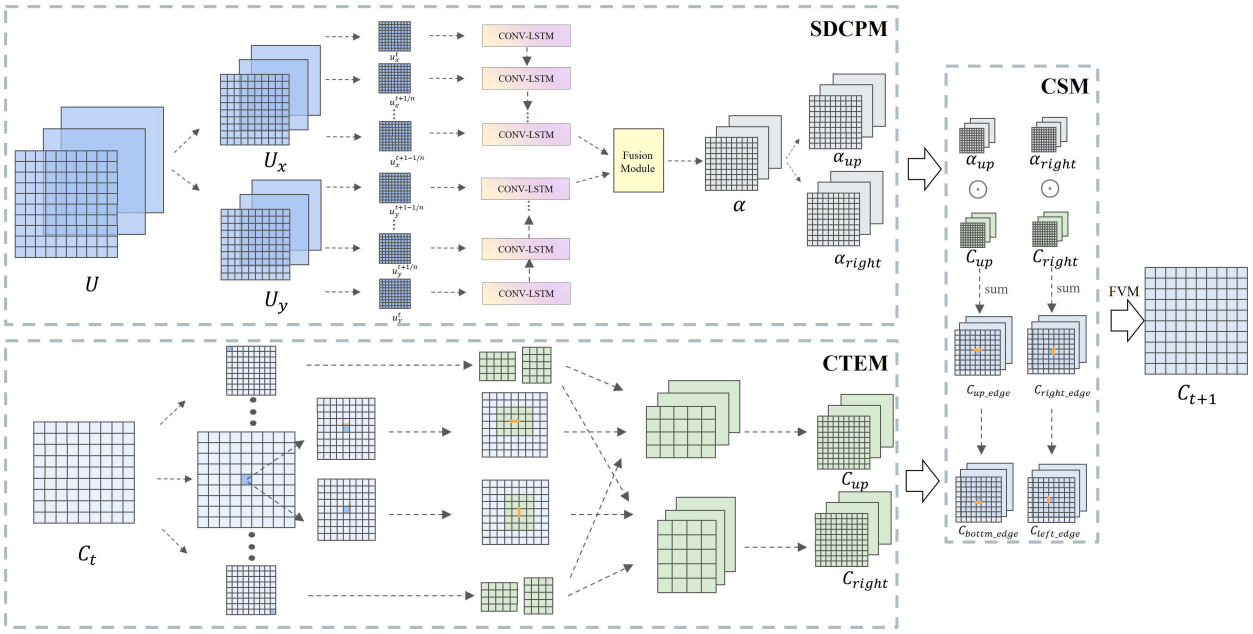


FIGURE 2
The framework of our approach. This framework is utilized for the solution of the passive scalar advection equation in a two-dimensional unsteady flow. It contains three modules: SDCPM, CTEM and CSM. SDCPM: This module receives multiple velocity field information at different times, extracts spatiotemporal features in two spatial dimensions (along the x-axis and y-axis), and finally fuses them in the spatial dimension to predict the spatial discretization coefficient of each grid point; CTEM: This module extracts the surrounding point concentration template of each grid point corresponding to the size of its spatial discretization coefficient template; CSM: This module calculates numerical solutions to the advection equations based on the finite volume method(FVM).

section entitled 'Spatial Discretization Coefficient Prediction Module'. Then, we describe in detail our proposed CTEM and CSM modules in the Concentration Template Extraction Module' and 'Concentration Solver Module Based on Finite Volume Numerical Format' sections, respectively. Finally, we discuss the loss function in the 'Loss Function' section 3.6

3.3 Spatial discretization coefficient prediction module

In this module, we design the time-sequence-involved space discretization neural network to predict the spatial discretization coefficients, and the prediction function is

$$\vec{\alpha} = f(U, W), \tag{5}$$

where U s the set of multiple two-dimensional velocity fields between t nd t+1, of which size is nW s the weight of our neural network. The time interval between the velocity fields is $\frac{1}{n}$

$$U = \left\{u_t, u_{t+\frac{1}{n}}, \ldots, u_{t+1-\frac{1}{n}}\right\} \tag{6}$$

We decompose U into velocity field groups U_x in the horizontal direction (along the x-axis) and U_y in the vertical direction (along the y-axis),

$$U_x = \left\{u_t^x, u_{t+\frac{1}{n}}^x, \ldots, u_{t+1-\frac{1}{n}}^x\right\} \tag{7}$$

$$U_y = \left\{u_t^y, u_{t+\frac{1}{n}}^y, \ldots, u_{t+1-\frac{1}{n}}^y\right\} \tag{8}$$

Then, we extract the spatiotemporal features separately for the decomposed velocity field sets in the two directions. Taking U_x as an example, we input the velocity field at each time step from the velocity field set $\{u_t^x, u_{t+\frac{1}{n}}^x, \ldots, u_{t+1-\frac{1}{n}}^x\}$as the spatial feature of each time step into the different conv-lstm structural unit,

$$S_k = u_k^x, \tag{9}$$

and S_k is regarded as the spatial feature at time $t + \frac{1}{k}$. A CONV-LSTM structural unit contains convolution operations and long-short-term memory unit processing operations. The calculation steps can be written in the following form,

$$i_k = Sigmoid(Conv(S_k; w_{xi}) + Conv(h_{k-1}; w_{hi}) + b_i) \tag{10}$$

$$f_k = Sigmoid(Conv(S_k; w_{xf}) + Conv(h_{k-1}; w_{hf}) + b_f) \tag{11}$$

$$g_k = Tanh(Conv(S_k; w_{xg}) + Conv(h_{k-1}; w_{hg}) + b_g) \tag{12}$$

$$c_k = f_k \odot c_{k-1} + i_k \odot g_k = f_k \cdot c_{k-1} + i_k \odot g_k \tag{13}$$

$$o_k = Sigmoid(Conv(S_k; w_{xo}) + Conv(h_{k-1}; w_{ho}) + b_o) \tag{14}$$

$$h_k = o_k \odot Tanh(c_k) \tag{15}$$

where i_k is the input gate, which is used to calculate how much information of the current state to retain. f_x is the forget gate, and its

function is to calculate how much of the output information of the previous moment is discarded. g_x is theinformation extracted from the current state. C_{k-1} is the information of the previous moment. C_k is the final state at the current moment, calculated by f_k, c_{k-1}, g_k, and i_k. $o_k i$ the output gate, which is used to calculate how much information needs to be output (to the cell at the next moment). h_k is the final output information of the state, which is calculated by o_k and c_k. w_{xi},w_{hi},w_{xf},w_{hf},w_{xg},w_{hg},w_{xo},w_{ho},b_i,b_f,b_g,*and* b_o are the weights designed in our neural network, and these weights will be updated during the model training process.

After the information processing and transmission of n conv-lstm structural units, the information $h_{t+1-\frac{1}{n}}$ output by the last unit is obtained. The final spatiotemporal fusion information I_x of the horizontal velocity field is calculated by using the output information.

$$I_x = conv(h_{t+1-\frac{1}{n}}) \tag{16}$$

In the same way, we obtain the final spatiotemporal fusion information I_y f the vertical velocity field.

After obtaining the spatiotemporal fusion information I_x and I_y in two directions, it is necessary to re-fuse the spatiotemporal features in the horizontal and vertical directions on I_x and I_y *concat*(), is a feature merging operation that integrates two features in a new dimension. After the feature merging operation, convolution is performed on the merged features to process the spatial information of the merged spatiotemporal features. Finally, the spatial discretization coefficient matrix α is obtained.

$$\alpha = conv(concat(I_x, I_y)) \tag{17}$$

The dimension of the α matrix is (s,s,*template_size**2) , where s is the side length of the input two-dimensional velocity field, and (s, s) is the dimension of the two-dimensional velocity field. *template_size* is the number of weights required for each grid point. We divide α into the grid upper boundary space discretization coefficient α_{up} and the grid right boundary space discretization coefficient α_{right} with dimensions (s, s, *template_size*).

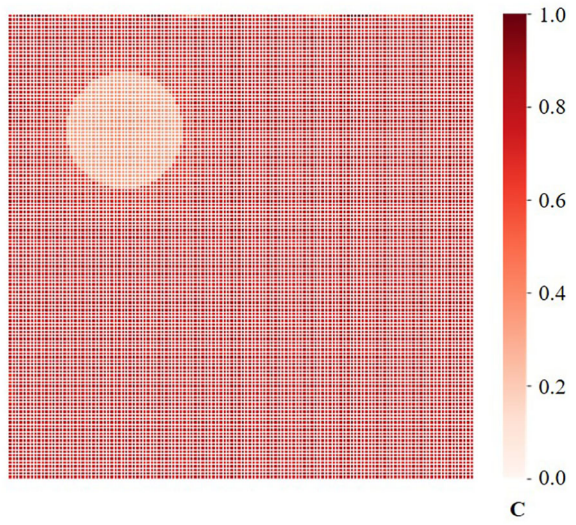


FIGURE 3
Design of the initial concentration field. The size of the concentration field is [0,1]×[0,1], and the concentration value is between 0 and 1.

3.4 Concentration template extraction module

This module and the next module follow the numerical solution part of the traditional advection equation adopted by Zhuang et al. Zhuang et al. (2021), and adopt the spatial derivative of the classical Euler algorithm.

$$\frac{\partial C}{\partial x}|_{x=x_i} = \sum_{j=0}^{n} \alpha_j C_{i+j} \tag{18}$$

In the previous part, we calculated the spatial discretization coefficient templates α_{up} and α_{right}, the dimensions of which are (s, s, *template_size*). Therefore, we need to find the surrounding grid point concentration templates C_{up} and C_{right} corresponding to the position of the coefficient template, the dimensions of which are both (s, s, *template_size*), which indicates that the number of surrounding grid point concentrations required for each point in the two-dimensional space field is *template_size*. As shown in Figure 3, we input the two-dimensional concentration field C_t at time t, and its dimension is (s, s). We model the upper and right boundaries of each point in the two-dimensional matrix and obtain the concentration values of m*n grid points around it as the grid point concentration template, where

$$template_size = m \star n, \tag{19}$$

m and n are the length and width of the two-dimensional grid point concentration template. Finally, we obtain C_{up} and C_{right} with dimensions (s, s, *template_size*).

3.5 Concentration solver module based on finite volume numerical format

In this module, we first calculate the upper boundary concentration C_{up_edge} and the right boundary concentration C_{up_edge}

$$C_{up_edge} = SUM(\alpha_{up} \odot C_{up}) \tag{20}$$

$$C_{right_edge} = SUM(\alpha_{right} \odot C_{right}) \tag{21}$$

SUM() is the defined summation of the last dimension of the matrix, i.e., after the matrix of (s, s, *template_size*) is obtained through the dot product operation, the last dimension is summed to obtain the boundary concentration C_{edge} with dimension size (s, s) .The lower boundary concentration C_{lower_edge} and the left boundary concentration C_{left_edge} of the grid point can be directly obtained from the upper boundary concentration of the adjacent grid below its position and the right boundary concentration of the adjacent grid to the left of its position. Then, we can obtain the boundary velocity u_{edge} bythe same method as the calculation of the

concentration boundary and boundary flux *via* C_{edge} u_{edge}. After obtaining the flux at the four boundaries of the grid, the traditional finite volume method is used to calculate the time derivative to obtain the concentration field distribution at the next time step, as shown in Figure 2.

3.6 Loss function

The format of the mean absolute error (MAE) used to train our model is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | \hat{C}_{t+1} - C_{t+1} | \tag{22}$$

where C_{t+1} is the concentration field at time t+1 redicted by our model, and $\hat{C}_{t+1}$ is the high-precision numerical solution at 16*16 low resolution grids. The numerical solution is calculated using 128*128 high resolution grids by the second-order Vanleer format and then transformed to the solution at 16*16 low resolution grids by the dimensionality reduction method Zhuang et al. (2021).

4 Experiments

In this section, we first briefly describe the datasets and implementation details. Additionally, we carry out a number of experiments, including comparisons with state-of-the-art (SOTA) methods and sufficient ablation studies, to demonstrate the excellent performance and advantages of our method.

4.1 Datasets

We used the theory of divergence-free velocity field described by Saad and Sutherland (2016) to generate a divergence-free random velocity field set with the resolution of 128*128. Then, the set was divided into two parts of divergence-free random velocity field sets: the training part and the test part. These two parts were completely different to ensure the generalization of the model.

For the training set, we generated a variety of random initial concentration fields and used the second-order Vanleer numerical format to calculate the numerical solution of the equation, i.e., the concentration fields at multiple time steps with the resolution of 128*128 based on the set of divergence-free random velocity fields in the training part. It is worth noting that for the $C_{t+1)}$ to be generated, our model needs to input the velocity field i at time $t, t + \frac{1}{n}, \ldots, t + 1 - \frac{1}{n}$. Therefore, we set the time step length of the velocity field to be smaller than the concentration field in the generation process to ensure that the velocity field in the time interval from t to t+1 could be generated. Next, we sampled both the velocity field and the concentration field at intervals to obtain a high-precision velocity field and concentration field with a resolution of 16*16 using the dimensionality reduction method Zhuang et al. (2021). Each training sample included an input part and an output part. The input part was the velocity field and the concentration field C_t at time t at multiple time steps in the time interval from t to t+1,and the output part was the concentration field C_{t+1}. The test set generation process was consistent with the training set, but it was necessary to ensure that the random initial concentration field generated in the test set was different from the training set.

The initial and boundary conditions for the velocity and concentration fields were set as follows. The size of the two-dimensional velocity field and the two-dimensional concentration field were both [0,1] × [0,1]. Our velocity field was a divergence-free random velocity field, and the magnitude of the velocity was limited between -1 and 1. The concentration field used periodic boundary conditions, and its initial condition is to

set the concentration value range between 0 and 1. The calculation process is shown in formulas (23)-(27).

$$r(x,y) = min(1, 4*\sqrt{(x-\frac{1}{4})^2 + (y-\frac{1}{4})^2}) \tag{23}$$

$$C_1(x,y) = \frac{1}{2}[1 + cos(\pi r)] \tag{24}$$

$$C_2(x,y) = 0.9 - 0.8*C_1^2 \tag{25}$$

$$C_3(x,y) = 1 \tag{26}$$

$$C(x,y) = 1 - 0.3*(C_1 + C_2 + C_3) \tag{27}$$

The $C(x, y)$ is as shown in Figure 3. C represents the concentration value.

4.2 Comparison with SOTA methods

In this part, four SOTA numerical solution methods for passive scalar advection in a two-dimensional unsteady flow were selected as our baseline: (1) traditional solvers based on 16*16 resolution grids using the second-order Vanleer discretization format(Vanleer 16*16) Lin et al. (1994); (2) traditional solvers based on 32*32 resolution grids using the second-order Vanleer discretization format (Vanleer 32*32) Lin et al. (1994); (3) traditional solvers based on 64*64 resolution grids using the second-order Vanleer discretization format (Vanleer 64*64) Lin et al. (1994); and (4) a hybrid solver based on a CNN and the finite volume method (CNN +FVM) Zhuang et al. (2021).

We first compared our TSI-SD method with traditional solvers, in which TSI-SD uses a 16*16 low-resolution grid. As shown in Figure 4, the TSI-SD method maintained the smallest prediction error over 32 time steps, which demonstrates that our method achieved a higher solution accuracy than the traditional method at a resolution of 4× lower than the traditional method.

Then, we compared TSI-SD with the CNN-FVM solver trained based on the previous spatial discretization scheme Zhuang et al. (2021). The CNN-FVM method is currently one of the most outstanding methods for solving partial differential equations in deep learning. It has been proven to achieve very good prediction

and solution results in various partial differential equations, such as Burgers' equation Bar-Sinai et al. (2019), and the advection equation Zhuang et al. (2021), Additionally, the method has been proven effective at solving complex Navier–Stokes equations Kochkov et al. (2021),and results are as accurate as baseline solvers, with 8–10× finer resolution in each spatial dimension, resulting in 40- to 80-fold computational speedups. The original CNN-FVM solver has a prediction error of 0.0043, a single-step prediction time of 0.2712s, and a single-sample training time of 4 ms per round during the training process. Our single-step solver had an error of 0.0029, a single-step prediction time of 0.2474s, and a single-sample training time of 2ms per round during training. Our single-step error was 32.56% lower than the previous method, and the iterative prediction error after 32 steps was greatly reduced. As shown in Figure 5, our method also outperformed the CNN-FVM solver in continuous prediction results within 32 time steps.

The reason why our solver outperformed the CNN-FVM solver in training time, prediction time, and prediction accuracy is as follows. In the spatial discretization coefficent prediction part, the inputs of the CNN solver's prediction deep-learning model are the concentration field with (*batch_size,1,grid,_size,grid_size*) and the two velocity field (along the x-axis and y-axis) at a time step with (*batch_size,2,grid,_size,grid_size*). The input to our TSI-SD was the horizontal velocity fields along the x-axis at two time steps with (*batch_size,2,grid,_size,grid_size*) and the vertical velocity fields along the y-axis at two time steps with (*batch_size,2,grid,_size, grid_size*), so our input size was larger than the previous input size. However, in the model part, the CNN-FVM solver used a five-layer convolutional neural network to process the data collected by the concentration field and the velocity field with (*batch_size,3,grid, _size,grid_size*) .We used the structure of a 1-layer

convolutional neural network to process the horizontal and vertical velocity fields respectively, and then a one-layer convolutional neural network was used to process the integrated features. After inference analysis, our model parameters were fewer than the original model parameters, which resulted in a shorter training time and prediction time in our model compared with the original model training time. This was also confirmed by a saved parameter file size comparison.

Finally, we demonstrated the evolution prediction effect of an initial concentration field under different models after 32 iterations. As shown in Figure 6, the third row shows the prediction effect of our model. The first row is our high-precision numerical solution generated using a 128*128 high-resolution grid. The second row shows how we use the averaging operation to obtain a high-precision numerical solution at a low resolution of 16*16, which is used as the ground truth of our model. The fourth and fifth rows are the results obtained using the second-order Vanleer 16*16 and CNN-FVM solvers. Figure 5 shows that our model is better than the CNN-FVM and traditional second-order Vanleer 16*16 solvers. In Figure 6, *C* represents the concentration value.

4.3 Comparison between models using velocity fields at different times as spatiotemporal features

In this part, we used different sets of time steps as the time series information input to TSI-SD, so that our model could extract different time features to predict the spatial discretization coefficient. The best prediction result represents the velocity fields at the selected time steps that have the greatest influence on the coefficients. Figure 7 shows that when the set of fine velocity fieldd $\{u_t, u_{t+\frac{1}{n}}, \ldots, u_{t+1-\frac{1}{n}}\}$ was selected to replace velocity field set $\{u_{t-n+1}, \ldots, u_{t-1}, u_t\}$ to predict u_{t+1} could reduce the prediction error of the model. The experimental result demonstrates that the set of fine velocity fields extracts spatiotemporal features more effectively. That is because the time interval of the velocity field set we chose was close to the time of the predicted concentration field, so the correlation between the velocity field set and the predicted

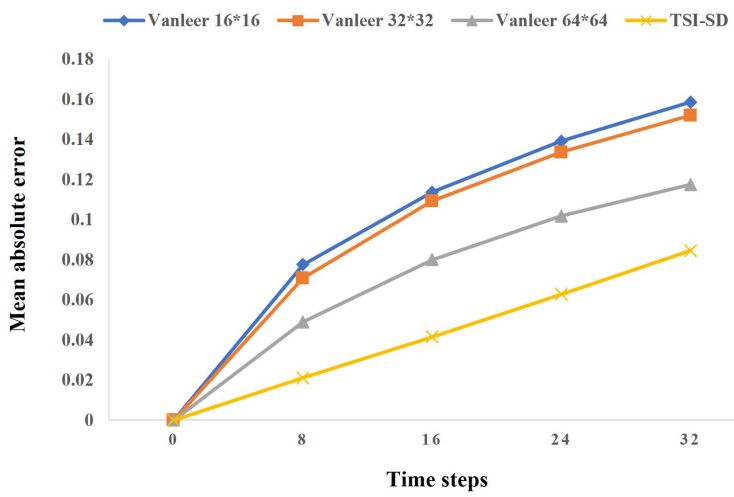


FIGURE 4
Results of our solver compared to traditional solvers. The yellow line represents our error in the 32-step iteration prediction, and the remaining three lines represent the error of the traditional solver at different resolutions.

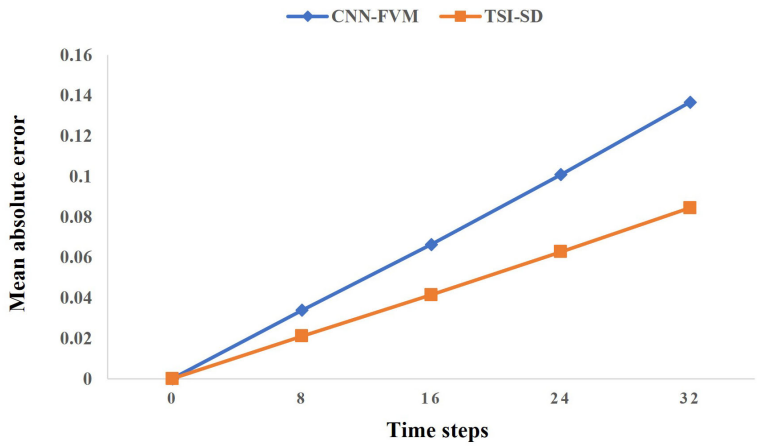


FIGURE 5
Results of our solver compared to CNN-FVM solvers by Zhuang Zhuang et al. (2021). The orange line represents our error in the 32-step iteration prediction, and the blue line represents the error of CNN-FVM solver.

concentration field was strong. The model could learn the spatiotemporal influence of the velocity field on the concentration field from this set of velocity fields, which could accurately predict the spatial discretization coefficient. Meanwhile, the prediction error of the velocity field using $\{u_t, u_{t+\frac{1}{2}}\}$ is the best, and experiments demonstrated that it involves lower computational cost; therefore, so we finally choose the velocity field of $\{u_t, u_{t+\frac{1}{2}}\}$ as the velocity field input of our final model. We think that for the 16*16 lower resolution grid, the model learned the time-space correlation between the velocity field set and the concentration field well through the analysis of the velocity fields at two times through a large amount of training data, which is also consistent with the experimental results as shown. In future studies, we will conduct more experiments on higher-resolution

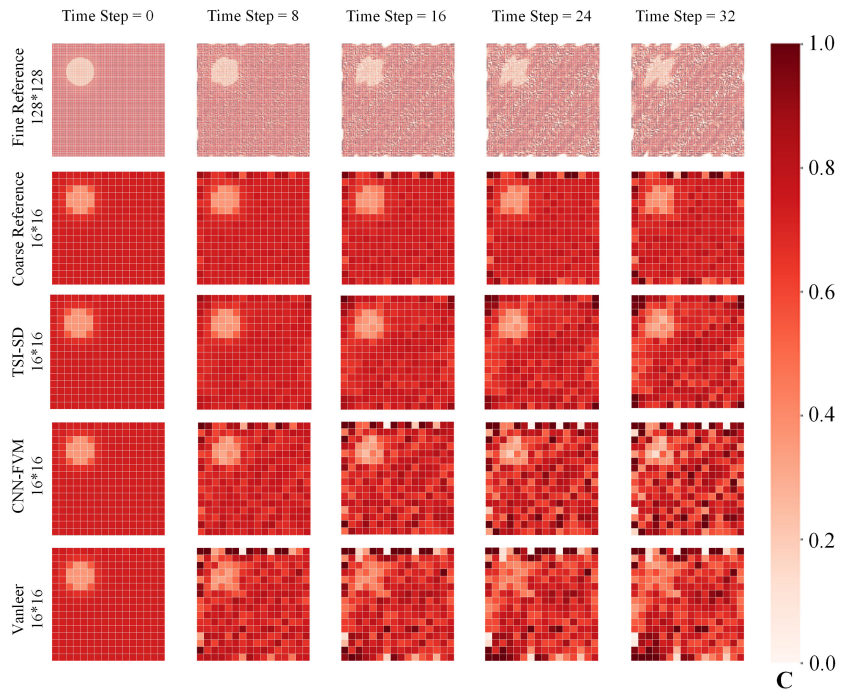


FIGURE 6
Visualization of evolution prediction effect of an initial concentration field under different models after 32 iterations. The first row represents the change in the concentration field calculated after 32 steps using a traditional 128*128 high-resolution solver. The second row represents the transformation of the 128x128 high-resolution solver solution into a 16x16 training set. The third row represents the prediction results of our model after training. The fourth row represents the prediction results of the CNN-FVM solver. The fifth row uses a traditional 16*16 low-resolution solver to calculate the change in the concentration field after 32 steps.

grids to obtain the optimal number of time steps after increasing computing power.

4.4 Performance and analysis of TSI-SD with other flow fields

In this section, we carried out an experiment to prove the excellent performance of our model under a constant velocity field and a two-dimensional deforming flow velocity field. We generated the concentration under a constant velocity field, and the two-dimensional deformation flow concentration field under the velocity field:

$$u(x,y,t) = sin^2(\pi x)sin(2\pi y)cos(\frac{\pi t}{T}) \tag{28}$$

$$v(x,y,t) = sin^2(\pi y)sin(2\pi x)cos(\frac{\pi t}{T}) \tag{29}$$

The predicted performance is shown in Figure 8 and Figure 9. *C* represents the concentration value. Our model achieved outstanding prediction results in the iterations of 32 time steps. However, at the same time,

there are also the following problems: even under a simple constant velocity field, the prediction effect will become worse and worse with the long-term iteration due to the accumulation of errors predicted by the model at each time step. We will try to fix this in the future.

4.5 Comparison of the performance of models with or without the concentration field as an input feature

In this part, we verified the advantage of only taking the velocity field as the input feature on our model. A contrast model that adds the concentration field as feature input was designed to prove our inference. The contrast model was identical to ours except that the concentration field features were fused with the spatiotemporal features extracted from the horizontal and vertical velocity fields (along the x-axis and y-axis) in the fusion module. Figure 10 shows that the iteration errors on 32 time steps of our model are lower than those of the contrast model. Therefore, we proved that the input of the concentration field information was redundant and verified our conclusion: the spatial discretization coefficients are strongly correlated with the velocity field at multiple time steps before, while the concentration field information becomes redundant when predicting the coefficients. In other words, the change in the velocity field is the main factor for the change in the concentration field. Our model extracts effective spatiotemporal features from the velocity field set to learn the influence of the change of the velocity field set on the change of the concentration field, which is very helpful for predicting the spatial discretization coefficient.

4.6 Experimental exploration of whether TSI-SD has up-wind properties

In this part, we proved that the spatial discretization coefficients predicted by our model have upwind properties on a constant velocity field. A two-dimensional velocity field U_1 with a horizontal velocity field (along the x-axis) of +1 and a vertical velocity field (along the y-axis) of +1, and a two-dimensional velocity field U_2 with a horizontal velocity field of -1 and a vertical velocity field of -1, were designed to prove our model's upwind properties on a constant velocity field. Under the two velocity fields, the visualization process of the concentration coefficients of the upper and right boundaries of grid points A and B was completed.

As shown in Figure 11, *C* represents the concentration value and *Coefficient* represents the coefficient value. For the upper boundary, the concentration on the right boundary of the constant velocity field is mainly determined by the concentration of the two adjacent grids. When the horizontal speed is +1 (i.e., the direction is to the right), the grid coefficient on the left of the right boundary of grid A is greater

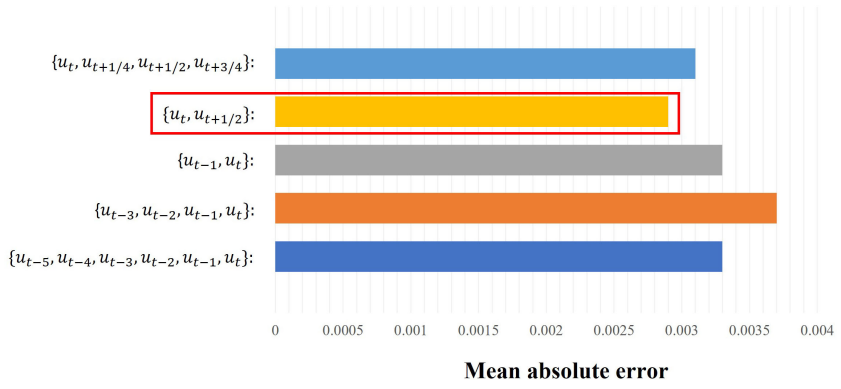


FIGURE 7
Mean absolute error comparison of the prediction results of models using different sets of time steps as the time series information input to TSI-SD. For the y-axis, the different colors represent different input sets of time steps. The legend represents mean absolute error and the red box shows the best result ($\{u_t, u_{t+\frac{1}{2}}\}$).

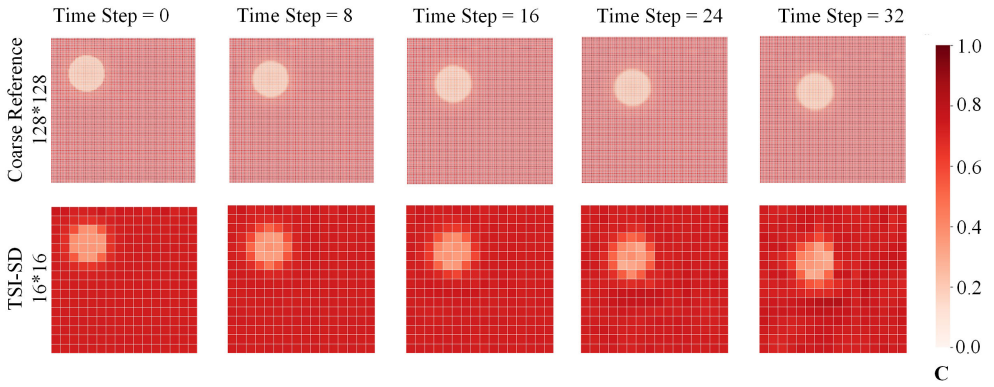


FIGURE 8
The predicted performance of our model in a constant velocity field of v_x=1 and v_y=1. The first row is the iterative solution of our 128*128 high-resolution solver after 32 time steps, and the second row is the iterative solution of our 16*16 solver after 32 time steps.

than the grid coefficient on the right; when the horizontal speed is −1 (i.e., the direction is to the left), the grid coefficient on the left of the right boundary of grid A is smaller than the grid coefficient on the right. For the right boundary, the concentration on the upper boundary of the constant velocity field is also mainly determined by the concentration of the two adjacent grids. When the vertical speed is +1 (i.e., the direction is downward), the grid coefficient above the lower boundary of grid A is greater than the grid coefficient below; when the horizontal speed is −1 (i.e., the direction is upward), the grid coefficient above the lower boundary of grid A is smaller than the grid coefficient below.

The concentration coefficient of another spatial grid point B is almost the same as that exhibited by A. Therefore, our grid coefficient has nothing to do with the distribution of the concentration field, but only with the distribution of the velocity field. The concentration field distributions at point A and point B are completely inconsistent, but under the same velocity field, the predicted spatial discretization coefficient distributions are basically the same, which proves that there is no significant correlation between the concentration field distribution and the spatial discretization coefficient.

5 Conclusion

We have presented a time-sequence-involved space discretization neural network of passive scalar advection in a two-dimensional unsteady flow. It can obtain adaptive spatial discretization derivatives according to the spatiotemporal property of the current environment. Then, we combined it with the finite volume method to form an advection equation solver that can calculate high-resolution solutions on low-resolution grids.

The highlight of our approach is the transformation of a novel deep neural network from the classic CONV-LSTM backbone. The network resolves spatiotemporal features by adding temporal information to a two-dimensional spatial grid along the x- and y-axes, and then fuses them through a post-fusion neural network. Through spatiotemporal feature fusion, we can predict more

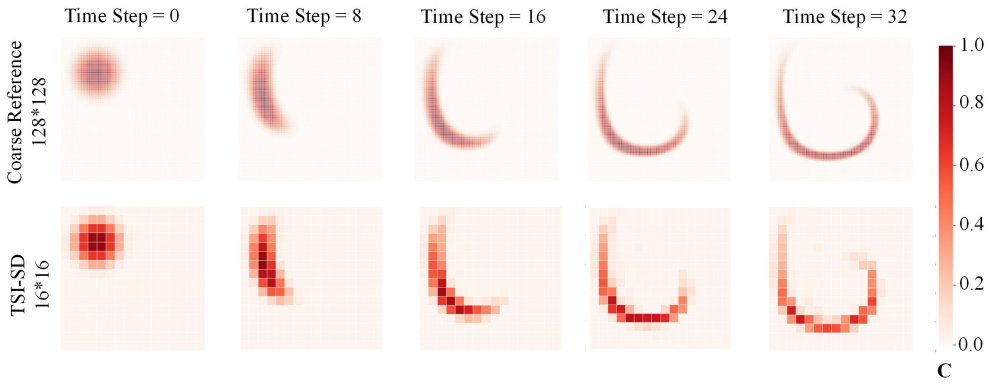


FIGURE 9
The predicted performance of our model in a deforming flow velocity field. The first row is the iterative solution of our 128*128 high-resolution solver after 32 time steps, and the second row is the iterative solution of our 16*16 solver after 32 time steps.

10.3389/fmars.2023.1132640

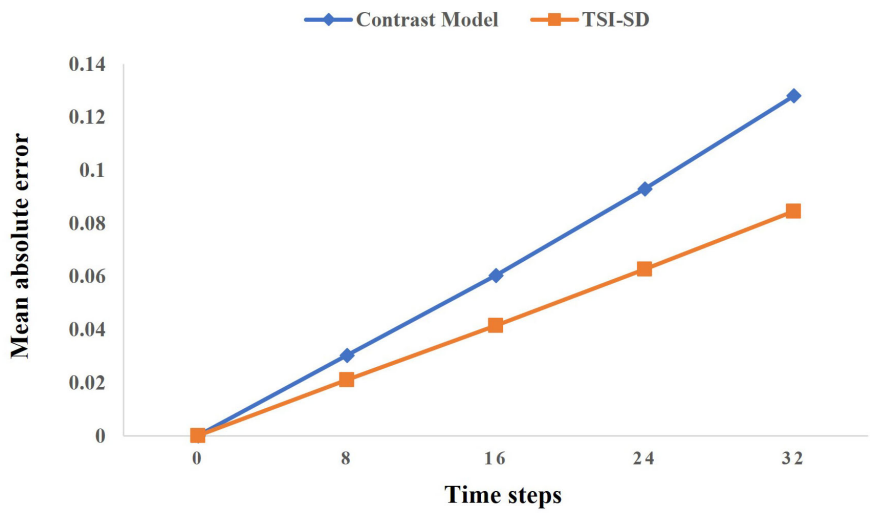


FIGURE 10
Results of our solver compared to the solver of adding concentration field as the model input. The orange line represents our error in the 32-step iteration prediction, and the blue line represents the error of the contrast model.

accurate spatial discretization coefficients and more accurate solutions. Additionally, we have made improvements in reducing computational costs. Finally, we compared our method with other traditional SOTA methods and demonstrated that it achieves better accuracy than traditional solvers on meshes with 4× lower resolution. In addition, compared with other deep-learning methods, our method has advantages in terms of both computational cost and accuracy.

The following problems were also encountered: (1) the problem of iterative error being too big after multiple time steps—we have proposed some solutions, such as re-iteration with ground-truth values after iterating over some time steps, which will be

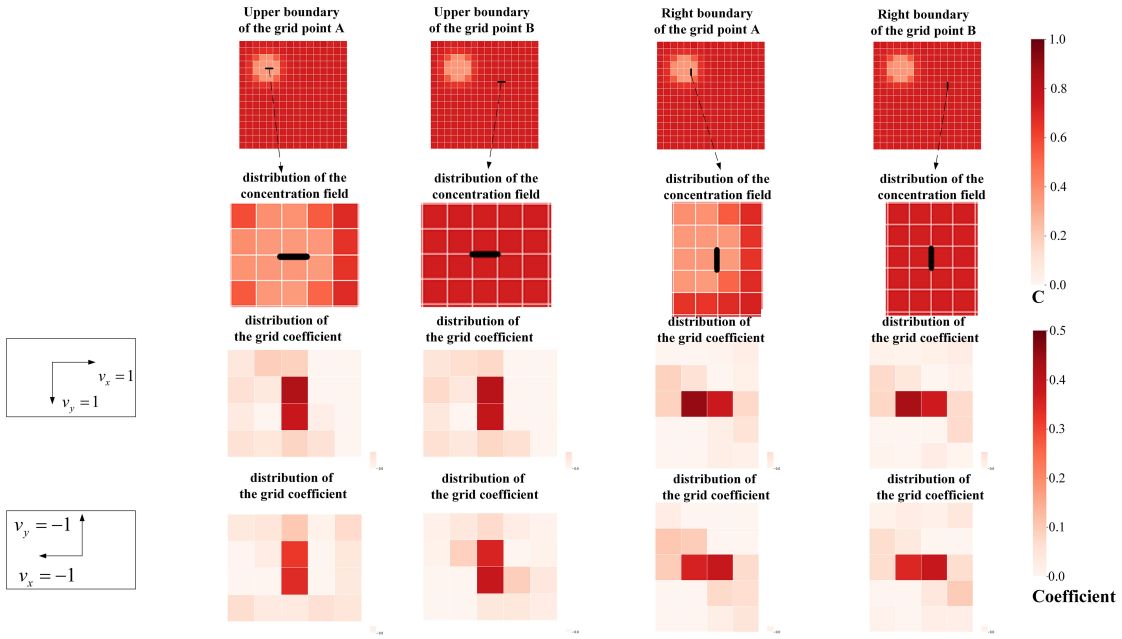


FIGURE 11
The comparison of prediction results of models using different temporal layers as features. The first line selects two spatial points with significant differences in surrounding concentrations from the spatial field and extracts the upper and right boundaries of the two points. The second row is the spatial discretization coefficient predicted by each boundary. The third row is a heat map made according to the different position coefficients in the coefficient template when the horizontal velocity field is +1, and the vertical velocity field is +1. The third row is a heat map made according to the different position coefficients in the coefficient template when the horizontal velocity field is -1, and the vertical velocity field is -1.

implemented in future work; and (2) low computing power leads to poor model generalization—in the future, we will seek to obtain more computing power to make our model more generalizable.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

NS carried out the methodology, data processing, modeling, and writing of the original draft. HT performed the conceptualization, validation, and review, and optimized the model framework. HG, JS, and YY performed the validation and investigation. ZW carried out the writing review. JN contributed to the conceptualization, writing review and editing, and supervision. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by the National Key Research and Development Program of China (2021YFF0704000), the National Natural Science Foundation of China (62172376), and Fundamental Research Funds for the Central Universities (202042008).

Acknowledgments

We thank two reviewers for their useful comments. We are very grateful to Prof. Song (Dehai Song, Key Laboratory of Physical Oceanography, Ocean University of China, Qingdao, China) for providing us with data and ideological support and help.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Bar-Sinai, Y., Hoyer, S., Hickey, J., and Brenner, M. P. (2019). Learning data-driven discretizations for partial differential equations. *Proc. Natl. Acad. Sci.* 116, 15344–15349. doi: 10.1073/pnas.1814058116

Berger, M. J., and Oliger, J. (1984). Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.* 53, 484–512. doi: 10.1016/0021-9991(84)90073-1

Bristeau, M., Pironneau, O., Glowinski, R., Periaux, J., Perrier, P., and Poirier, G. (1985). On the numerical solution of nonlinear problems in fluid dynamics by least squares and finite element methods (ii). application to transonic flow simulations. *Comput. Methods Appl. Mechan Eng.* 51, 363–394. doi: 10.1016/0045-7825(85)90039-8

Brown, J. (1982). "A multigrid mesh-embedding technique for three-dimensional transonicpotential flow analysis," in *20th aerospace sciences meeting* [Orlando, FL, U.S.A: American Institute for Aeronautics and Astronautics (AIAA)], 107. doi: 10.2514/6.1982-107.

Cai, S., Mao, Z., Wang, Z., Yin, M., and Karniadakis, G. E. (2022). Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechan Sin.* [Orlando, FL, U.S.A.: American Institute for Aeronautics and Astronautics (AIAA)], 1–12. doi: 10.1007/s10409-021-01148-1

Dwyer, H. A., Smooke, M. D., and Kee, R. J. (1982). *Adaptive gridding for finite difference solutions to heat and mass transfer problems* (Fort Belvoir, Virginia: California Univ Davis Dept Of Mechanical Engineering).

Eliasof, M., Haber, E., and Treister, E. (2021). Pde-Gcn: Novel architectures for graph neural networks motivated by partial differential equations. *Adv. Neural Inf. Process. Syst.* 34, 3836–3849. doi: 10.48550/arXiv.2108.01938

Ferziger, J. H., Perić, M., and Street, R. L. (2002). *Computational methods for fluid dynamics* (Cham, Switzerland: Springer), 3.

Fletcher, C. A. (2012). *Computational techniques for fluid dynamics: Specific techniques for different flow categories* (Springer-Verlag Berlin Heidelberg New York: Springer Science & Business Media).

Ji, W., Qiu, W., Shi, Z., Pan, S., and Deng, S. (2021). Stiff-pinn: Physics-informed neural network for stiff chemical kinetics. *J. Phys. Chem. A* 125, 8098–8106. doi: 10.1021/acs.jpca.1c05102

Jin, M., Liu, W., and Chen, Q. (2014). Accelerating fast fluid dynamics with a coarse-grid projection scheme. *HVAC&R Res.* 20, 932–943. doi: 10.1080/10789669.2014.960239

Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S. (2021). Machine learning–accelerated computational fluid dynamics. *Proc. Natl. Acad. Sci.* 118, e2101784118. doi: 10.1073/pnas.2101784118

Kraichnan, R. H. (1959). The structure of isotropic turbulence at very high reynolds numbers. *J. Fluid Mechan* 5, 497–543. doi: 10.1017/S0022112059000362

Lantz, R. (1971). Quantitative evaluation of numerical diffusion (truncation error). *Soc. Petroleum Engineers J.* 11, 315–320. doi: 10.2118/2811-PA

Leschziner, M. (1989). Modeling turbulent recirculating flows by finite-volume methods–current status and future directions. *Int. J. Heat Fluid Flow* 10, 186–202. doi: 10.1016/0142-727X(89)90038-6

Lin, S.-J., Chao, W. C., Sud, Y., and Walker, G. (1994). A class of the van leer-type transport schemes and its application to the moisture transport in a general circulation model. *Monthly Weather Rev.* 122, 1575–1593. doi: 10.1175/1520-0493(1994)122<1575:ACOTVL>2.0.CO;2

Lumley, J. L. (1979). Computational modeling of turbulent flows. *Adv. Appl. mechanics* 18, 123–176. doi: 10.1016/S0065-2156(08)70266-7

Mazhukin, V., Bobeth, M., and Semmler, U. (1993). *A dynamically adaptive grid method for solving one-dimensional non-stationary partial differential equations.* (Dresden, Germany: Max-Planck-Gesellschaft zur Förderung der Wissenschaften eV). 1–18.

Mikula, K., Ohlberger, M., and Urbán, J. (2014). Inflow-implicit/outflow-explicit finite volume methods for solving advection equations. *Appl. Numerical Math.* 85, 16–37. doi: 10.1016/j.apnum.2014.06.002

Molenkamp, C. R. (1968). Accuracy of finite-difference methods applied to the advection equation. *J. Appl. Meteorol. Climatol.* 7, 160–167. doi: 10.1175/1520-0450(1968)007<0160:AOFDMA>2.0.CO;2

Obiols-Sales, O., Vishnu, A., Malaya, N., and Chandramowliswharan, A. (2020). "Cfdnet: A deep learning-based accelerator for fluid simulations," in *Proceedings of the 34th ACM international conference on supercomputing* (Barcelona, Spain), 1–12. doi: 10.1145/3392717.3392772

Patel, R. G., Trask, N. A., Wood, M. A., and Cyr, E. C. (2021). A physics-informed operator regression framework for extracting data-driven continuum models. *Comput. Methods Appl. Mechan Eng.* 373, 113500. doi: 10.1016/j.cma.2020.113500

Pathak, J., Mustafa, M., Kashinath, K., Motheau, E., Kurth, T., and Day, M. (2020). Using machine learning to augment coarse-grid computational fluid dynamics simulations. *arXiv preprint arXiv:2010.00072*. doi: 10.48550/arXiv.2010.00072

Peyret, R., and Taylor, T. D. (2012). *Computational methods for fluid flow* (New York, USA: Springer Science & Business Media).

Phillips, R., and Schmidt, F. (1984). Multigrid techniques for the numerical solution of the diffusion equation. *Numerical Heat Transfer* 7, 251–268. doi: 10.1080/01495728408961824

Phillips, R., and Schmidt, F. (1985). Multigrid techniques for the solution of the passive scalar advection-diffusion equation. *Numerical heat transfer* 8, 25–43. doi: 10.1080/01495728508961840

Rai, M. M., and Moin, P. (1991). Direct simulations of turbulent flow using finite-difference schemes. *J. Comput. Phys.* 96, 15–53. doi: 10.1016/0021-9991(91)90264-L

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* 378, 686–707. doi: 10.1016/j.jcp.2018.10.045

Ranade, R., Hill, C., and Pathak, J. (2021). Discretizationnet: A machine-learning based solver for navier–stokes equations using finite volume discretization. *Comput. Methods Appl. Mechan Eng.* 378, 113722. doi: 10.1016/j.cma.2021.113722

RUMSEY, C., and VATSA, V. (1993). "A comparison of the predictive capabilities of several turbulence models using upwind and central-difference computer codes," in *31st aerospace sciences meeting* [Reno, NV, U.S.A.: American Institute for Aeronautics and Astronautics (AIAA)], 192.

Saad, T., and Sutherland, J. C. (2016). Comment on "diffusion by a random velocity field"[phys. fluids 13, 22 (1970)]. *Phys. Fluids* 28, 22. doi: 10.1063/1.4968528

Takayasu, A., Yoon, S., and Endo, Y. (2019). Rigorous numerical computations for 1d advection equations with variable coefficients. *Japan J. Ind. Appl. Math.* 36, 357–384. doi: 10.1007/s13160-019-00345-7

Toro, E. F. (2013). *Riemann Solvers and numerical methods for fluid dynamics: a practical introduction* (Springer-Verlag Berlin Heidelberg New York: Springer Science & Business Media).

Vadyala, S. R., Betgeri, S. N., and Betgeri, N. P. (2022). Physics-informed neural network method for solving one-dimensional advection equation using pytorch. *Array* 13, 100110. doi: 10.1016/j.array.2021.100110

Vinuesa, R., and Brunton, S. L. (2021). The potential of machine learning to enhance computational fluid dynamics. *arXiv preprint arXiv:2110.02085*. doi: 10.1038/s43588-022-00264-7

Zhang, J. (1997). *Multigrid acceleration techniques and applications to the numerical solution of partial differential equations* (The George Washington University).

Zhao, S., Zhou, J., Jing, C., and Li, L. (2019)Improved finite volume method for solving 1-d advection equation (IOP Publishing) (Accessed Journal of Physics: Conference Series).

Zhuang, J., Kochkov, D., Bar-Sinai, Y., Brenner, M. P., and Hoyer, S. (2021). Learned discretizations for passive scalar advection in a two-dimensional turbulent flow. *Phys. Rev. Fluids* 6, 064605. doi: 10.1103/PhysRevFluids.6.064605

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 27 February 2023
DOI 10.3389/fmars.2023.1129852

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Nikos Petrellis,
University of Peloponnese, Greece
Amaya Alvarez,
Mediterranean Institute for Advanced
Studies (CSIC), Spain

*CORRESPONDENCE
Ercan Avsar
erca@aqua.dtu.dk

SPECIALTY SECTION
This article was submitted to
Ocean Observation,
a section of the journal
Frontiers in Marine Science

RECEIVED 22 December 2022
ACCEPTED 14 February 2023
PUBLISHED 27 February 2023

CITATION
Avsar E, Feekings JP and Krag LA (2023)
Estimating catch rates in real time:
Development of a deep learning based
Nephrops (*Nephrops norvegicus*) counter
for demersal trawl fisheries.
Front. Mar. Sci. 10:1129852.
doi: 10.3389/fmars.2023.1129852

COPYRIGHT
© 2023 Avsar, Feekings and Krag. This is an
open-access article distributed under the
terms of the Creative Commons Attribution
License (CC BY). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that
the original publication in this journal is
cited, in accordance with accepted
academic practice. No use, distribution or
reproduction is permitted which does not
comply with these terms.

Estimating catch rates in real time: Development of a deep learning based *Nephrops* (*Nephrops norvegicus*) counter for demersal trawl fisheries

Ercan Avsar[1,2]*, Jordan P. Feekings[1] and Ludvig Ahm Krag[1]

[1]Technical University of Denmark, Institute of Aquatic Resources, Section for Fisheries Technology, Hirtshals, Denmark, [2]Computer Engineering Department, Dokuz Eylul University, Izmir, Türkiye

Demersal trawling is largely a blind process where information on catch rates and compositions is only available once the catch is taken onboard the vessel. Obtaining quantitative information on catch rates of target species while fishing can improve a fisheries economic and environmental performance as fishers would be able to use this information to make informed decisions during fishing. Despite there are real-time underwater monitoring systems developed for this purpose, the video data produced by these systems is not analyzed in near real-time. In other words, the user is expected to watch the video feed continuously to evaluate catch rates and composition. This is obviously a demanding process in which quantification of the fish counts will be of a qualitative nature. In this study, underwater footages collected using an in-trawl video recording system were processed to detect, track, and count the number of individuals of the target species, *Nephrops norvegicus*, entering the trawl in real-time. The detection was accomplished using a You Only Look Once v4 (YOLOv4) algorithm. Two other variants of the YOLOv4 algorithm (tiny and scaled) were included in the study to compare their effects on the accuracy of the subsequent steps and overall speed of the processing. SORT algorithm was used as the tracker and any *Nephrops* that cross the horizontal level at 4/5 of the frame height were counted as catch. The detection performance of the YOLOv4 model provided a mean average precision (mAP@50) value of 97.82%, which is higher than the other two variants. However, the average processing speed of the tiny model is the highest with 253.51 frames per second. A correct count rate of 80.73% was achieved by YOLOv4 when the total number of *Nephrops* are considered in all the test videos. In conclusion, this approach was successful in processing underwater images in real time to determine the catch rates of the target species. The approach has great potential to process multiple species simultaneously in order to provide quantitative information not only on the target species but also bycatch and unwanted species to provide a comprehensive picture of the catch composition.

KEYWORDS

demersal trawling, *Nephrops* counting, object detection, object tracking, sort, underwater video processing, YOLO

Introduction

Demersal trawling is an effective way of catching various species. However, usage of demersal trawls is challenged by several factors such as high bycatch rates and negative effects on the biomass and biodiversity (Eigaard et al., 2017). In addition, disturbance of the seabed by bottom trawls results in aqueous CO_2 emissions which may inhibit marine carbon cycling after years of continuous trawling (Sala et al., 2021). Despite the presence of such concerns, demersal trawling is critical for catching economically valuable commercial species like shrimp, whitefish, and *Nephrops*.

Nephrops excavate burrows in mud or mud/sand substrates and emerge at specific times to feed, mate and maintain their burrows, among others (Tully and Hillis, 1995; Aguzzi and Sardà, 2008; Feekings et al., 2015). Their behavior is influential on catch rates when trawling as they need to be outside of the burrows to be caught (Main and Sangster, 1985). Besides, *Nephrops*-directed bottom trawling is known to have high discard rate which eventually causes not only economic loss but also loss of undersized individuals (Bergmann et al., 2002). In addition to these issues, is demersal trawling a blind process, meaning that the catch and size composition is unknown until the trawl is taken onboard after hours of trawling.

Advancements in underwater camera technologies may provide solutions to some limitations in demersal trawling. In particular, such cameras allow for recognition, counting and measurement of the individuals making it possible to understand the catch rates of *Nephrops* and unwanted species. Even though there are different tasks such as species identification and length measurement (Underwood et al., 2014; Underwood et al., 2018; Allken et al., 2021), and segmentation of the fish from the background (Prados et al., 2017) accomplished using in-trawl camera systems, they do not concern determining the catch composition in real time. The real-time processing of video footage collected by underwater in-trawl cameras is important to quantify catch rates of the target species. This information is valuable for the fishermen as it provides insight about the ongoing fishing process and further enable active search for better catch rates during the fishing operation. Deep learning-based methods enable automated extraction of such information. In fisheries research, deep learning is mostly used for processing visual data collected either onboard or by using underwater cameras. However, the main issue related with deep learning methods is the substantiality of the associated computation amount which brings about drawbacks like latency in processing and requirement of hardware with sufficient computational capacity. To address this issue, various deep learning models with different sizes have been developed, and they can be applied to different problems. A review of related literature is provided in Section 2. There are deep learning-based methods available that are applicable to underwater videos collected by in-trawl cameras for real-time detection and counting of *Nephrops*. A fast and accurate video processing system in *Nephrops* fisheries is useful for generating the spatial distribution of catch items as well as determining the number of *Nephrops* caught.

In this study, a real-time processing pipeline for underwater videos to determine the number of *Nephrops* caught during demersal trawling is proposed as such information will provide a strong decision tool for fishers to optimize their catching operation. The processed video footages were collected by an in-trawl camera developed earlier (Sokolova et al., 2021b). The algorithm for *Nephrops* counting has three major steps that are *i*) *Nephrops* detection, *ii)* tracking of the detected *Nephrops*, and *iii*) determining the true tracks accounted for *Nephrops* catches. The accurate detection of *Nephrops* in the video frames is important as the subsequent steps rely on the detected *Nephrops*. The detection has been accomplished using You Only Look Once v4 (YOLOv4) model which is known to be a fast deep learning model for object detection operating at high frames-per-second (FPS) values. In addition, two variants of YOLOv4, namely, YOLOv4-Tiny and YOLOv4-Scaled are used separately for *Nephrops* detection, and their effects on the tracking, counting, and the overall processing speed are observed and compared. The second step, tracking detections, is necessary for making association between the detections in the consecutive video frames. Simple Online Realtime Tracking (SORT) algorithm is used as the object tracker. For benchmarking purposes, the tracking performance of SORT is compared with two other object tracking algorithms, those being Minimum Output Sum of Squared Error (MOSSE) and DeepSORT. Finally, tracked objects satisfying some predefined conditions are considered as a *Nephrops* catch. These steps are illustrated in Figure 1. In this study we address the following research questions:

- How do the different YOLO-based object detection methods affect the overall speed and accuracy of the counting process?
- What is the range of the processing speed of the proposed algorithm, and can it be considered as real-time under different circumstances?
- Is it possible to provide simple decision parameters for the fishers during trawling operation?
- What is the relation between the precision of the object detection and rate of correct *Nephrops* counts?

Related work

Utilization of deep learning methods in computer vision applications has become widespread in recent years due to their major advantage of automated feature extraction. However, the deep learning models typically possess many computational layers with high numbers of parameters. Performing all the calculations throughout all layers of the network takes time and hence the latency becomes an issue when the input data needs to be processed in real time.

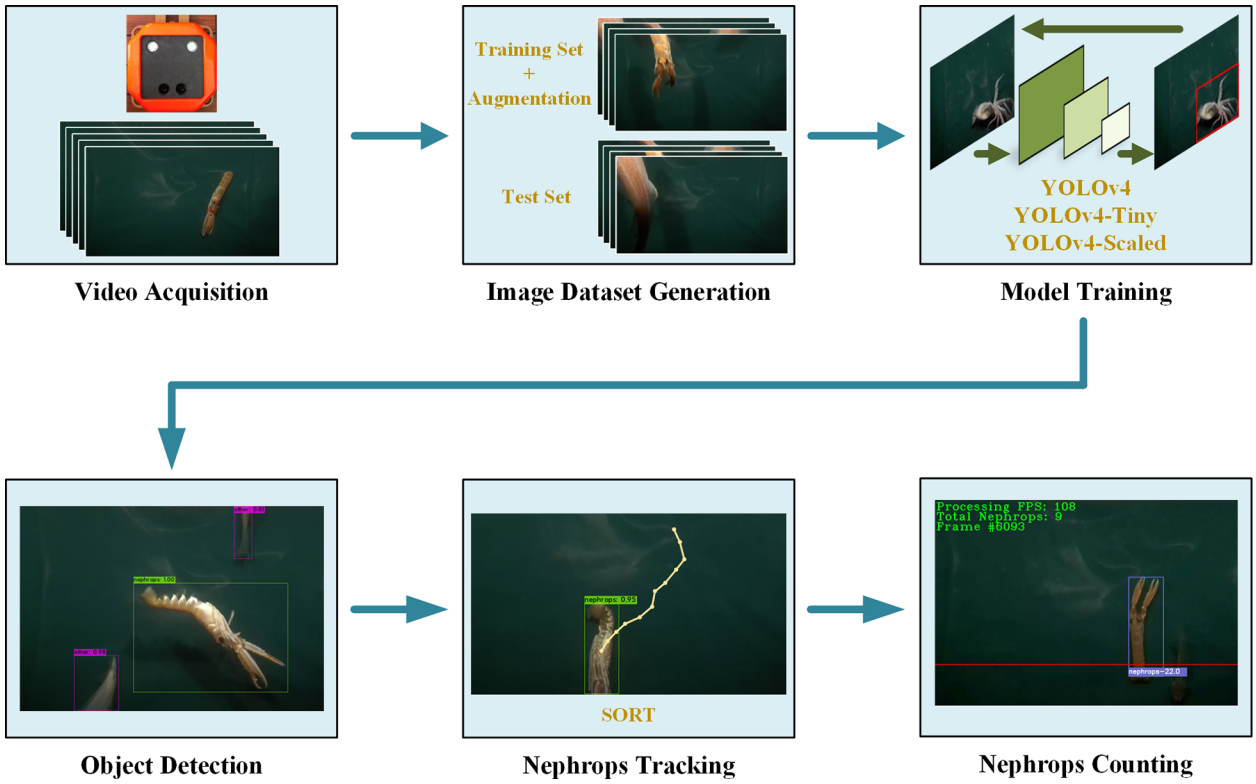


FIGURE 1
Overview of the algorithm steps.

Depending on the type of the problem (e.g. image classification, object detection, instance segmentation), there are various techniques to reduce the computational cost of the deep learning models while keeping the model performance as high as possible. For instance, MobileNets are efficient models developed to be used in hardware with limited computational resources (Howard et al., 2017) and can be used as a standalone classifier for animal classification in underwater images (Liu et al., 2019). Together with two other improved versions (Sandler et al., 2018; Howard et al., 2019) and single shot object detectors (SSD), they have more diverse applications such as detection of sea cucumbers (Yao et al., 2019), underwater objects with different scales (Zhang et al., 2021; Wang et al., 2022b), and *Nephrops* burrows (Naseer et al., 2020).

Another object detection method with many versions is YOLO, which is known for being very fast and accurate at the same time (Redmon et al., 2015). It can predict the bounding box coordinates and the corresponding confidence scores with one single network. There are numerous YOLO versions dedicated to operating on underwater images for detection of various objects such as starfish, shrimp, crab, scallop, and waterweed (Liu et al., 2020; Zhao et al., 2022). Among these models, the recently proposed model, YOLO-fish was designed for fish detection and is reported to be performing close to YOLOv4 model on two different public datasets (Muksit et al., 2022). Even though it is claimed to be a lightweight model the associated number of parameters and the detection time are between those of YOLOv3 and YOLOv4 (Muksit et al., 2022). In another study, an underwater imaging system to develop and test a lightweight YOLO model for automated fish behavior analysis was introduced (Hu et al., 2021). In that study, a modified version of YOLOv3-Lite model was proposed, and its detection performance as well as the prediction speed were compared with other state of the art models. It was shown that the proposed model works at 240 FPS processing speed while detecting the fish with higher precision and recall values.

Changing the detection scale, increasing the number of anchor boxes, or defining a new loss function are some of the modifications that can be done in the YOLO network structure (Raza and Hong, 2020). Moreover, combining the output of the YOLO model with other information sources such as optical flow and Gaussian mixture models is another strategy to obtain an improved detection in underwater images (Jalal et al., 2020).

In addition to underwater image and video processing methods, there are different applications to identify fish types on the vessel. Such studies involve usage of image classifiers based on convolutional neural networks (CNN) (Zheng et al., 2018) or instance segmentation networks such as Mask R-CNN (French et al., 2020; Tseng et al., 2020). Such segmentation operations are also useful in making morphological measurements on underwater

fish images (Petrellis, 2021). This approach may be practical when the aim is to get an estimate of the individual fish sizes and weights in the catch.

The existing studies focus on either improving the detection performance, the computational load in individual images or application of the deep learning models to a new problem domain. In particular, object detection and tracking are widely studied today in various problem domains such as face recognition (Vijaya Kumar and Mahammad Shafi, 2022), processing of aerial images (ElTantawy and Shehata, 2020; Wu et al., 2022), and maritime surveillance (Jin et al., 2020). Despite the presence of many studies with different purposes and strategies, the number of studies concerning the real-time processing while tracking and counting the detected fish is very limited. In a study that is aimed to serve as a precursor to fish counting tasks, deep learning was used to classify the environmental conditions (Soom et al., 2022). According to the detected conditions, some traditional image processing methods were applied to the image to detect the presence/absence of fish. Even though no object detection and tracking were involved, the processing speed and power consumption of the proposed algorithm was evaluated on different hardware with various specifications.

On the other hand, there exists tracking algorithms developed for underwater objects like fish schools (Liu et al., 2022). In that work, a ResNet50 model was used as the feature extractor and an amendment detection module was proposed to improve the object detection and hence the performance of the tracking. The proposed model was compared with four different tracking algorithms, and it was shown that it outperforms the others in three out of four metrics. In two other studies, an experimental setup was prepared for collecting video footage using a web cam placed above a small fish tank. The fish in the tank were detected by YOLOv3-Tiny model that is trained on the specific dataset. Next, the tracking of the detections was accomplished by optical flow (Mohamed et al., 2020) or Euclidean distance (Wageeh et al., 2021). In these studies, tracking performances are provided poorly with no clear definition of a fish count and a correct track. In another study about fish tracking, an end-to-end model was proposed to detect and track the fish in a tank and determine the abnormal behaviors (Wang et al., 2022a). For the detection task, a modified version of YOLOv5 was used and the tracking was accomplished by SiamRPN++. The proposed model was shown to be operating at 84 FPS with higher detection performance than the other object detectors.

As can be understood from the existing studies, there are many efforts for object detection and tracking in underwater videos. However, the number of applications aimed at counting specific individuals by tracking them is very limited. One example can be the method based on Mask R-CNN to detect and count the catch items during trawling (Sokolova et al., 2021a). In that study, the detections and catch counts were collected under four classes, namely, *Nephrops*, round fish, flat fish, and other. The study involves detailed experiments about different data augmentation methods together with tracking and counting of the catch belonging to the specified classes. Though, it focuses on improvement of the object detection performance, overlooking the detection speed of the algorithm.

Current study differs from previous studies in *i)* counting of *Nephrops* in real-time by detecting and tracking them in underwater videos, *ii)* comparing the effects of three different YOLO models to the performances at every stage of the algorithm as well as the overall processing speed, and *iii)* showing the possibility of real-time monitoring and automated description of the catch items during trawling.

Materials and methods

The video dataset

The dataset used in this study consists of five videos collected using an underwater image acquisition system mounted at the codend entrance of a demersal trawl that allows in-trawl observation during fishing (Sokolova et al., 2021b). The videos were recorded on June 27, 2020, in Skagerrak on commercial *Nephrops* grounds where the catch in each haul were length measured to provide size and count for all caught species The footages have different durations and *Nephrops* ground truth counts. The object densities in the videos are different and such a diversity allows for better performance estimation for real-world applications. The details about the videos are provided in Table 1. The stereo camera of the image acquisition system was set to record videos with a resolution of 1280 × 720 pixels at 60 frames per second (FPS). Only the videos from the right camera were used for processing the frames as the entire data output from the stereo camera is useful for generating depth maps which is not within the scope of this study.

Nephrops detection models

Among various versions of YOLO, the fourth version (YOLOv4) is efficient and stable with various applications in different domains (Bochkovskiy et al., 2020). The object detection task is considered as a regression problem by YOLOv4, and it eliminates the necessity of using large mini-batches during training. It optimizes the trade-off between the detection speed and accuracy, which means that it is possible to obtain accurate detections at high FPS values. Therefore, YOLOv4 has been selected as the primary model for *Nephrops* detection in this study. In addition, two variants of this model, YOLOv4-Tiny and YOLOv4-Scaled, are used to compare their performances.

TABLE 1 Details of the video footages.

	Duration (min)	Total *Nephrops* (no.)	FPS
Video 1	00:55	4	60
Video 2	01:31	6	60
Video 3	07:30	36	60
Video 4	08:10	40	60
Video 5	06:29	23	60

YOLOv4 uses a CSPDarknet53 model as the feature extractor backbone. It contains 29 convolutional layers and has advantages like high receptive field and a large number of parameters that are required for an accurate object detection (Bochkovskiy et al., 2020). The output feature maps of the CSPDarknet53 are passed through a multi-scale max-pooling operation. This operation is implemented by a spatial pyramid pooling (SPP) layer where outputs of four max-pooling operations with kernel sizes 1x1, 5x5, 9x9, and 13x13 are concatenated. Processing with the SPP layer is important for increasing the receptive field and separate the contextual features. YOLOv4 also uses features at different levels of the feature extractor backbone. To accomplish this, feature maps from three layers of the CSPDarknet53 model are input to the path aggregation network (PANet) in which the features are fused both in top-down and bottom-up directions. Such an aggregation allows for simultaneous utilization of localization information present in the lower level features and semantic information in the higher level features. The extracted features with this structure are then passed through a YOLOv3 head to predict bounding box locations and the corresponding confidence scores. To improve generalization and reduce the risk of overfitting, two new methods are introduced in the algorithm: Mosaic and Self-Adversarial Training (SAT). In addition, a continuously differentiable and smooth function Mish is used as the activation between the layers of the network.

YOLOv4-Tiny is a lightweight version of the original YOLOv4 architecture. The major differences are in the numbers of anchor boxes and the convolutional layers in the backbone. Specifically, the tiny model has six anchor boxes while the original version has nine. Also, the number of YOLO prediction layers was reduced from three to two, which allows higher prediction speed while performing poor on the small objects. The scaled version of YOLOv4 (YOLOv4-Scaled) introduces modifications in the backbone and neck structures of the YOLOv4 architecture (Wang et al., 2020). In particular, the first CSP layer in the CSPDarknet53 backbone was replaced by a Darknet residual layer. In addition, up and down feature scaling operations in the PANet and pooling operations in the SPP module are enhanced by CSP blocks that ultimately may decrease the computation cost by 40%.

Tracking and counting of the detected *nephrops*

Since the main goal of the study is to automatically count the number *Nephrops* entering the trawl, the detected *Nephrops* should be tracked as they appear in the frames. To accomplish this, an algorithm to make association between the detections in the consecutive frames should be implemented. This is done by object tracking algorithms that are particularly useful when the object of interest is occluded or not detected for a certain number of frames.

Simple Online and Real-time Tracking (SORT) is the object tracking method used in this study (Bewley et al., 2016). SORT uses 2D motion information for modeling the state (i.e. bounding box location, area, and aspect ratio) of each track in the video. Kalman filter with a linear velocity model predicts the state of the tracks for the next frame (Kalman, 1960). The association between the detections and the predicted tracks is accomplished by applying the Hungarian algorithm (Kuhn, 1955) on the cost matrix whose entries are the IoU values between the detections and predictions. In order to highlight the suitability of the SORT algorithm for real time *Nephrops* tracking, the performance of two other tracking methods, MOSSE and DeepSORT, are tested as well. Details of this comparison are given in Section 4.4.

Due to occlusions or inaccuracy of the object detector model, the target objects may not be detected in all frames when they are in the field of view of the camera. These discontinuities in the detection constitute a challenge for the tracking process. SORT algorithm is capable of predicting the bounding box coordinates in case of such discontinuities. However, if a track is not associated with a detection for 30 consecutive frames, then this track is considered finished. This means that the finished track will not be considered for association with the new detections anymore.

In order to determine the count for the *Nephrops* catches, the tracks output by the SORT tracker are checked. This is done with the help of a horizontal level defined at the top 4/5 of the frame height. When the *Nephrops* are leaving the frame from the bottom, they are partly visible, and this may cause the object tracker to assign different identities to the same *Nephrops* as they are about to disappear. Such an identity switch may generate false positive counts if the horizontal threshold is set to be the bottom of the frame. This is the reason for selecting a level different than the bottom of the frame.

In particular, any track satisfying at least one of the following conditions increases the counter by one:

i. *The track with the lower level of the associated bounding box crosses the horizontal level.* When the *Nephrops* is tracked successfully with no occlusions or distortions, this condition is easily satisfied. This is the most common condition.
ii. *The track with the center of the associated bounding box crosses the horizontal level.* Due to occlusions, tracking of some *Nephrops* are initialized after the lower level of their bounding box is below the horizontal level. This condition is useful for counting such *Nephrops*.
iii. *The track with the height of the associated bounding box is greater than 2/3 of the frame height.* Some *Nephrops* pass very close to the camera causing them to appear very large and in small number of frames. In such cases, the first two conditions cannot be satisfied. So this condition allows for detecting these *Nephrops*.

One sample counting instance for each condition are given in Figure 2.

Model training

The models mentioned in Section 3.2 are trained using an image dataset generated by the frames extracted from the videos included

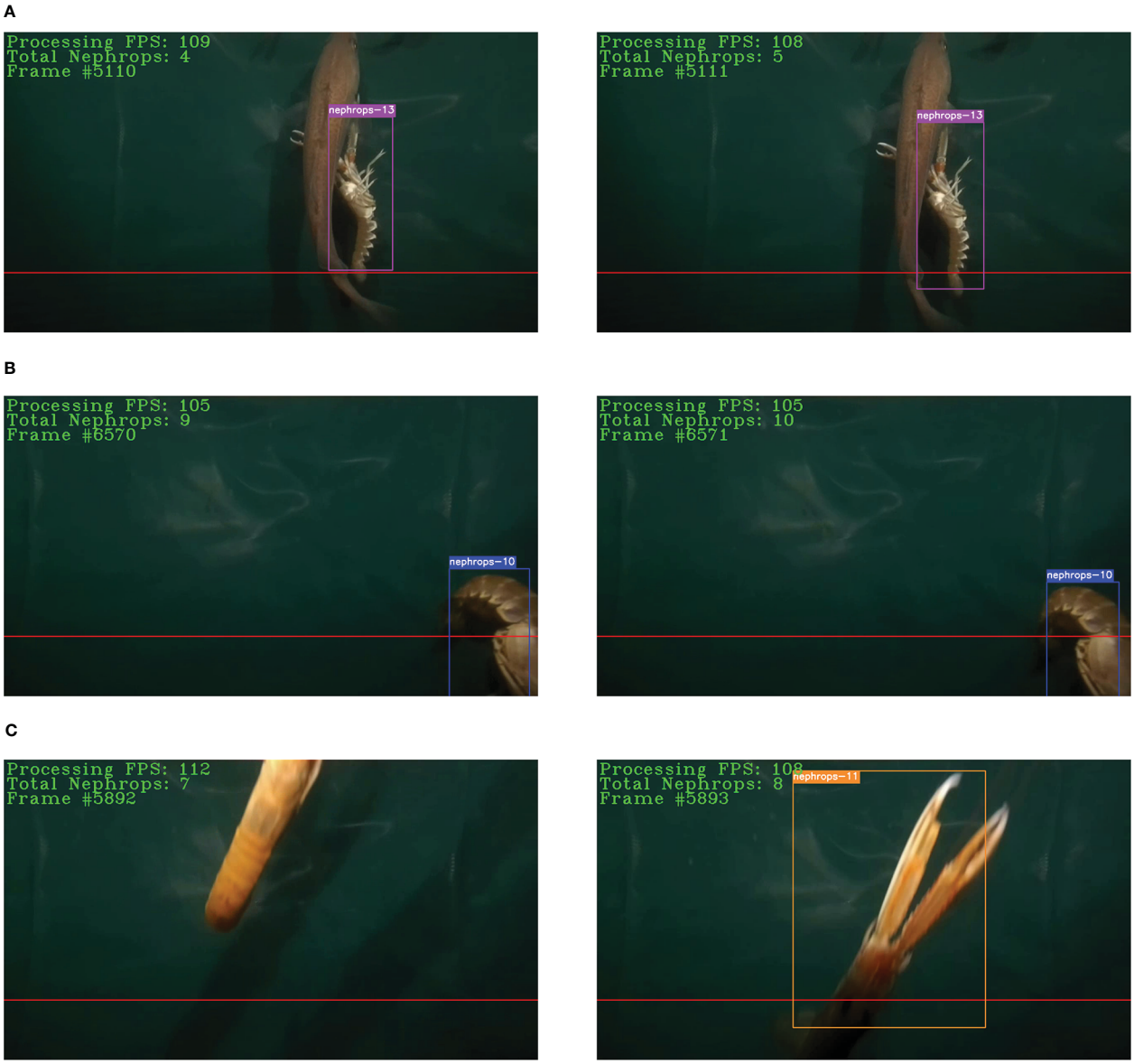


FIGURE 2
Illustration of the counting conditions. Two consecutive frames in the columns. **(A–C)** correspond to the conditions *i, ii, iii*, respectively.

in this study. The majority of the frames in the videos do not contain any objects and are consequently not useful for the training process. Therefore, a manual selection of the frames with some objects is required. A total number of 4044 images were selected according to the presence of *Nephrops*, fish, or others. After the selection of frames, the bounding boxes for the objects in all the frames were manually labeled using the VIA annotation tool (Dutta and Zisserman, 2019). Since the aim is to count the number of *Nephrops* entering the gear, any object other than *Nephrops* was labeled as *other*. Therefore, the object detection step is considered as a binary detection problem.

The dataset was randomly divided into training and test sets with proportions of 87.5% and 12.5%, respectively. Next, 1000 images were generated using the Copy-Paste (CP) augmentation method and added to the training set (Ghiasi et al., 2021). When performing the CP augmentation, pixel values corresponding to the masks of the objects in the source images were pasted onto the destination images. To improve the diversity in the augmented images, some geometric transformations were applied to the images as explained in (Sokolova et al., 2021a). The details, like number of images and the object instances in the image dataset after the augmentation are given in Table 2, and three sample images are provided in Figure 3.

TABLE 2 Numbers of images and instances from both classes in the training and test sets used in the object detection step.

	Images	*Nephrops* Instances	*Other* Instances
Training Set	4538	3766	8014
Test Set	506	204	775

The darknet framework was used for the training of the models (Redmon, 2016). The training and testing were performed on a Tesla A100 GPU with 40 GB RAM, CUDA 11.1, and cudnn v8.0.4.30. All the coding was done with Python v3.9.12 following the instructions and model configuration files made available at (Bochkovskiy, 2022). Some of the hyperparameters regarding the models and their training are listed in Table 3. Note that all the models were trained for 6000 iterations and the weights yielding the best detection performance were used in the subsequent steps.

Performance evaluation metrics

The performances of each step in the study are evaluated and reported separately in Section 4. To evaluate the object detection performance, different *mAP* values are calculated for each of the models using the test set. *mAP* is a quantification of the detection performance by comparing the amount of overlap between the ground truth and predicted bounding boxes. It is a widely used metric and has good representation of the detection performance as it considers both the prediction confidence score and the

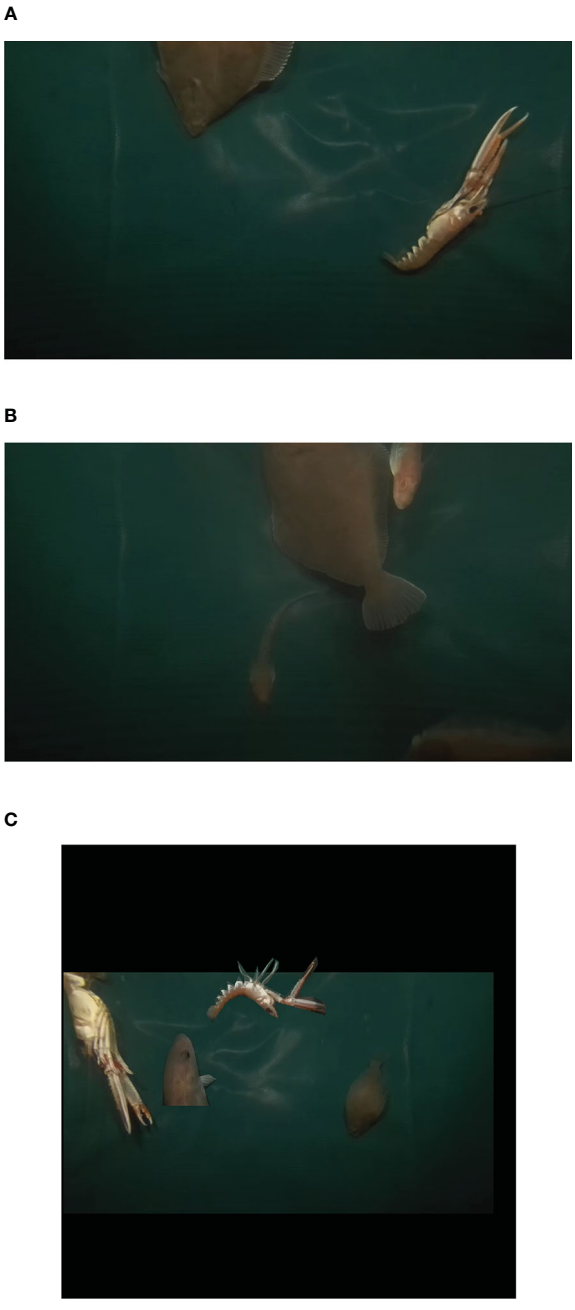


FIGURE 3
Samples from the image dataset. **(A)** An image with a *Nephrops* instance. **(B)** An image with some *other* instances. **(C)** An image with copy-paste augmentation.

TABLE 3 Summary of the model settings.

	Network Size	Initial Learning Rate	Momentum	Decay	Training Epochs
YOLOv4	416	0.00100	0.949	0.0005	6000
YOLOv4-Tiny	416	0.00261	0.900	0.0005	6000
YOLOv4-Scaled	640	0.00100	0.949	0.0005	6000

intersection over union (*IoU*) values. First, the confidence scores for the bounding boxes are converted into class labels for different threshold values. This allows to obtain a confusion matrix for each threshold and hence calculate the precision and recall values using the True Positive (*TP*), False Positive (*FP*), and False Negative (*FN*) in each matrix given by the following equations.

$$Precision_n = \frac{TP_n}{TP_n + FN_n}$$

$$Recall_n = \frac{TP_n}{TP_n + FN_n}$$

Here the subscript n represents different confidence score thresholds. The multiple (recall, precision) points correspond to a curve in 2D space (precision-recall curve), and the average precision (*AP*) value is the weighted mean of the precisions with the weights being the changes in the recall values.

$$AP = \sum_{i=0}^{n-1} (Recall_i - Recall_{i-1}) Precision_i$$

This *AP* calculation procedure is repeated for all classes separately in the dataset. The average of all the *AP* values is defined as the *mAP* which can be obtained by

$$mAP = \frac{1}{c} \sum_{i=1}^{c} AP_i$$

where c represents the number of classes in the dataset and AP_i is the *AP* value for the ith class.

The *mAP* value can be computed for different IoU thresholds that affects the shape of the precision-recall curves. As a convention, the *mAP* value is calculated for $IoU = 0.50$ (*mAP*@.50). However, for benchmarking purposes, *mAP* values at different IoU thresholds are calculated and averaged as well. In this study, three *mAP* values are provided as the detection performance of the models: *mAP*@.50, *mAP*@.75, and *mAP*@.50:.05:.95 (*mAP* values averaged for the thresholds from 0.50 to 0.95 with steps of 0.05). In addition, since the purpose is to track and count the *Nephrops* only, the *AP* values belonging to *Nephrops* class (AP_{nep}) are also given for the same IoU thresholds.

Having obtained the tracks as the algorithm output as explained in Section 3.3, the tracking performance metrics were calculated. Among the calculated metrics, multi-object tracking accuracy (MOTA) is a combination of three error types namely, number of misses, false positives, and mismatches. It is obtained by normalizing the total of these three errors by the number of ground truth tracks. In calculation of MOTA, only the track locations are used. In other words, no bounding box information is considered in MOTA. To overcome this situation, another metric called multi-object tracking precision (MOTP) is defined. MOTP is the average overlap between the bounding boxes of predictions and ground truths. Mostly tracked (MT) and mostly lost (ML) are two quality measures that consider the ratio of successfully tracked frames for an object. A track is MT if it is tracked for at least 80% of its life span. If the tracking ratio is less than 20%, then is called ML. Within the context of object tracking, it is also desirable to obtain tracks preserving their identities with small numbers of untracked frames. Therefore, it is possible to mention two more metrics here. Identity switch (ID-Sw) is the total number of tracks changing their identity for the same ground truth object. Fragmentation is the number of interruptions in the track where no tracking is made. Finally, higher order tracking accuracy (HOTA) combines errors originating from both association and detection (Luiten et al., 2021). Specifically, it is the geometric mean of association accuracy and detection accuracy.

Results

Detection performance of the models

The *mAP* and AP_{nep} values for different IoU thresholds for all three models are given in Table 4. These values are obtained by passing the test set samples in the image dataset introduced in Section 3.1 through the trained models. Note that the best weights determined during the training phase are used for prediction on the test set which can be considered as a regularization step to avoid overfitting. In other words, the weights calculated in the subsequent iterations are not considered for *Nephrops* detection. The best weights are obtained at iterations 4962, 5245, and 4113 for YOLOv4, YOLOv4-Tiny, and YOLOv4-Scaled, respectively.

In most of the performance metrics, YOLOv4-Scaled outperforms the other two models. Nevertheless, the differences between YOLOv4 and YOLOv4-Scaled are minor which precludes suggesting the best model for all cases. For the threshold $IoU = 0.5$, the scaled version is slightly better at detection of the *Nephrops*, but when the *AP* values for both classes are considered, YOLOv4 has a higher *mAP* value. This means that YOLOv4-Scaled is not as precise as YOLOv4 when detecting the objects from the other class. On the other hand, the difference between the performances of YOLOv4-Tiny and the other two models is smaller when $IoU = 0.5$. This indicates that the tiny version is capable of detecting the bounding boxes but not with as high IoU values as those obtained by the other models.

TABLE 4 Performance comparison of the detector models.

	mAP (%)			AP_{nep} (%)		
	@.50	@.75	@.50:.05:.95	@.50	@.75	@.50:.05:.95
YOLOv4	**97.82**	85.58	71.89	97.84	91.37	74.76
YOLOv4-Tiny	95.10	73.06	62.71	94.57	76.95	64.28
YOLOv4-Scaled	97.55	**88.10**	**72.28**	**98.47**	**94.05**	**75.97**

Best values are provided in bold.

Tracking and counting performance of the models

Note that only the tracks satisfying the count conditions were involved in the tracking performance calculation because these are the tracks used in counting performance calculation as well. In addition, the tracking metrics were obtained for all five videos separately, but their average values are provided here as one single clustered column chart (Figure 4). The MOTA, MOTP, and HOTA values are given as percentages (Figure 4A) and the rest are number of tracks (Figure 4B).

The *Nephrops* counts output by the algorithm associated with the tracks are given in Table 5. The numbers of true positive counts are reported together with the numbers of false positive and false negative counts together with the correct count rates for each individual video. The lowest total number of false positives is achieved by YOLOv4-Scaled which has the highest false negative tracks as well. Therefore, it is possible to explain the low false

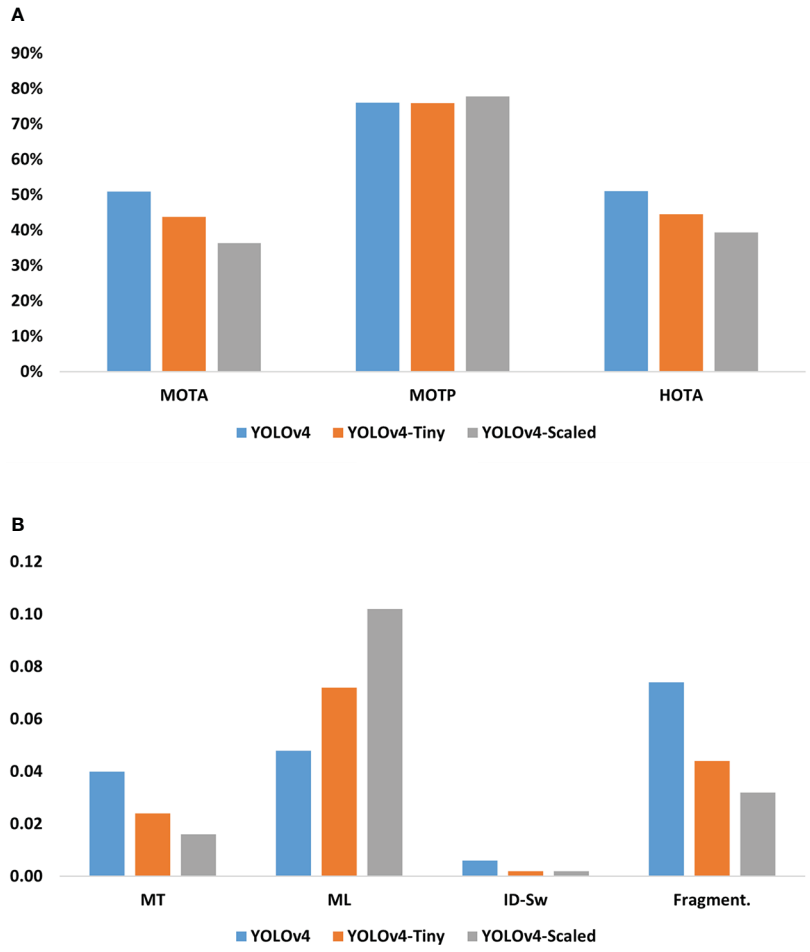


FIGURE 4
Tracking performances associated with the detectors. **(A)** Percentage values for MOTA, MOTP, and HOTA, **(B)** MT, ML, ID-Sw, and Fragmentation numbers averaged over the test videos.

TABLE 5 Detailed numbers of counts obtained by the detection models.

		Video-1	Video-2	Video-3	Video-4	Video-5	Total
	Ground Truth	4	6	36	40	23	109
YOLOv4	**Output**	4	4	39	31	19	97
	True Positives	4	4	34	27	19	88
	False Positives	0	0	5	4	0	9
	False Negatives	0	2	2	13	4	21
	Correct Count Rate (%)	100.00	66.67	94.44	67.50	82.61	80.73
YOLOv4-Tiny	**Output**	4	4	33	24	18	83
	True Positives	3	4	31	21	18	77
	False Positives	1	0	2	3	0	6
	False Negatives	1	2	5	19	5	32
	Correct Count Rate (%)	75.00	66.67	86.11	52.50	78.26	70.64
YOLOv4-Scaled	**Output**	3	4	27	19	18	71
	True Positives	3	4	25	17	18	67
	False Positives	0	0	2	2	0	4
	False Negatives	1	2	11	23	5	42
	Correct Count Rate (%)	75.00	66.67	69.44	42.50	78.26	61.46

positive rate by its inefficiency in generating tracks satisfying the count conditions. The lowest amount of false tracks are achieved by YOLOv4 which also has the highest true positives. Specifically, the related F-scores calculated on the total counts for YOLOv4, Tiny, and Scaled versions are 85.44%, 80.21%, and 74.44%, respectively.

Processing speed comparison of the models

The required amount of calculations in the model and the hardware specifications are the two major factors affecting the processing speed. The calculation amounts are determined at the design stage of the models, and this can be adjusted to some degree by changing the input image sizes which is also named as network size (see Table 3). Typically, a larger network size in the model yields better object detection, sacrificing the processing speed and vice versa. The input image size for the YOLOv4-Scaled model was adjusted to be higher than the other two models to improve its detection accuracy. Such an adjustment allowed for obtaining a similar accuracy with YOLOv4 model and hence benchmarking their tracking, counting and speed performances.

The FPS values for each model and video are summarized in Table 6. As expected, the YOLOv4-Tiny model is the fastest in all the videos because it has a reduced number of computational layers to enhance its speed. The slowest model is YOLOv4-Scaled. The reason for its lower FPS values is related with its larger network size. However, a smaller network size for this model would cause lower detection and tracking performances eventually yielding a lower number of true positive counts.

Benchmarking with other trackers

To evaluate the suitability of SORT, two other object tracking algorithms were tested on the same dataset. One of these methods is based on a correlation filter, namely, Minimum Output Sum of Squared Error (MOSSE) filter (Bolme et al., 2010). The reason for selecting this object tracker is that its processing speed is claimed to reach 669 FPS (Bolme et al., 2010). In addition, usage of MOSSE was

TABLE 6 Comparison of image processing speed between models in frames per second (mean [min-max]).

	Video-1	Video-2	Video-3	Video-4	Video-5	Average
YOLOv4	116.49 [65-123]	115.64 [76-123]	116.67 [75-123]	114.77 [69-123]	115.76 [62-122]	115.87 [69.4-122.8]
YOLOv4-Tiny	267.51 [84-323]	248.58 [96-267]	251.22 [76-318]	251.50 [75-316]	248.72 [91-311]	253.51 [84.4-307.0]
YOLOv4-Scaled	78.93 [39-80]	79.51 [51-81]	80.31 [40-82]	79.93 [44-82]	80.73 [48-82]	79.88 [44.4-81.4]

shown to be one of the effective trackers tested in underwater videos (Lopez-Marcano et al., 2021). The MOSSE algorithm initializes a correlation filter based on a detected object in a frame. Next, in the subsequent frames, the algorithm looks for a location having the highest correlation with the initially detected object. Due to the changes in appearance of the same *Nephrops* instances throughout the video, the *Nephrops* detection used for generating the correlation filter is updated every fifth frame. This approach was implemented earlier for tracking of yellowfin bream in underwater videos (Lopez-Marcano et al., 2021).

The other tracker evaluated is DeepSORT, an improved version of the SORT algorithm (Wojke et al., 2017). DeepSORT uses the appearance information of the detected objects together with their motion information in 2D. The motion information is quantified by the Mahalanobis distance between the detected bounding box centroids and the Kalman filter predictions under a constant velocity model. On the other hand, the appearance features for each detection are obtained by passing the bounding box region through a pre-trained CNN containing two convolutional and six residual layers. The minimum cosine distance between the appearance features of the detections and the last 100 features of each track is determined as the second metric used by DeepSORT. For the benchmarking experiments, the resources and the instructions made available in the official repository of DeepSORT are utilized (Wojke, 2019).

Instead of reporting the full detailed results for benchmarking trackers, only MOTA, HOTA, correct count rate, average FPS values, and F-scores for YOLOv4 model are provided (Figure 5). Evaluation of these metrics is sufficient for comparing the trackers by understanding their overall performance.

Discussion

A major challenge in demersal trawling is the lack of information about the catch entering the gear during fishing. This study demonstrates a full pipeline to acquire, process and display catch information for *Nephrops*, in close to real-time, to act as a decision tool for the fisher during the fishing operation. The applicability of such tools in commercial trawling and their potential improvements is discussed below.

One advantage of the proposed algorithm is the powerful image acquisition system that provides mostly sediment-free clear videos for being processed in the subsequent steps (Sokolova et al., 2021b; Sokolova et al., 2022). In the existing literature for underwater image processing, there are some papers where the effects of preprocessing on underwater images are analyzed for improving the detection performance (Han et al., 2020; Zhou et al., 2022). But the preprocessing requires some time, degrading the overall processing speed. In addition, there are different types of degradations such as low contrast and color distortion present in the underwater images (An et al., 2021). Our method does not require any preprocessing to enhance the detection accuracy because the image acquisition system is robust and capable of capturing clear videos with adjustable illumination (Sokolova et al., 2021b).

Evaluation of the algorithm steps

Since the followed strategy is tracking-by-detection, successful *Nephrops* detection is expected to imply more accurate tracking which eventually may result in better *Nephrops* counts. Hence, achieving high *mAP* is critical at the object detection step. The performances of object detector models may be considered as sufficiently successful for an accurate tracking and counting task because all three models have *mAP* @.50 values above 95% (Table 4). In addition, the *Nephrops* detection performance, AP_{nep} value, associated with YOLOv4-Scaled model is the highest indicating a better detection capability of *Nephrops*. However, this situation is in connection with the increased size of the YOLOv4-Scaled model which slows down its respective detection speed (Table 6).

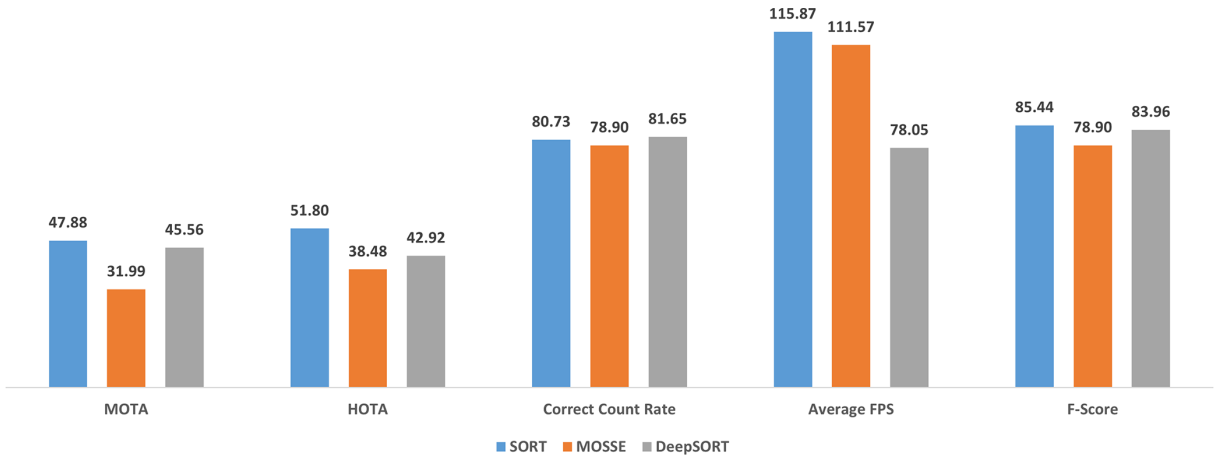


FIGURE 5
Some performance metrics obtained by three different object tracking algorithms.

In the literature, there are numerous metrics defined for evaluating the performance of an object tracking algorithm. For simplicity, only those metrics commonly mentioned in the object tracking literature are provided in this paper. Among the three models, YOLOv4 model has the best values for MOTA, MT, ML, and HOTA. For a detection model, having higher MT and lower ML track count means that their associated successive detections are good enough to attain a valid track. This idea is also supported by the high accuracy values in MOTA and HOTA. On the other hand, an identity switch can be the source of a false positive count provided that the switching happens somewhere close to the horizontal level defined for counting conditions. As for the MOTP, it is very close for three of the models. This means that they have nearly the same level of success in bounding box localization throughout the tracks and cannot be used as a distinguishing factor for commenting on the counting performance.

Finally, it is possible to mention the performance for total *Nephrops* counts and the processing speeds of the method. Checking only the total counts at the end of the video may be misleading since some *Nephrops* are not counted while there may be multiple counts for some others. Therefore, checking the false positive and false negative counts together with the true positives gives better insight about the counting performance. The quantification of these three types of tracks is done by calculating the F-scores for each detector model. In addition, the rates for correct counts in each video are provided. At this point, it is notable that the correct count rates for Video-4 are relatively low when compared to the other four videos. The reason for such a remarkable difference is that Video-4 has some sediments degrading the visibility of the objects in the video. This situation highlights the importance of sediment-free video acquisition. Furthermore, when Tables 4, 5 are considered together, it is possible to conclude that high performance at the object detection step does not always imply better correct count rates. This is apparent for the YOLOv4-Scaled model which has a very high detection rate but fails to achieve good count performance.

As for the processing speed, it is measured in terms of FPS. It is the type of the detector model that has a major impact on the overall duration of processing a frame. In addition, updating the object tracks by the SORT algorithm takes some time. During the experiments on the videos, it was observed that, on average, 1.6% of the total processing duration of the frames are used by SORT tracking algorithm when YOLOv4 is used as the object detector. However, tracking is effective only when there is a tracked object in the frame. Nevertheless, the maximum processing speed related with three of the models is higher than the FPS value of the input video (Table 6). This means that the detectors are capable of running at real-time processing speed, but this speed may be reduced when there is a tracked object in the video. On average, the processing speeds of YOLOv4-Scaled is slightly below the real time threshold while the other two models are fast enough to be considered real-time.

The benchmarking results of SORT with MOSSE and DeepSORT trackers revealed that SORT is a better tracker for this application in terms of tracking accuracy, *Nephrops* counting, and processing speed. The major problem with the MOSSE tracker is the requirement for updating the correlation filters frequently. This process slows down the procedure considerably. On the other hand, tracking without any correlation filter update step, MOSSE is quite inefficient for this problem because the *Nephrops* individuals float and rotate under the influence of water flow causing their appearance to be changed as they are in the field of view of the camera. As for DeepSORT, it is more accurate than MOSSE in terms of counting performance. However, the CNN-based feature extraction step slows down the overall tracking speed and eventually causes the slowest processing.

Implications for the *nephrops* fishing

Demersal trawling is a blind process today, which means that fishers do not know if they are catching the target species during trawling operation. This study constitutes a basis for addressing this problem by outputting the target catch count with a real-time speed. In other words, it demonstrates the possibility of providing the *Nephrops* catch amount throughout the trawling operation. Such information is useful for not only improving the catch rates of the target species but also reducing the bycatch amounts, oil and energy consumption, and ultimately improve the economic, environmental, and social sustainability of the fishery.

Further development

The first step for further improvement of the proposed method is to run it on an edge device with limited computational power. Note that the reported results in this study were obtained using a powerful processing unit (Section 3.4). In real world applications, it may not be practical to access such a computer. Therefore, experimentation with an edge device, which is more accessible onboard commercial fishing vessels, is one of the improvement plans with high priority. The change of the processing platform may not affect the correct count rates, but will have an influence on the overall processing speed. Nevertheless, the achieved speed with YOLOv4-Tiny model is promising and it may still perform sufficiently fast on an edge device.

When there is a tracked object in the video, the tracking speed drops considerably. In other words, tracking step is a bottleneck in the procedure. However, SORT is known to be one of the fast tracking algorithms in the literature, which is also supported by the benchmarking results. In case of requiring higher speed, skipping some intermediate frames may be helpful at cost of degradation in the count accuracy. This may contribute to the compensation of the speed loss due to the edge device. Besides, even if there is a small delay, the achieved processing speed may be considered as a significant improvement when compared to hours of delay associated with the current situation, where information on catch rates and compositions is only available once the catch is taken onboard the vessel.

In the longer term, the method may be extended to detect and count more species and contribute to a larger scale in fisheries. However, this requires generation of a larger video dataset

containing more diverse species. In addition, the edge processing unit may be connected to the stereo camera directly by integrating them inside the underwater camera box. This may be coupled with a wireless transceiver device that transmits the count information, e.g. acoustically to a screen onboard. This key information is sufficient for the fisher to decide whether to continue fishing in the same area.

Conclusion

This study demonstrates the possibility of using state-of-the-art deep learning methods to develop real-time decision tools for the trawl fisheries demonstrated here as a *Nephrops* counter. In particular, the experiments are carried out with three different object detector models on underwater videos collected by an in-trawl camera. The detection, tracking, and counting performances as well as the processing speeds associated with these models are calculated. According to the obtained results, it is possible to conclude that such a system is promising for improving the sustainability of trawl fisheries.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://doi.org/10.11583/DTU.21769442.

Ethics statement

Ethical review and approval was not required for the animal study because all data collection was conducted during trawl fishing at sea which do not require an ethical permit or animal welfare approval.

Author contributions

EA: methodology, coding, manuscript writing. JF: conceptualization, supervision, manuscript writing and editing. LK: funding acquisition, conceptualization, supervision, manuscript writing and editing. All authors contributed to the article and approved the submitted version.

Funding

This work has received funding from the European Maritime and Fisheries Fund (EMFF), the Ministry of Food, Agriculture and Fisheries of Denmark, and the European Union's Horizon 2020 research and innovation program as part of the projects: Development of a real-time catch monitoring system with automatic detection of the catch composition to minimize catch of unwanted species and sizes [AutoCatch (33112-P-18-051)], Udvikling af SELEKTive redskaber og teknologier til kommercielle fiskerier [SELEKT (33113-I-22-187)], and Smart fisheries technologies for an efficient, compliant and environmentally friendly fishing sector [SMARTFISH (agreement no: 7553521)].

Acknowledgments

The authors thank the skipper and the crew on DTU's research vessel RV Havfisken for assistance in data collection at sea.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Aguzzi, J., and Sardà, F. (2008). A history of recent advancements on nephrops norvegicus behavioral and physiological rhythms. *Rev. Fish Biol. Fish* 18, 235–248. doi: 10.1007/S11160-007-9071-9/FIGURES/6

Allken, V., Rosen, S., Handegard, N. O., and Malde, K. (2021). A deep learning-based method to identify and count pelagic and mesopelagic fishes from trawl camera images. *ICES J. Mar. Sci.* 78, 3780–3792. doi: 10.1093/ICESJMS/FSAB227

An, D., Hao, J., Wei, Y., Wang, Y., and Yu, X. (2021). Application of computer vision in fish intelligent feeding system–a review. *Aquac Res.* 52, 423–437. doi: 10.1111/ARE.14907

Bergmann, M., Wieczorek, S. K., Moore, P. G., and Atkinson, R. J. A. (2002). Discard composition of the nephrops fishery in the Clyde Sea area, Scotland. *Fish Res.* 57, 169–183. doi: 10.1016/S0165-7836(01)00345-9

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B. (2016). "Simple online and realtime tracking," in *Proceedings - International Conference on Image Processing, ICIP*, , 2016-August. 3464–3468. doi: 10.1109/ICIP.2016.7533003

Bochkovskiy, A. (2022) *GitHub - AlexeyAB/darknet: YOLOv4 / scaled-YOLOv4 / YOLO - neural networks for object detection (Windows and Linux version of darknet)*. Available at: https://github.com/AlexeyAB/darknet (Accessed November 14, 2022).

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. doi: 10.48550/arxiv.2004.10934

Bolme, D. S., Beveridge, J. R., Draper, B. A., and Lui, Y. M. (2010). "Visual object tracking using adaptive correlation filters," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2544–2550. doi: 10.1109/CVPR.2010.5539960

Dutta, A., and Zisserman, A. (2019). "The VIA annotation software for images, audio and video," in *Proceedings of the 27th ACM International Conference on Multimedia MM '19*, New York, NY, USA. 2276–2279 (Association for Computing Machinery). doi: 10.1145/3343031.3350535

Eigaard, O. R., Bastardie, F., Hintzen, N. T., Buhl-Mortensen, L., Buhl-Mortensen, P., Catarino, R., et al. (2017). The footprint of bottom trawling in European waters: distribution, intensity, and seabed integrity. *ICES J. Mar. Sci.* 74, 847–865. doi: 10.1093/ICESJMS/FSW194

ElTantawy, A., and Shehata, M. S. (2020). Local null space pursuit for real-time moving object detection in aerial surveillance. *Signal Image Video Process* 14, 87–95. doi: 10.1007/S11760-019-01528-Y/FIGURES/3

Feekings, J., Christensen, A., Jonsson, P., Frandsen, R., Ulmestrand, M., Munch-Petersen, S., et al. (2015). The use of at-sea-sampling data to dissociate environmental variability in Norway lobster (Nephrops norvegicus) catches to improve resource exploitation efficiency within the Skagerrak/Kattegat trawl fishery. *Fish Oceanogr* 24, 383–392. doi: 10.1111/FOG.12116

French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., et al. (2020). Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES J. Mar. Sci.* 77, 1340–1353. doi: 10.1093/ICESJMS/FSZ149

Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.-Y., Cubuk, E. D., et al. (2021). "Simple copy-paste is a strong data augmentation method for instance segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2918–2928. doi: 10.48550/arXiv.2012.07177

Han, F., Yao, J., Zhu, H., and Wang, C. (2020). Underwater image processing and object detection based on deep CNN method. *J. Sens* 2020, 1–20. doi: 10.1155/2020/6707328

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., et al. (2019). Searching for MobileNetV3. doi: 10.48550/arxiv.1905.02244

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. doi: 10.48550/arxiv.1704.04861

Hu, J., Zhao, D., Zhang, Y., Zhou, C., and Chen, W. (2021). Real-time nondestructive fish behavior detecting in mixed polyculture system using deep-learning and low-cost devices. *Expert Syst. Appl.* 178, 115051. doi: 10.1016/J.ESWA.2021.115051

Jalal, A., Salman, A., Mian, A., Shortis, M., and Shafait, F. (2020). Fish detection and species classification in underwater environments using deep learning with temporal information. *Ecol. Inform* 57, 101088. doi: 10.1016/J.ECOINF.2020.101088

Jin, J., Zhang, J., Liu, D., Shi, J., Wang, D., and Li, F. (2020). Vision-based target tracking for unmanned surface vehicle considering its motion features. *IEEE Access* 8, 132655–132664. doi: 10.1109/ACCESS.2020.3010327

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *J. Basic Eng.* 82, 35–45. doi: 10.1115/1.3662552

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Res. Logistics Q.* 2, 83–97. doi: 10.1002/NAV.3800020109

Liu, T., He, S., Liu, H., Gu, Y., and Li, P. (2022). A robust underwater multiclass fish-school tracking algorithm. *Remote Sens.* 14, 4106. doi: 10.3390/RS14164106

Liu, X., Jia, Z., Hou, X., Fu, M., Ma, L., and Sun, Q. (2019). "Real-time marine animal images classification by embedded system based on mobilenet and transfer learning," in *OCEANS 2019 - Marseille*. 1–5. doi: 10.1109/OCEANSE.2019.8867190

Liu, H., Song, P., and Ding, R. (2020). "Towards domain generalization in underwater object detection," in *2020 IEEE International Conference on Image Processing (ICIP)*. 1971–1975. doi: 10.1109/ICIP40778.2020.9191364

Lopez-Marcano, S., Jinks, L., Buelow, C. A., Brown, C. J., Wang, D., Kusy, B., et al. (2021). Automatic detection of fish and tracking of movement for ecology. *Ecol. Evol.* 11, 8254–8263. doi: 10.1002/ECE3.7656

Luiten, J., Ošep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., et al. (2021). HOTA: A higher order metric for evaluating multi-object tracking. *Int. J. Comput. Vis.* 129, 548–578. doi: 10.1007/S11263-020-01375-2/FIGURES/18

Main, J., and Sangster, G. I. (1985). "The behaviour of the Norway lobster, nephrops norvegicus (L.), during trawling," in *Scottish Fisheries research report*, vol. 43. (Aberdeen: Department of Agriculture and Fisheries for Scotland), 1–23.

Mohamed, H. E. D., Fadl, A., Anas, O., Wageeh, Y., Elmasry, N., Nabil, A., et al. (2020). MSR-YOLO: Method to enhance fish detection and tracking in fish farms. *Proc. Comput. Sci.* 170, 539–546. doi: 10.1016/J.PROCS.2020.03.123

Muksit, A., Hasan, F., Hasan Bhuiyan Emon, M. F., Haque, M. R., Anwary, A. R., and Shatabda, S. (2022). YOLO-fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecol. Inform* 72, 101847. doi: 10.1016/J.ECOINF.2022.101847

Naseer, A., Baro, E. N., Khan, S. D., and Gordillo, Y. V. (2020). "Automatic detection of nephrops norvegicus burrows in underwater images using deep learning," in *2020 Global Conference on Wireless and Optical Technologies (GCWOT)*. 1–6. doi: 10.1109/GCWOT49901.2020.9391590

Petrellis, N. (2021). Measurement of fish morphological features through image processing and deep learning techniques. *Appl. Sci.* 11, 4416. doi: 10.3390/APP11104416

Prados, R., Garcia, R., Gracias, N., Neumann, L., and Vagstol, H. (2017). "Real-time fish detection in trawl nets," in *OCEANS 2017*, Aberdeen, 2017-October. 1–5. doi: 10.1109/OCEANSE.2017.8084760

Raza, K., and Hong, S. (2020). Fast and accurate fish detection design with improved YOLO-v3 model and transfer learning. *Int. J. Advanced Comput. Sci. Appl.* 11, 7–16. doi: 10.14569/IJACSA.2020.0110202

Redmon, J. (2016) *Darknet: Open source neural networks in c*. Available at: https://pjreddie.com/darknet/ (Accessed November 14, 2022).

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2015). You only look once: Unified, real-time object detection. doi: 10.48550/arxiv.1506.02640

Sala, E., Mayorga, J., Bradley, D., Cabral, R. B., Atwood, T. B., Auber, A., et al. (2021). Protecting the global ocean for biodiversity, food and climate. *Nature* 592, 397–402. doi: 10.1038/s41586-021-03371-z

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. doi: 10.48550/arxiv.1801.04381

Sokolova, M., Mompó Alepuz, A., Thompson, F., Mariani, P., Galeazzi, R., and Krag, L. A. (2021a). A deep learning approach to assist sustainability of demersal trawling operations. *Sustainability* 13, 12362. doi: 10.3390/SU132212362

Sokolova, M., O'Neill, F. G., Savina, E., and Krag, L. A. (2022). Test and development of a sediment suppressing system for catch monitoring in demersal trawls. *Fish Res.* 251, 106323. doi: 10.1016/J.FISHRES.2022.106323

Sokolova, M., Thompson, F., Mariani, P., and Krag, L. A. (2021b). Towards sustainable demersal fisheries: NepCon image acquisition system for automatic nephrops norvegicus detection. *PloS One* 16, e0252824. doi: 10.1371/JOURNAL.PONE.0252824

Soom, J., Pattanaik, V., Leier, M., and Tuhtan, J. A. (2022). Environmentally adaptive fish or no-fish classification for river video fish counters using high-performance desktop and embedded hardware. *Ecol. Inform* 72, 101817. doi: 10.1016/J.ECOINF.2022.101817

Tseng, C.-H., Kuo, Y.-F., Tseng, C.-H., and Kuo, Y.-F. (2020). Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. *ICES J. Mar. Sci.* 77, 1367–1378. doi: 10.1093/ICESJMS/FSAA076

Tully, O., and Hillis, J. P. (1995). Causes and spatial scales of variability in population structure of nephrops norvegicus (L.) in the Irish Sea. *Fish Res.* 21, 329–347. doi: 10.1016/0165-7836(94)00303-E

Underwood, M. J., Rosen, S., Engas, A., and Eriksen, E. (2014). Deep vision: An in-trawl stereo camera makes a step forward in monitoring the pelagic community. *PloS One* 9, e112304. doi: 10.1371/JOURNAL.PONE.0112304

Underwood, M. J., Rosen, S., Engås, A., Jorgensen, T., and Fernö, A. (2018). Species-specific residence times in the aft part of a pelagic survey trawl: implications for inference of pre-capture spatial distribution using the deep vision system. *ICES J. Mar. Sci.* 75, 1393–1404. doi: 10.1093/ICESJMS/FSX233

Vijaya Kumar, D. T. T., and Mahammad Shafi, R. (2022). A fast feature selection technique for real-time face detection using hybrid optimized region based convolutional neural network. *Multimed Tools Appl.*, 1–14. doi: 10.1007/S11042-022-13728-9

Wageeh, Y., Mohamed, H. E. D., Fadl, A., Anas, O., ElMasry, N., Nabil, A., et al. (2021). YOLO fish detection with euclidean tracking in fish farms. *J. Ambient Intell. Humaniz Comput.* 12, 5–12. doi: 10.1007/S12652-020-02847-6/FIGURES/6

Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2020). "Scaled-YOLOv4: Scaling cross stage partial network," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 13024–13033. doi: 10.48550/arxiv.2011.08036

Wang, J., He, X., Shao, F., Lu, G., Jiang, Q., Hu, R., et al. (2022b). A novel attention-based lightweight network for multiscale object detection in underwater images. *J. Sens* 2022, 2582687. doi: 10.1155/2022/2582687

Wang, H., Zhang, S., Zhao, S., Wang, Q., Li, D., and Zhao, R. (2022a). Real-time detection and tracking of fish abnormal behavior based on improved YOLOV5 and SiamRPN++. *Comput. Electron Agric.* 192, 106512. doi: 10.1016/J.COMPAG.2021.106512

Wojke, N. (2019) *GitHub - nwojke/deep_sort: Simple online realtime tracking with a deep association metric*. Available at: https://github.com/nwojke/deep_sort (Accessed November 14, 2022).

Wojke, N., Bewley, A., and Paulus, D. (2017). "Simple online and realtime tracking with a deep association metric," in *Proceedings - International Conference on Image Processing, ICIP*, , 2017-September. 3645–3649. doi: 10.48550/arxiv.1703.07402

Wu, X., Li, W., Hong, D., Tao, R., and Du, Q. (2022). Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey. *IEEE Geosci Remote Sens Mag* 10, 91–124. doi: 10.1109/MGRS.2021.3115137

Yao, Y., Qiu, Z., and Zhong, M. (2019). "Application of improved MobileNet-SSD on underwater sea cucumber detection robot," in *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. 402–407. doi: 10.1109/IAEAC47372.2019.8997970

Zhang, M., Xu, S., Song, W., He, Q., and Wei, Q. (2021). Lightweight underwater object detection based on YOLO v4 and multi-scale attentional feature fusion. *Remote Sens (Basel)* 13, 1–22. doi: 10.3390/rs13224706

Zhao, S., Zheng, J., Sun, S., and Zhang, L. (2022). An improved YOLO algorithm for fast and accurate underwater object detection. *Symmetry (Basel)* 14, 1–16. doi: 10.3390/sym14081669

Zheng, Z., Guo, C., Zheng, X., Yu, Z., Wang, W., Zheng, H., et al. (2018). "Fish recognition from a vessel camera using deep convolutional neural network and data augmentation," in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans, OCEANS - Kobe 2018*. doi: 10.1109/OCEANSKOBE.2018.8559314

Zhou, J., Yang, Q., Meng, H., and Gao, D. (2022). An underwater target recognition method based on improved YOLOv4 in complex marine environment. *Syst. Sci. Control. Eng.* 10, 590–602. doi: 10.1080/21642583.2022.2082579

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 10 March 2023
DOI 10.3389/fmars.2023.1099479

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Christophe Guinet,
Centre National de la Recherche
Scientifique (CNRS), France
Duane Edgington,
Monterey Bay Aquarium Research Institute
(MBARI), United States

*CORRESPONDENCE
Antoine Gagné-Turcotte
antoine@whaleseeker.com

SPECIALTY SECTION
This article was submitted to
Ocean Observation,
a section of the journal
Frontiers in Marine Science

RECEIVED 15 November 2022
ACCEPTED 17 February 2023
PUBLISHED 10 March 2023

CITATION
Boulent J, Charry B, Kennedy MM,
Tissier E, Fan R, Marcoux M, Watt CA and
Gagné-Turcotte A (2023) Scaling whale
monitoring using deep learning: A
human-in-the-loop solution for
analyzing aerial datasets.
Front. Mar. Sci. 10:1099479.
doi: 10.3389/fmars.2023.1099479

COPYRIGHT
© 2023 Boulent, Charry, Kennedy, Tissier,
Fan, Marcoux, Watt and Gagné-Turcotte.
This is an open-access article distributed
under the terms of the Creative Commons
Attribution License (CC BY). The use,
distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Scaling whale monitoring using deep learning: A human-in-the-loop solution for analyzing aerial datasets

Justine Boulent[1], Bertrand Charry[1], Malcolm McHugh Kennedy[1], Emily Tissier[1], Raina Fan[1], Marianne Marcoux[2], Cortney A. Watt[2] and Antoine Gagné-Turcotte[1]*

[1]Whale Seeker, Montreal, Quebec, Canada, [2]Aquatic Research Division, Fisheries and Oceans Canada, Winnipeg, Manitoba, Canada

To ensure effective cetacean management and conservation policies, it is necessary to collect and rigorously analyze data about these populations. Remote sensing allows the acquisition of images over large observation areas, but due to the lack of reliable automatic analysis techniques, biologists usually analyze all images by hand. In this paper, we propose a human-in-the-loop approach to couple the power of deep learning-based automation with the expertise of biologists to develop a reliable artificial intelligence assisted annotation tool for cetacean monitoring. We tested this approach to analyze a dataset of 5334 aerial images acquired in 2017 by Fisheries and Oceans Canada to monitor belugas (*Delphinapterus leucas*) from the threatened Cumberland Sound population in Clearwater Fjord, Canada. First, we used a test subset of photographs to compare predictions obtained by the fine-tuned model to manual annotations made by three observers, expert marine mammal biologists. With only 100 annotated images for training, the model obtained between 90% and 91.4% mutual agreement with the three observers, exceeding the minimum inter-observer agreement of 88.6% obtained between the experts themselves. Second, this model was applied to the full dataset. The predictions were then verified by an observer and compared to annotations made completely manually and independently by another observer. The annotating observer and the human-in-the-loop pipeline detected 4051 belugas in common, out of a total of 4572 detections for the observer and 4298 for our pipeline. This experiment shows that the proposed human-in-the-loop approach is suitable for processing novel aerial datasets for beluga counting and can be used to scale cetacean monitoring. It also highlights that human observers, even experienced ones, have varied detection bias, underlining the need to discuss standardization of annotation protocols.

KEYWORDS

semantic segmentation, automated cetacean detection, active learning, wildlife monitoring, artificial intelligence

1 Introduction

Our ability to detect and identify wildlife is the foundation of all successful conservation and management plans, and research (Caughley, 1974; Pollock and Kendall, 1987; Yoccoz et al., 2001; Mackenzie et al., 2005). Conservationists, managers, and scientists increasingly rely on remote sensing data, such as satellite and aerial imagery to survey larger areas for tracking wildlife, and monitoring distribution, which can provide information on population trends over time (Fretwell et al., 2014; Cubaynes et al., 2019; Charry et al., 2020; Shah et al., 2020; Charry et al., 2021).

Cetaceans, composed of over 90 species of dolphins, whales, and porpoises, are central to our ocean ecosystems, contributing to nutrient cycling and carbon sequestration, and are viewed as keystone species to assess the overall health of our marine ecosystems (Wilkinson et al., 2003; Pershing et al., 2010). Scientists, conservationists, and other marine stakeholders traditionally rely on human marine mammal observers working with survey data collected from boats, aircraft, satellites, and other vessels to assess cetacean abundance. The use of aerial digital photography onboard manned and unmanned aircraft has yielded large amounts of data for assessing population distribution and demography (Heide-Jørgensen, 2004; Charry et al., 2018; Gray et al., 2019). However, the terabytes of photographs collected are tediously manually analyzed by humans; the lack of scalable, standardized, automated image analysis solutions limit the speed and cost-effectiveness of image-based surveys, as well as the mitigation and management goals they support.

During the last decade, the fields of ecology and conservation have benefited from the artificial intelligence (AI) and deep learning revolution, which has led to great advances in automatic wildlife recognition. Convolutional neural networks have been employed for several applications related to cetacean monitoring from images (Rodofili et al., 2022). Borowicz et al. (2019) used them to locate areas containing large whales in WorldView-3 satellite images. Lee et al. (2021) used convolutional neural networks to automate the detection of belugas (*Delphinapterus leucas*) in aerial images, also exploring the generalizability of a model on data collected in two different years. Berg et al. (2022) proposed a weakly supervised approach based on anomaly detection to detect marine animals, including cetaceans, in aerial images.

Despite these advances in image analysis, automating cetacean detection for aerial image datasets remains a challenge, notably due to the difficulty of building a rich enough dataset to train a generalizable model (Borowicz et al., 2019; Gray et al., 2019; Guirado et al., 2019; Gheibi, 2021; Lee et al., 2021; Berg et al., 2022; Rodofili et al., 2022). Firstly, image acquisition in marine environments is a costly and difficult task, especially for monitoring whale populations, as these animals are constantly on the move over an extremely large area and only surface intermittently. Secondly, marine environments are far from homogeneous, and undergo constant changes that can influence visual animal detection including sea state, water turbidity, and solar reflection. There are also several natural and anthropogenic objects that may be sources of confusion for computer vision analysis, such as rocks, seaweed, icebergs, floating waste, and boats. Lastly, cetaceans are challenging animals to observe even in the best of conditions, both for deep learning models and for biologists. For example, a whale's visibility depends on its posture and depth in the water column at the time of image acquisition (Figure 1). Given these constraints, datasets often gather hundreds of negative (no whales) images for only a few with whales, and at best cover a few species, geographic areas, and environmental conditions. Therefore, it is difficult to develop an automatic detection tool that is reliable.

In this study, we aimed to overcome these challenges by using a human-in-the-loop approach with the goal of combining speed and consistency of automated AI analysis with human's ability to generalize and deal with novelty. Human-in-the-loop can be defined as the set of strategies and techniques that associate human and machine intelligence to solve tasks automatically (Monarch et al., 2021). Overall, this combination aims to achieve expert-human-level accuracy with as little manual annotation time as possible. One of the pillars of human-in-the-loop is active learning. The assumption behind active learning is that not all samples have the same value when training a model, with some samples containing more significant information than others. For example, applied to beluga whale detection, images with objects likely to be confused with belugas are of greater interest than images with homogeneous



FIGURE 1
Examples of image diversity of belugas and narwhals in different environments and with varying estimated depths: **(A)** In surface waters, **(B)** Animals located between 0 and 1 meter from the surface, **(C)** Animals between 1 and 2 meters from the surface, **(D)** Animals deeper than 2 meters.

water, without confounding objects or rough waters. Therefore, by strategically selecting and annotating these most important samples, we can limit annotation effort while maximizing accuracy (Ren et al., 2021). A few studies have successfully applied active learning to wildlife monitoring, achieving high correct prediction rates while using fewer annotated examples than in classical transfer learning (Kellenberger et al., 2019; Miao et al., 2021).

We present a human-in-the-loop approach to partly automate cetacean detection from unannotated aerial images. The objective is not to develop a single model able to perform a perfect analysis, but to develop a methodology to efficiently assist biologists in the analysis of new aerial datasets, allowing for faster and more standardized results. To evaluate our approach, we applied it to aerial images of a beluga survey dataset from Fisheries and Oceans Canada (DFO) that was previously analyzed manually. In this study, we first trained a semantic segmentation model using active learning. On a test subset, we compared the model predictions with manual annotations of three observers. Once the model results reached human level quality, we analyzed the complete aerial dataset and compared the detections from the human-in-the-loop pipeline with the manual annotations.

2 Material and methods

2.1 Methods overview

Before diving into the details of the experiments, we provide a high-level description of the human-in-the-loop approach we adopted to assist marine mammal experts in the analysis of new incoming datasets of whale surveys. The method overview is intended to give an insight on the main components of the analysis, especially for readers not familiar with AI. For those readers, we also recommend the following references on the use of machine learning for wildlife monitoring (Weinstein, 2018; Tuia et al., 2022).

Our human-in-the-loop approach comprises three main steps:

(1) *Preliminary analysis* (Figure 2A): When a new dataset is received for analysis, limited *a priori* information is available – we do not have an estimate of the total number of whales, nor do we know the diversity of environmental conditions. These unknowns impede the use of AI and the initialization of the active learning loop. For active learning to be effective, it is necessary first to select examples of images including whales but also representative of the dataset's diversity, both to be able to train and evaluate the model. To overcome this issue and gather valuable information to start the active learning loop, we begin with a preliminary analysis based on generic deep learning models not trained on the new dataset. First, we use a land segmentation model and human verification to produce a binary land cover map. This map is used to exclude images covered entirely by land from further analysis, and to automatically dismiss predictions of whales made on land as false positives. Next, we use a dimensionality reduction technique, Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP, McInnes et al., 2018), to plot and cluster the environmental diversity of the dataset; this enables the selection of diverse and representative images to annotate, preventing manual analysis of redundant images during a single iteration. Finally, we run a model for cetacean segmentation trained

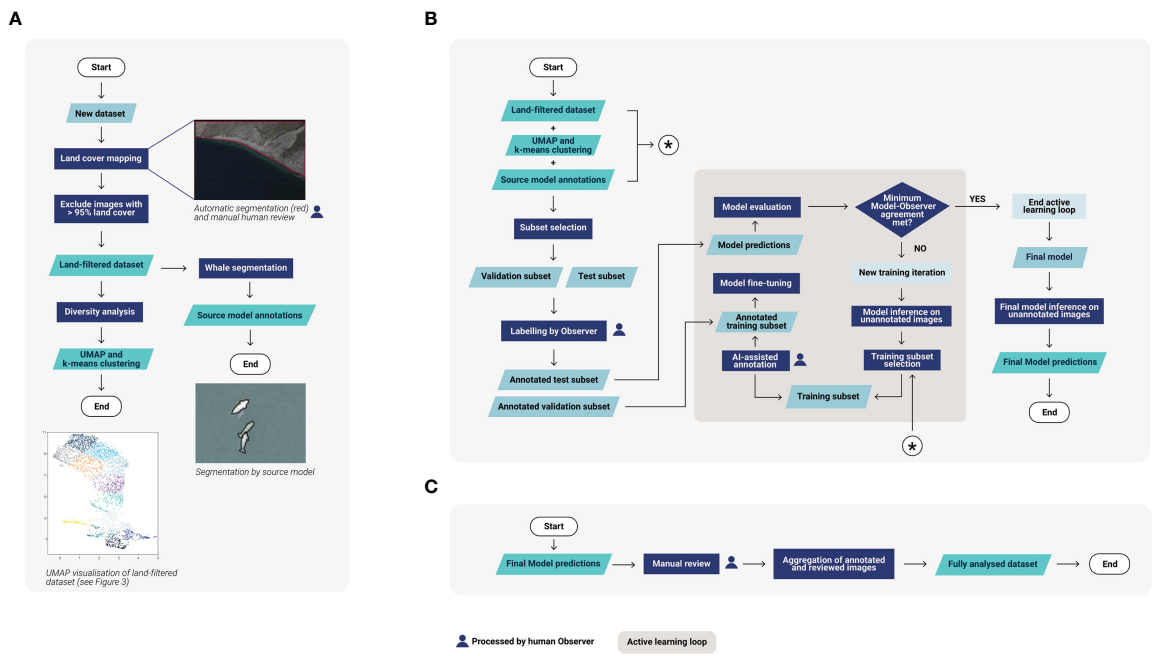


FIGURE 2
Overview of the human-in-the-loop pipeline. The pipeline is divided into three main stages: **(A)** The preliminary analysis, **(B)** The active learning loop, **(C)** The human review of the predictions.

on prior data (called the source model) on the new dataset, minus the images excluded covered entirely by land. Although its initial outputs are not accurate enough to be used as is, the outputs are used to find images containing potential cetaceans, providing a good starting subset for the active learning pipeline. For further details, see section 2.3.1 Preliminary analysis.

(2) *Active learning pipeline* (Figure 2B): To develop a cetacean segmentation model adapted to the new dataset without having to annotate a significant number of images, an active learning approach is adopted. Using the information from the preliminary analysis but without sharing the predictions with the human annotator, validation and test subsets are selected for manual annotation. Training images are also selected; however, this time, predictions are used for an AI-assisted annotation. Depending on the quality of the predictions, the human annotator either approves or corrects the targets detected by the model, or adds missing individuals. They also transform any false positives into negative examples, which are used for training in the next iteration. The whale source model is then fine-tuned using both the annotations from the new and the source datasets. Using this complementary source data serves to maintain the generalist features already present in the source model, and to provide enough whale examples for the fine-tuning, which is not always possible, as positive examples may be scarce in cetacean datasets. Similar iterations of "training images selection – images annotation – model fine-tuning and evaluation" are then repeated until satisfactory results are reached on the test subset (see section 2.3.2.1 Subsets selection and annotation). At this point, the fine-tuned model is used to analyze the whole dataset. For further details, see section 2.3.2 Active learning pipeline.

(3) *Human review of predictions* (Figure 2C): To improve the quality of the final analysis, a human annotator manually checks all the detections provided by the model and corrects them if necessary. For further details, see section 2.3.3 Human review of predictions.

In the entirety of this pipeline, the human annotator is involved in four tasks: (1) validating the segmentation of the land areas, (2) annotating validation and test images used to monitor the deep learning model, (3) annotating training images selected by active learning techniques, and (4) reviewing all predictions after the model's final analysis.

2.2 Data specification

2.2.1 Study area

The aerial survey was designed to detect and monitor beluga whales of the Cumberland Sound population in Clearwater Fjord, Canada. This population is composed of roughly 1,400 individuals (Watt et al., 2021) who are believed to reside year-round in Cumberland Sound, an Arctic waterway, based on information derived from telemetry data of 14 individuals (Richard and Stewart, 2008). During the open-water season in summer a large portion of this population congregates in Clearwater Fjord, located at the northern end of the sound (66°34' N, 67°26' W).

2.2.2 Data collection

In 2017, DFO conducted a photographic survey of the Cumberland Sound beluga population from 29 July to 12 August. Surveys were performed using a twin-engine Havilland Twin Otter 300 plane, flying at 100-110 knots at a goal altitude of 610m. Photographic surveys were performed over Clearwater Fjord following 26 pre-determined parallel transect lines 700m apart oriented east-west. To collect photographs a Nikon D810 camera, with 25mm lens, was mounted and positioned straight down at the rear of the aircraft to capture photographs. The camera was linked to a GPS receiver and was set to capture one photograph every seven to eight seconds. Each photograph covered an area of about 875m x 585m, with a 20% overlap on consecutive and adjacent photographs along transects. The photographs were acquired over four days flying over the same area.

2.2.3 Manual data analysis

The 5334 photographs of the area of interest were first examined to detect belugas by a photo-analyst from DFO, called Observer 3 in this paper. The analyst examined the georeferenced photographs using ArcMap 10.1 software by Esri. Each image was scanned and upon detection of a beluga whale a point annotation was added to the target in the image. Observer 3 detected 4572 beluga occurrences within the dataset. All detections noted in our study are whale targets in the images we processed; we did not remove duplicate targets detected in the overlap portions of images or interpret any abundance of these whale populations. Those annotations were only used for comparison with the results of our human-in-the-loop pipeline, not for training the pipeline.

Since this fully manual analysis was not conducted within this study, the time spent analyzing the dataset has not been recorded. However, it can be estimated that between 1328 hours (8 months working at 8 hours a day) and 2016 hours (12 months at 8 hours a day) were needed to perform this task without AI-assistance.

2.3 Detailed pipeline for experiments

2.3.1 Preliminary analysis

2.3.1.1 Land cover mapping

To automatically exclude images containing only land from our analysis, and automatically dismiss any predictions falling on land, we performed AI-assisted annotation to get a binary land segmentation mask for each image of the dataset. The land segmentation model used had a UNet50-ResNeXt architecture, and was trained on a dataset of 11,702 images from similar, but non-overlapping, Arctic surveys. This dataset was split into training, validation, and test subsets with ratios of 70%, 15%, and 15% respectively. The model was trained for 11 epochs, with a

learning rate of 2e-4. Loss was computed using the Log-Cosh Dice coefficient. Since this model was not fine-tuned on the new dataset, it made errors, especially in areas of shallow and muddy water, so we then manually vetted the predicted annotations, modifying any predictions that did not accurately reflect the observed coastlines.

2.3.1.2 Source whale model

A semantic segmentation model trained on another dataset, i.e., the source model, was used to find cetaceans in the first iteration. The source and new datasets differ in flight altitude, geographic area covered, and predominant species found. The source dataset was acquired by DFO in 2013, over the Canadian Arctic Archipelago, with a target flight altitude of about 305m. In 1562 images, 10,253 cetaceans were annotated. They consisted mostly of narwhals (about 80%), but also belugas (about 20%) and bowhead whales (less than 1%).

To train the source whale model, images from the source dataset were split into training, validation, and test subsets with ratios of 70%, 15%, and 15% respectively. This split was done randomly, but with the constraint that two images with a geospatial overlap could not be in different subsets, so as to prevent cross-contamination. A supervised training was carried out, using a U-Net architecture (Ronneberger et al., 2015), and with EfficientNet-b3 (Tan and Le, 2019) as an encoder. It was trained for 50 epochs with an initial learning rate of 2e-4. The optimizer used was AdamW and the loss was computed with the Dice coefficient. Of the 1658 whales in the 234 test images, 1568 were segmented by the model, giving a recall of 94.6% at 95.66% of precision. For more details on the metrics used, refer to the section 2.3.4 Metrics.

2.3.1.3 Diversity analysis

In order to minimize redundancy in the images sent for manual annotation, and hence the number of iterations to reach the stopping criterion, the automatic selection of the images to annotate was done in such a way that represented the diversity of oceanic environments seen across all images.

To do this, we first ran all the images in the dataset through an off-the-shelf pre-trained convolutional neural network (ResNet-50 (He et al., 2016) from TorchVision), and extracted the final activation layer after a forward pass through the network. The activation layer for each image was then fed into a nonlinear dimensionality reduction tool, UMAP (McInnes et al., 2018), which is designed to reduce the dimensionality of high-dimensional data, while retaining some of the meaningful characteristics of the data, such as similar elements clustering together across space. We chose to reduce the representation of each image to two dimensions, to enable human-readable visualizations (Figure 3). The two-dimensional representations did indeed cluster similar environmental conditions together in space, so that images dominated by land cover, shallow water, white caps, or muddy water, for instance, clustered in contiguous regions of the 2D space.

To use this information for image sampling and based on a visual assessment of the UMAP representation, we binned the images into 12 discrete clusters using the k-means clustering algorithm, assigning each image in the dataset an arbitrary number according to which environmental cluster it fell into.

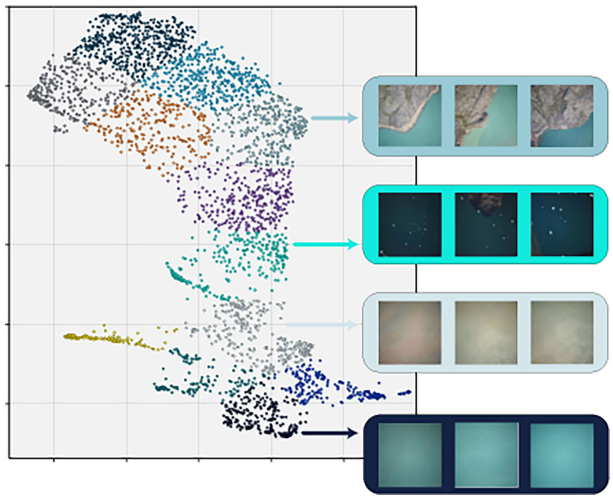


FIGURE 3
Clustered UMAP visualization of the new dataset. Each point represents an image and is color-coded by cluster. To illustrate the similarity of images within clusters, rectangles on the right show examples of three random images from four different clusters.

Using this representation, images were picked successively and randomly from the different clusters to obtain a representative selection of the environmental diversity.

2.3.2 Active learning pipeline

2.3.2.1 Subsets selection and annotation

2.3.2.1.1 Validation and test subsets

Creating validation and test subsets including whales was challenging, since no *a priori* knowledge on the dataset was used. Random sampling would have likely yielded subsets without any whales, and that did not represent the dataset's true range of environmental diversity. For this reason, we relied on the preliminary analysis results. For each of the test and valid subsets, 50 images were selected successively and randomly, alternating between the different UMAP clusters to provide representative sampling of environmental diversity. The selection algorithm also ensured that two images with space-time overlap were not in different subsets. For 20 images of each subset, another selection rule was imposed using the predictions made by the source whale model: these images had to contain at least two predictions of whales scoring above 60% confidence to be selected. Although there is some bias in this approach since the source model's predictions were used to select images for its own evaluation, it was the best way to ensure we included cetaceans in validation and testing, without having to manually evaluate the dataset. Since belugas live in groups, selecting an image with at least two predicted whales generally gave access to a larger group, including whales not detected by the model. Moreover, as the source model was not yet adapted to the target domain, the selections also included false positives. Using a selection of images that included not only true positives, but also false predictions enabled us to automatically create validation and test subsets capable of tracking the evolution of the model's fine-tuning. Following the selection of images for the validation and test subsets,

we proceeded to annotate them. One of the challenges of AI for wildlife monitoring is that the ground truth is based on human annotations, and therefore contains some degree of difference, owing to inter- and intra-observer variability. To calculate the variability of annotation between different expert marine mammal biologists, the test subset was analyzed independently by three observers (Table 1) (see section 2.3.4.2 Measuring agreement for further details). Only the test subset was analyzed by multiple observers as it contained a representative sample of environmental diversity of the full dataset and to limit the annotation workload. Observer 1, a Whale Seeker biologist, was the primary annotator, since in addition to the test subset, they also annotated the validation and train subsets, as well as doing the final prediction reviews. Observer 2 was also a Whale Seeker biologist. They both used the annotation software DIVE to draw individual polygons around each whale. Observer 3 was a DFO biologist who had previously annotated the entire dataset (see section 2.2.3 Manual analysis). Since the annotations from Observers 1 and 2 were individualized polygons while those from Observer 3 were points centered on the whales, we transformed these points into a 2*2 pixels square to allow comparison. Hence, a polygon intersecting a square is considered as a common annotation between observers.

Using the test-set annotations of the three observers, we calculated their inter-observer agreement, a key metric in a context where there is no real ground truth. This metric was used as the stopping criterion of the active learning loop: the loop would be ended once the agreement between the model predictions and the human annotations equaled or exceeded this value.

2.3.2.1.2 Training subsets

At each iteration, 50 images were selected to be annotated for fine-tuning. To sample images with the most uncertain targets, we used the least confidence criterion (Monarch et al., 2021) to select 20 images based on the confidence score of the predicted targets. An additional 25 images were selected using a most confidence criterion. This criterion is based on the number of targets in an image with a confidence above a specified threshold value, in this case 90%. This criterion had the advantage of generating true whale predictions that can be easily transformed into annotations when the segmentation has a high enough quality. It also allowed us to catch false positives with a high level of confidence, a frequent occurrence when analyzing new environments. Since we were selecting entire images and not just targets, this criterion provided access to a large number of beluga whales, and thereby potentially to false negatives. Finally, five images were also randomly selected for annotation. To avoid redundancy of information, we used the UMAP representation to select the images.

The annotation was performed by Observer 1 with the model's assistance, i.e., the observer had access to the predictions of the model to speed up analysis. To enrich the pool of negative examples sent to the model during training, we followed a hard negative mining approach, which means we transformed the false positives from selected images into negative examples for the next training iteration. Since the dataset images measured 7360 per 4912 pixels — too large to be fed directly into machine learning algorithms — tiles of 256 per 256 pixels were extracted around each whale and hard negative example. To complete the dataset, negative tiles were also extracted randomly. To avoid an unbalanced dataset, the same number of positive and negative tiles were fed to the model. Because positive examples are typically scarce in cetacean surveys, 750 positive examples from the source dataset were also selected randomly to supplement those from the new dataset. A summary of the data used in each iteration can be found in Table 2.

2.3.2.2 Model fine-tuning

A complete fine-tuning of the previously trained model was performed on each iteration. For the first iteration, the starting point was the source model. We used a U-Net architecture with an EfficientNet-B3 encoder. During each training phase, several runs were performed with different random seed states. Since the human annotator only verifies images that contain at least one whale prediction, we needed a fairly sensitive model. For each iteration, between all the models from the different runs, we chose the model with the best recall for an accuracy over 85%. More details about the hyperparameter values used can be found in Table 3.

2.3.3 Human review of predictions

Once the stopping criterion was reached, the final iteration of the model was used for inference on all remaining unannotated images. The list of images with at least one whale detected was then sent to Observer 1 for manual revision. During this process, the observer could approve, remove, or correct the predictions. They could also add targets not predicted by the model, and separated groups of whales that were segmented as one by the model, to facilitate an individual count of the number of cetaceans.

2.3.4 Metrics

2.3.4.1 Computer vision metrics

To evaluate the performance of the models, precision (Eq. 1), recall (Eq. 2), and F1-score (Eq. 3) were calculated. For our application, since it was not the quality of the segmentations that

TABLE 1 Summary of annotations for the validation and test subsets according to the three observers.

Subset type	Number of images per subset	Number of annotated whales per subset		
		Observer 1	Observer 2	Observer 3
Validation	50	390	N/A	N/A
Test	50	289	304	315

N/A stands for "not applicable".

TABLE 2 Summary of the data used in each training iteration.

	Iteration 1	Iteration 2
Annotated images	50 *(+50)*	100 *(+50)*
Positive tiles		
Annotated whales from the DFO dataset	768 *(+768)*	1283 *(+515)*
Annotated whales from the source domain	750 *(N/A)*	750 *(N/A)*
Total of positive tiles	1518 *(+768)*	2033 *(+515)*
Negative tiles		
Hard negative tiles	157 *(+157)*	301 *(+144)*
Random negative tiles	1361 *(+1361)*	1732 *(+371)*
Total of negative tiles	1518 *(+1518)*	2033 *(+515)*

All training annotation was performed by Observer 1. The numbers displayed represent the cumulative total number of images or annotations used for each iteration. The numbers in brackets and italics represent the number of new images or annotations added for each iteration.

was important but rather binary detection quality, these three metrics were computed not at the pixel but at the target level. Each group of contiguous positive pixels was considered a target. Each whale prediction that intersected a human annotation was counted as a true positive. Recall is the most critical metric for this application since we focus on missing as few individuals as possible. High precision is nonetheless important so that the observer does not spend too much time checking for false positives.

$$Precision = \frac{True\ \ Positives}{True\ \ Positives\ +\ False\ \ Positives} \quad (1)$$

$$Recall = \frac{True\ \ Positives}{True\ \ Positives\ +\ False\ \ Negatives} \quad (2)$$

$$F1 - Score = \frac{2\ \times\ Precision\ \times\ Recall}{Precision\ +\ Recall} \quad (3)$$

TABLE 3 Hyperparameters used to fine-tune the model.

Architecture	U-Net with Efficient-Net B3 as encoder (Ronneberger et al., 2015; Tan and Le, 2019) https://github.com/qubvel/segmentation_models.pytorch
Initial Learning Rate	1e-5 to 6e-4
Optimizer	AdamW
Loss function	Dice Coefficient
Batch Size	30
Maximum number of epochs	30
Transformations	Randomly applied: rotation in 90-degree steps, horizontal or vertical flip, and hue color jitter

2.3.4.2 Measuring agreement

One challenge of quantifying automated approach success using remote detection is the inherent variability in ground-truth data, both between expert human observers and within the same observer. Numerous studies across various taxa have measured inter-observer variability in overall animal counts given the same remote sensing imagery (Linchant et al., 2015; Wanless et al., 2015; Schlossberg et al., 2016; Fossette et al., 2021). These studies report count discrepancies in the range of 5 - 15%. Disagreement across matched detections (rather than the overall count) is less well documented but is likely significantly higher.

This range of inter-observer variability, even among experts, makes 100% recall and precision a moving target, and not a realistic or desirable goal for automated or manual approaches. Instead, an automated solution's recall and precision can instead be interpreted as the algorithm's "agreement" with the observer who created the ground-truth annotations, and can be expected, at best, to approach the agreement values human experts have with respect to one another. Specifically, we defined agreement between two observers (human or computer) as the intersection over union (IOU) between them, which is the number of shared detections divided by the size of the union of the two observer's detections (Eq. 4).

$$Inter\ -\ observer\ \ agreement = \frac{Detections_{ObsA,\ \ ObsB}}{Detections_{ObsA,\ \ ObsB}\ +\ Detections_{ObsA}\ +\ Detections_{ObsB}} \quad (4)$$

Where $Detections_{ObsA,\ ObsB}$ represents the number of whales detected by both observers, while $Detections_{ObsA}$ represents the number of detections made only by Observer A, and $Detections_{ObsB}$ represents the number of detections made only by Observer B.

We chose this metric since, unlike concepts such as recall and precision, it is symmetric between the two observers, rather than assuming one to be ground truth.

3 Results

3.1 Land cover exclusion

Using the land use mapping done in the preliminary analysis, 1977 images (37% of the total) were excluded from further analysis because they were covered by more than 95% land, leaving 3357 images to be analyzed for whales.

3.2 Evaluation on the test subset

3.2.1 Inter-observer agreement

The number of whales found in the 50 test images varied between observers. Observer 1 was the most conservative annotator, disregarding targets that were deep in the water column, whereas Observer 3 was less conservative and included deep-water targets. Therefore, the number of whales detected in the 50 images ranged between 239 to 315. The percentage of agreement between pairs of observers ranged from 88.5% to 92.88% (Table 4).

Most of the disagreements between observers concerned targets that might be whales swimming deep in the water column (Figures 4A, B). Some discrepancies were due to targets resembling waves (Figure 4C) or birds (Figure 4D).

3.2.2 Active learning loop performance

Two iterations, totaling 100 annotated images (~2% of the complete dataset), enabled the model to exceed the minimum inter-observer agreement value on the test subset, with model–observer agreement percentages ranging from 90.03% to 91.37% (Table 5; Figure 5).

Despite differences between the source and new datasets, the source model provided an initial recall on the test subset ranging from 75.87% to 79.93% depending on the observer. The incorporation of target domain annotations greatly improved the detection capabilities: the number of false negatives shrank more than sixfold between the source model and the iteration 1 model. After iteration 2, the recall ranged from 94.75% to 98.96%. Interestingly, across all the false negatives, none had consensus by all three observers, highlighting the alignment between inter-observer discrepancies and model-observer discrepancies. Precision increased by an average of 28.8 percentage points after 50 annotated images were added. This upward trend continued less steeply between iteration 1 and 2, with an average gain of 4.23 percentage points. After iteration 1, some of the false positives were recognizable objects like rocks, glare effects and waves, but after iteration 2, the false positives related to objects that we couldn't identify. All three observers agreed on only 7 of the false positives, and some of them could indeed be belugas that were missed by all three (Figure 6).

3.3 Evaluation on the whole dataset

Once the active learning loop was complete, Observer 1 proceeded to the final step of the pipeline: reviewing the predictions on the remaining 3157 images that had not been manually annotated. In this review, 572 predictions were removed, and 58 detections were added.

The annotations from the human-in-the-loop pipeline were then compared with those made without AI assistance by Observer 3. In total, 4298 belugas were detected by the pipeline, while the Observer 3 detected 4572 belugas, a difference of 274 individuals. The level of mutual agreement reached 84%, representing 4051 mutual detections. Observer 1 detected 247 belugas that were not detected by Observer 3, and Observer 3 detected 521 belugas that were not detected by Observer 1.

As no third-party biologist reviewed the disagreements, we were not able to arbitrate on the presence or absence of belugas. Nevertheless, to better understand the disagreements between the human-in-the-loop pipeline and Obsrver 3 detections, Observer 1 manually inspected the discrepancies.

Out of the 768 targets in disagreement, he assessed that 60% of them could not be annotated with certainty, due to a lack of visibility, related to the turbidity of the water, the conditions at sea, and especially, to the depth of the detected target (Figure 7). While image annotation protocols generally specify a maximum depth for a target to be counted as a whale, in practice it is difficult to follow these guidelines, which leaves room for some interpretation. When analyzing groups of whales, we noticed that observers were inclined to annotate targets at great depths as belugas, while similar targets outside whale groups were not annotated as such. About 35% of the uncertain targets were found in beluga whale groups. The proximity of the belugas and the turbulence they create rendered individualization difficult (Figure 7).

TABLE 4 Annotation agreement on the test subset between the three observers.

	Agreement (%)	Number of mutual whales' detections	Number of whales found only by the 1st Observer	Number of whales found only by the 2nd Observer
Obs. 1 – Obs. 2	92.9	287	2	20
Obs. 1 – Obs. 3	88.6	285	7	30
Obs. 2 – Obs. 3	92.9	300	8	15

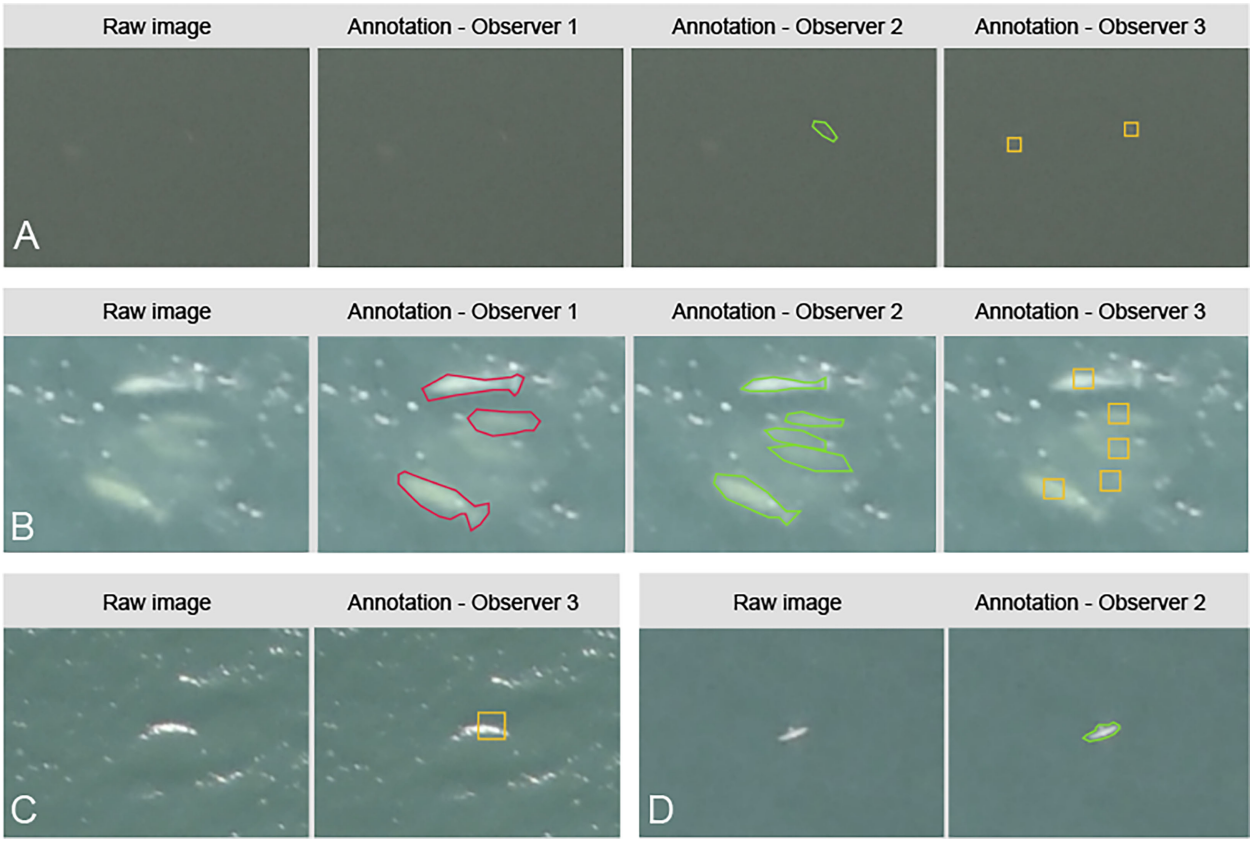


FIGURE 4
Examples of annotation disagreement in the test subset. **(A, B)** Targets annotated differently by each observer with red outlines for Observer 1, green outlines for Observer 2, and yellow boxes for Observer 3. **(C)** Target annotated only by Observer 3. **(D)** Target annotated only by Observer 2.

3.4 Time-tracking

We tracked the time spent by Observer 1 annotating images and reviewing predictions to estimate the time needed for an observer to analyze a dataset while being assisted by the human-in-the-loop pipeline (Figure 8). In total, 53 hours were spent for the complete analysis of this dataset of 5534 images. The AI-assisted annotation of the land took approximately 23 hours, given that about 80% of the images included land. Whale detection required approximately 31 hours of manual work to analyze the eligible 3357 images (i.e.,

TABLE 5 Summary of the results between the model and the three observers on the test subset.

	Agreement (%)	F1-score (%)	Recall (%)	Precision (%)	FP	FN	TP
Observer 1							
Source model	52.14	68.55	79.93	60.00	154	58	231
Iteration 1	87.11	93.11	98.27	88.47	37	5	284
Iteration 2	**91.37**	95.49	98.96	92.26	24	3	286
Observer 2							
Source model	51.42	75.87	75.87	61.60	149	76	239
Iteration 1	85.45	93.65	93.65	90.77	30	20	295
Iteration 2	**90.96**	94.75	94.60	94.90	16	17	298
Observer 3							
Source model	51.50	67.92	76.97	60.78	151	70	234
Iteration 1	85.50	92.16	94.74	89.72	33	16	288
Iteration 2	**90.03**	95.27	96.05	94.5	17	12	292

In bold, the agreement values exceeding the minimum inter-observer agreement. FP, false positives; FN, false negatives; TP, true positives.

10.3389/fmars.2023.1099479

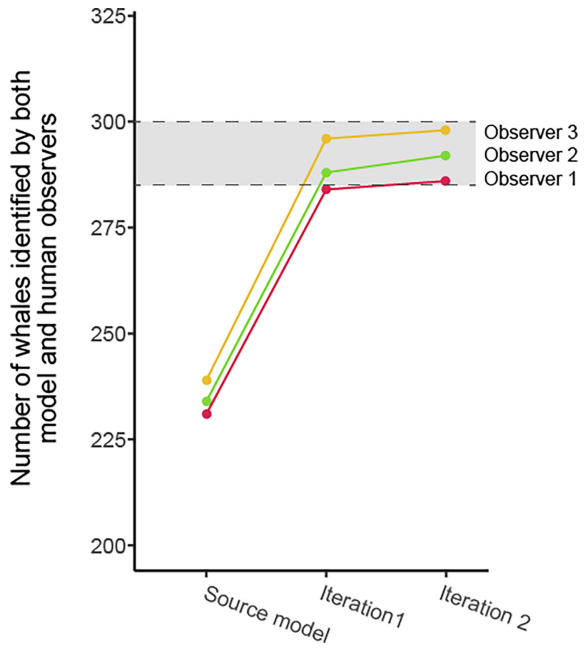


FIGURE 5
Evolution of the observer–model whale detection agreement on the test subset through the model training iterations. The shaded area between hash lines indicates the inter-observer minimum and maximum whale detection agreement.

with a land cover under 95%). Given that a fully manual analysis took an estimated 1328 to 2016 hours, the time savings for the observer using our AI-assisted approach are in the range of 96-97%.

4 Discussion

4.1 Scaling the adoption of AI for aerial whale monitoring

Our study presents an original deep learning-based solution using a human-in-the-loop framework to detect whales from aerial imagery. AI-assisted detection can process imagery significantly faster than manual detection, thereby providing more time for interpretation and development of mitigation strategies. Manual analysis of a survey can take months or years, delaying evaluation of mitigation plans, which can be detrimental to the species of interest.

Although there has been previous work using deep learning to analyze imagery of marine mammals, they have not yet gained traction with the global community of wildlife managers and other ocean stakeholders. While data democratization is often put forward as a roadblock to implement AI solutions in ecology (Ditria et al., 2022), another major challenge is the lack of knowledge sharing and understanding between AI experts and wildlife managers. Creating a widespread usable framework not only requires deep expertise and communication from multiple disciplines such as computer science and ecology, but also the involvement of all marine stakeholders.

Full photographic surveys are desirable in the field because they are cost-effective, requiring fewer personnel, which also means less human risk; however, processing vast amounts of imagery that are acquired is a major bottleneck. Our methodology, including the use of UMAP to select the most impactful data for re-training, helps to make full photographic surveys a viable monitoring solution, by cutting down the number of manual annotations needed for re-training.

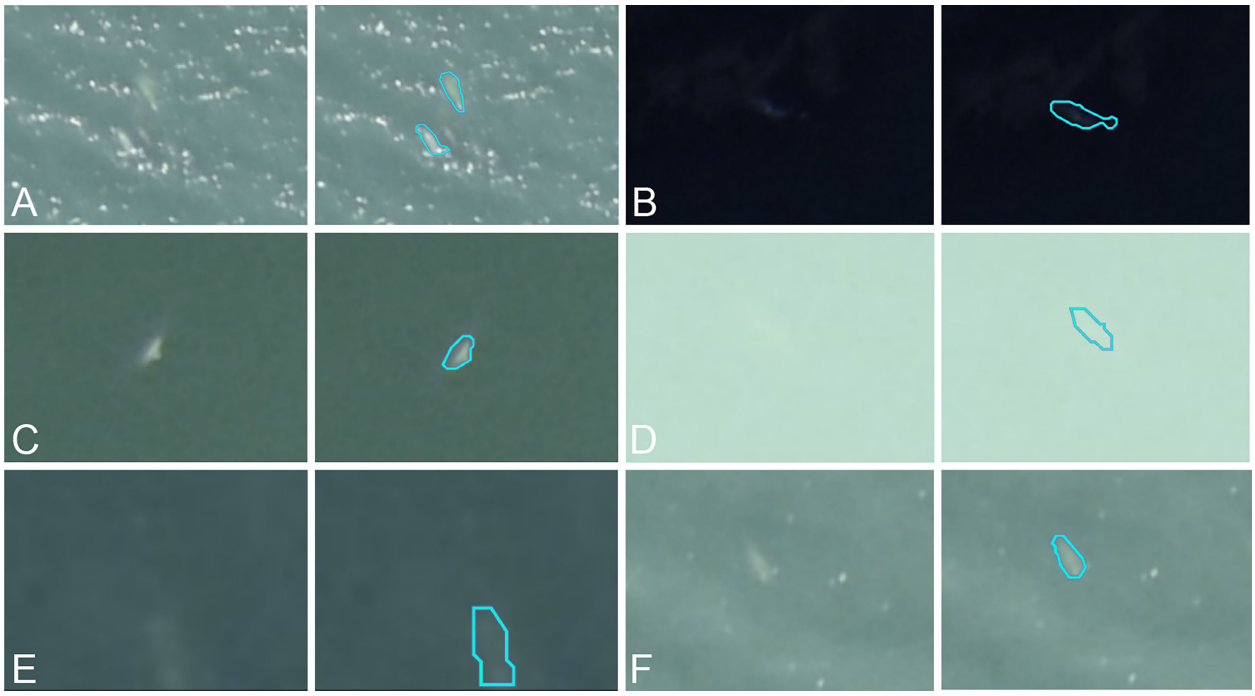


FIGURE 6
(A–F) Examples of predictions (original image on left, turquoise outline prediction on right) from iteration 2 identified as false positives for all three observers.

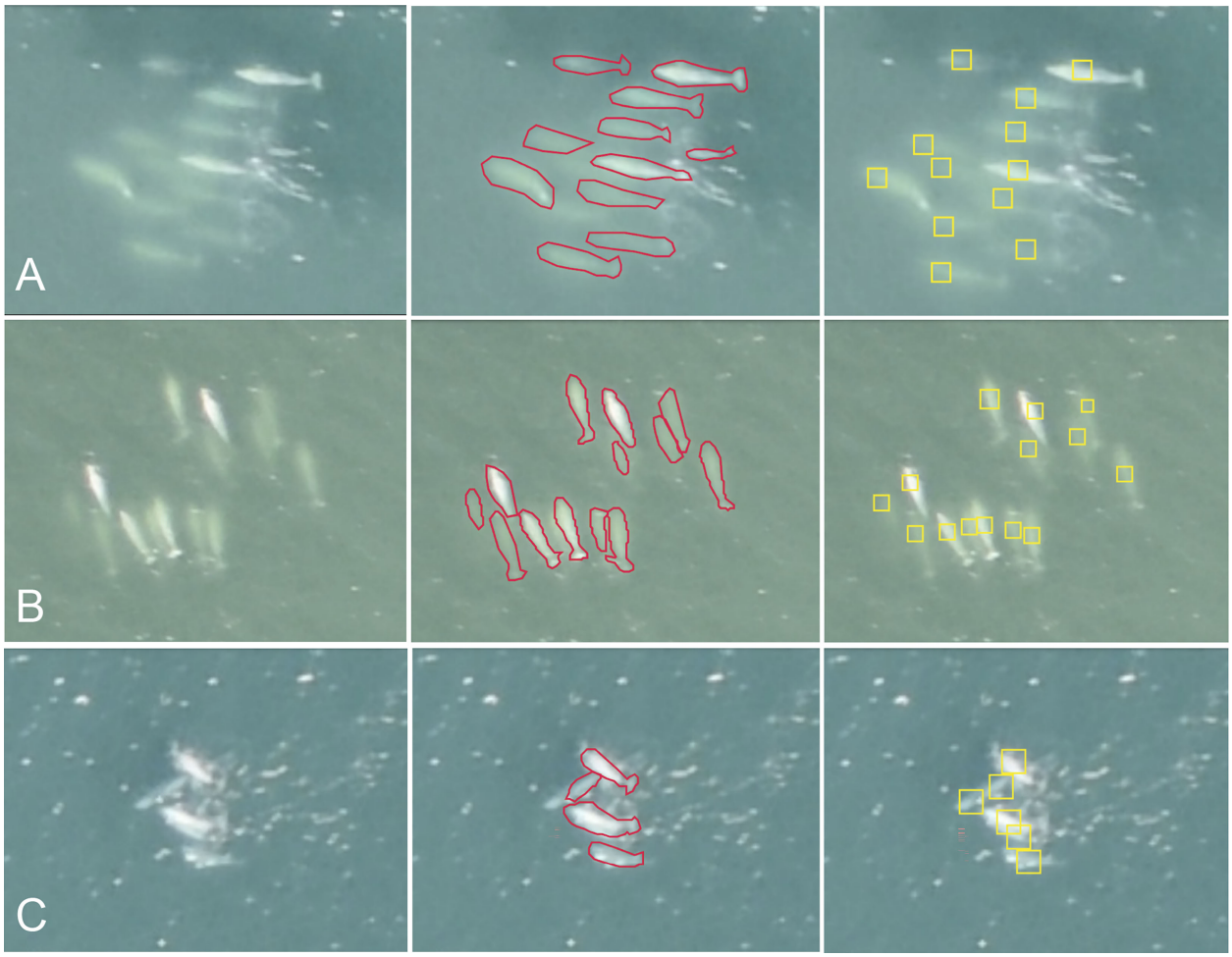


FIGURE 7
Examples of annotation disagreements between Observer 1 (middle, in red) and Observer 3 (right, in yellow). Original unannotated image on left. Total count of belugas in **(A)** Observer 1: 11, Observer 3: 12; **(B)** Observer 1: 13, Observer 3: 14; **(C)** Observer 1: 4, Observer 3: 6.

Since each dataset is different, it is expected that the time an expert spends on each AI-assisted analysis will vary. The greatest time savings will likely be for repeated surveys from one year to the next, or for analyzing historical datasets, where the target species and geographic area are constant.

4.2 The need of standardization and transparency

By analyzing a dataset with a single model, AI improves standardization: each image is processed identically, without the biases

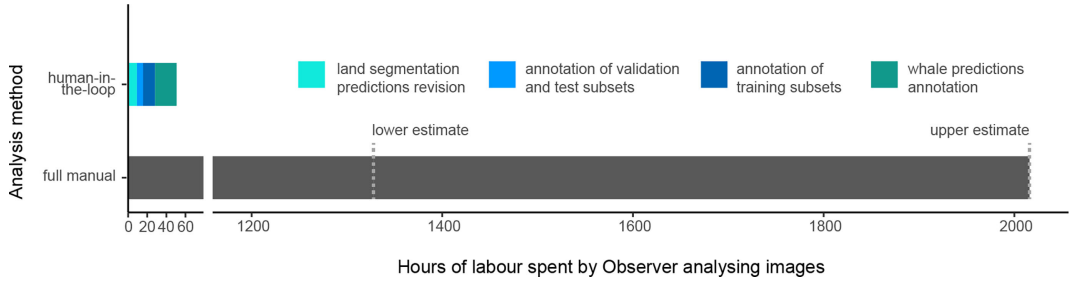


FIGURE 8
Comparison of the time spent by Observer 1 to analyze the dataset with the AI-assisted approach versus the time spent by Observer 3 to analyze the dataset fully by hand. The exact time spent for the full manual analysis was not recorded, hence the lower and upper estimates of the time needed to analyze a dataset of 5334 images.

and variability that can occur during manual annotation. However, this approach does not mean we can do without observers' intervention: their expertise is required for fine-tuning data as well as prediction verification. Therefore, the consistency of an AI solution is limited by the consistency of manual interventions and establishing a robust manual annotation protocol from the outset is essential, especially regarding common conditions for inter-observer discrepancy such as deep targets and murky water. Standardization of protocols for assessing difficult cases would ensure temporally spaced surveys are consistent, even if they cannot be ground-truthed. As the AI-assisted annotation process greatly reduces the time taken by observers to analyze the images, multiple observers could be asked to review the annotations and arbitrate the difficult cases. Because marine mammal management often has large environmental, monetary, and cultural implications, a standardized approach offers transparency for stakeholders and can go a long way to developing trust in the scientific process.

4.3 AI perspectives

Improvements can be made to the pipeline presented here. Going from semantic segmentation to an approach that isolates individuals could speed up the manual revision process. However, this approach needs to be robust to the proximity, and even overlap, of individuals. Developing a source model with a higher generalization capacity would also be an improvement since better pre-analysis requires fewer active learning iterations. Improving generalization remains an area of ongoing research (Wang et al., 2021). Developing specialized source models for given species and geographic areas could also improve the pre-analysis results. Finally, extending the model's scope from whale detection to species identification would allow for better monitoring of multiple species within the same geographical area.

5 Conclusion

In this study, we proposed and applied a human-in-the-loop approach to address the challenge of a real-world cetacean monitoring application case: analyzing a novel dataset of aerial images for beluga whale monitoring. Through this approach and the close collaboration between AI and the observer, expert-quality analysis was quickly provided for the 5334 images in the dataset, with only 100 annotated images for training. Generalization of this approach to aerial image analysis could significantly improve cetacean monitoring in quantity and quality. Keeping the expert in the loop ensures human-level quality results and better adaptation to new environmental and biological conditions in the imagery. Using computing power instead of total human analysis also allows more data to be analyzed in a dramatically shorter time period, allowing more meaningful time sensitive decisions. Improvements can still be made to the proposed method, both for AI (better generalization of source models, multi-species identification) and for cetacean monitoring methodology (standardized taxonomy and image annotation protocol), and yet the human-in-the-loop approach proposed here constitutes a first innovative and practical solution for automating imagery analysis for cetacean monitoring.

Data availability statement

The data analyzed in this study is subject to the following licenses/restrictions: Crown Copyright. Requests to access these datasets should be directed to CW, cortney.watt@dfo-mpo.gc.ca.

Author contributions

JB, MK, ET and AG-T conceived the ideas. JB, MK, and AG-T designed the methodology. MM and CW collected the data. BC and RF annotated manually the data. JB, MK and AG-T implemented the designed methodology and proceeded to the automatic analysis of the dataset. JB, BC, ET and MK led the writing of the manuscript. All authors contributed to the article and approved the submitted version.

Funding

The Polar Continental Shelf Program, Fisheries and Oceans Canada, Species at Risk and the Nunavut Wildlife Management Board financially supported imagery acquisition. Whale Seeker financially supported the data analyses.

Acknowledgments

Thanks to L. Montsion for manual image annotation. Thanks to the community of Pangnirtung, the Pangnirtung Hunters and Trappers Association, C. Matthews, B. Dunn, M. Ghazal, and C. Hornby for photographic acquisition.

Conflict of interest

Authors JB, MK, and RF are employed by Whale Seeker, a B-corp company specialized in marine mammal detection that was founded by authors BC, AG-T and ET. Whale Seeker sells an image analysis service for the detection of marine mammals whose artificial intelligence-assisted annotation tool, Mobius, is associated with this research.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The reviewer AB declared a past collaboration with the author ET to the handling editor.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Berg, P., Santana Maia, D., Pham, M.-T., and Lefèvre, S. (2022). Weakly supervised detection of marine animals in high resolution aerial images. *Remote Sens.* 14, 339. doi: 10.3390/rs14020339

Borowicz, A., Le, H., Humphries, G., Nehls, G., Höschle, C., Kosarev, V., et al. (2019). Aerial-trained deep learning networks for surveying cetaceans from satellite imagery. *PloS One* 14, e0212532. doi: 10.1371/journal.pone.0212532

Caughley, G. (1974). Bias in aerial survey. *J. Wildlife Manage.* 38, 921–933. doi: 10.2307/3800067

Charry, B., Marcoux, M., Cardille, J. A., Giroux-Bougard, X., and Humphries, M. M. (2020). Hierarchical classification of narwhal subpopulations using social distance. *J. Wildlife Manage.* 84, 311–319. doi: 10.1002/jwmg.21799

Charry, B., Marcoux, M., and Humphries, M. M. (2018). Aerial photographic identification of narwhal (Monodon monoceros) newborns and their spatial proximity to the nearest adult female. *Arctic Sci.* 4, 513–524. doi: 10.1139/as-2017-0051

Charry, B., Tissier, E., Iacozza, J., Marcoux, M., and Watt, C. A. (2021). Mapping Arctic cetaceans from space: A case study for beluga and narwhal. *PloS One* 16, e0254380. doi: 10.1371/journal.pone.0254380

Cubaynes, H. C., Fretwell, P. T., Bamford, C., Gerrish, L., and Jackson, J. A. (2019). Whales from space: Four mysticete species described using new VHR satellite imagery. *Mar. Mammal Sci.* 35, 466–491. doi: 10.1111/mms.12544

Ditria, E. M., Buelow, C. A., Gonzalez-Rivero, M., and Connolly, R. M. (2022). Artificial intelligence and automated monitoring for assisting conservation of marine ecosystems: A perspective. *Front. Mar. Sci.* 9, 918104. doi: 10.3389/fmars.2022.918104

Fossette, S., Loewenthal, G., Peel, L. R., Vitenbergs, A., Hamel, M. A., Douglas, C., et al. (2021). Using aerial photogrammetry to assess stock-wide marine turtle nesting distribution, abundance and cumulative exposure to industrial activity. *Remote Sens.* 13, 1116. doi: 10.3390/rs13061116

Fretwell, P. T., Staniland, I. J., and Forcada, J. (2014). Whales from space: Counting southern right whales by satellite. *PloS One* 9, e88655. doi: 10.1371/journal.pone.0088655

Gheibi, M. (2021). *Helping biologists find whales: AI-in-the-Loop support for environmental dataset creation* (Halifax, Canada: Dalhousie University). (Master thesis).

Gray, P. C., Bierlich, K. C., Mantell, S. A., Friedlaender, A. S., Goldbogen, J. A., and Johnston, D. W. (2019). Drones and convolutional neural networks facilitate automated and accurate cetacean species identification and photogrammetry. *Methods Ecol. Evol.* 10, 1490–1500. doi: 10.1111/2041-210X.13246

Guirado, E., Tabik, S., Rivas, M. L., Alcaraz-Segura, D., and Herrera, F. (2019). Whale counting in satellite and aerial images with deep learning. *Sci. Rep.* 9, 14259. doi: 10.1038/s41598-019-50795-9

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition, in: 2016 IEEE conference on computer vision and pattern recognition (CVPR)," in *Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE*, Las Vegas, NV, USA. 770–778. doi: 10.1109/CVPR.2016.90

Heide-Jørgensen, M. P. (2004). Aerial digital photographic surveys of narwhals, monodon monoceros, in northwest Greenland. *Mar. Mammal Sci.* 20, 246–261. doi: 10.1111/j.1748-7692.2004.tb01154.x

Kellenberger, B., Marcos, D., Lobry, S., and Tuia, D. (2019). Half a percent of labels is enough: Efficient animal detection in UAV imagery using deep CNNs and active learning. *IEEE Trans. Geosci. Remote Sens.* 57, 9524–9533. doi: 10.1109/TGRS.2019.2927393

Lee, P. Q., Radhakrishnan, K., Clausi, D. A., Scott, K. A., Xu, L., and Marcoux, M. (2021). Beluga whale detection in the Cumberland sound bay using convolutional neural networks. *Can. J. Remote Sens.* 47, 276–294. doi: 10.1080/07038992.2021.1901221

Linchant, J., Lhoest, S., Quevauvillers, S., Semeki, J., Lejeune, P., and Vermeulen, C. (2015). Wimuas: Developing a tool to review wildlife data from various uas flight plans. *ISPRS - Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci. XL3*, 379–384. doi: 10.5194/isprsarchives-XL-3-W3-379-2015

Mackenzie, D. I., Nichols, J., Sutton, N., Kawanishi, K., and Bailey, L. L. (2005). Improving inferences in population studies of rare species that are detected imperfectly. *Ecology* 86, 1101–1113. doi: 10.1890/04-1060

McInnes, L., Healy, J., Saul, N., and Großberger, L. (2018). UMAP: Uniform manifold approximation and projection. *J. Open-Source Software* 3, 861. doi: 10.21105/joss.00861

Miao, Z., Liu, Z., Gaynor, K. M., Palmer, M. S., Yu, S. X., and Getz, W. M. (2021). Iterative human and automated identification of wildlife images. *Nat. Mach. Intell.* 3, 885–895. doi: 10.1038/s42256-021-00393-0

Monarch, M., Munro, R., and Monarch, R. (2021). Human-in-the-Loop machine learning: Active learning and annotation for human-centered AI. *Simon Schuster*.

Pershing, A. J., Christensen, L. B., Record, N. R., Sherwood, G. D., and Stetson, P. B. (2010). The impact of whaling on the ocean carbon cycle: Why bigger was better. *PloS One* 5, e12444. doi: 10.1371/journal.pone.0012444

Pollock, K. H., and Kendall, W. L. (1987). Visibility bias in aerial surveys: A review of estimation procedures. *J. Wildlife Manage.* 51, 502–510. doi: 10.2307/3801040

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B. B., et al. (2021). A survey of deep active learning. *ACM Comput. Surv.* 54, 180:1–180:40. doi: 10.1145/3472291

Richard, P., and Stewart, D. B. (2008). *Information relevant to the identification of critical habitat for Cumberland sound belugas (Delphinapterus leucas) (No. 2008/085)* (Canadian Science Advisory Secretariat).

Rodofili, E. N., Lecours, V., and LaRue, M. (2022). Remote sensing techniques for automated marine mammals detection: A review of methods and current challenges. *PeerJ* 10, e13540. doi: 10.7717/peerj.13540

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-Net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention – MICCAI 2015, lecture notes in computer science*. Eds. N. Navab, J. Hornegger, W. M. Wells and A. F. Frangi (Cham: Springer International Publishing), 234–241. doi: 10.1007/978-3-319-24574-4_28

Schlossberg, S., Chase, M. J., and Griffin, C. R. (2016). Testing the accuracy of aerial surveys for Large mammals: An experiment with African savanna elephants (Loxodonta africana). *PloS One* 11, e0164904. doi: 10.1371/journal.pone.0164904

Shah, K., Ballard, G., Schmidt, A., and Schwager, M. (2020). Multidrone aerial surveys of penguin colonies in Antarctica. *Sci. Robotics* 5, eabc3000. doi: 10.1126/scirobotics.abc3000

Tan, M., and Le, Q. (2019). "EfficientNet: Rethinking model scaling for convolutional neural networks, in: Proceedings of the 36th international conference on machine learning," in *Presented at the International Conference on Machine Learning* (Long Beach, CA, USA: PMLR). 6105–6114. doi: 10.48550/arXiv.1905.11946

Tuia, D., Kellenberger, B., Beery, S., Costelloe, B. R., Zuffi, S., Risse, B., et al. (2022). Perspectives in machine learning for wildlife conservation. *Nat. Commun.* 13 (1), 792. doi: 10.1038/s41467-022-27980-y

Wang, J., Lan, C., Liu, C., Ouyang, Y., and Qin, T. (2021). "Generalizing to unseen domains: A survey on domain generalization," in *Presented at the Thirtieth International Joint Conference on Artificial Intelligence*, Montreal, Canada. 4627–4635. doi: 10.24963/ijcai.2021/628

Wanless, S., Murray, S., and Harris, M. P. (2015). Aerial survey of northern gannet (Morus bassanus) colonies off NW Scotland 2013 - NERC open research archive (No. 696). *Scottish Natural Heritage*.

Watt, C. A., Marcoux, M., Hammill, M., Montsion, L., Hornby, C., Charry, B., et al. (2021). *Abundance and total allowable landed catch estimates from the 2017 aerial survey of the Cumberland sound beluga (Delphinapterus leucas) population (No. 2021/50)* (Canadian Science Advisory Secretariat (CSAS).

Weinstein, B. G. (2018). A computer vision for animal ecology. *J. Anim. Ecol.* 87, 533– 545. doi: 10.1111/1365-2656.12780

Wilkinson, T., Agardy, T., Perry, S., Rojas, L., Hyrenbach, D., Morgan, K., et al. (2003). "Marine species of common conservation concern. protecting species at risk across international boundaries," in *Presented at the Fifth International SAMPAA (Science and Management of Protected Areas), University of Victoria*, Victoria, B.C., Canada.

Yoccoz, N. G., Nichols, J. D., and Boulinier, T. (2001). Monitoring of biological diversity in space and time. *Trends Ecol. Evol.* 16, 446–453. doi: 10.1016/S0169-5347(01)02205-4

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 13 March 2023
DOI 10.3389/fmars.2023.1143499

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Peng Ren,
China University of Petroleum (East China), China
Zhenya Song,
Ministry of Natural Resources, China

*CORRESPONDENCE
Wenhui Li
liwenhui@tju.edu.cn
An-An Liu
anan0422@gmail.com

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 13 January 2023
ACCEPTED 20 February 2023
PUBLISHED 13 March 2023

CITATION
Song D, Su X, Li W, Sun Z, Ren T, Liu W and Liu A-A (2023) Spatial-temporal transformer network for multi-year ENSO prediction.
Front. Mar. Sci. 10:1143499.
doi: 10.3389/fmars.2023.1143499

COPYRIGHT
© 2023 Song, Su, Li, Sun, Ren, Liu and Liu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Spatial-temporal transformer network for multi-year ENSO prediction

Dan Song[1], Xinqi Su[1], Wenhui Li[1]*, Zhengya Sun[2], Tongwei Ren[3], Wen Liu[4] and An-An Liu[1]*

[1]School of Electrical and Information Engineering, Tianjin University, Tianjin, China, [2]Institute of Automation, Chinese Academy of Sciences, Beijing, China, [3]State Key Laboratory for Novel Software Technology,Nanjing University, Nanjing, China, [4]School of Navigation, Wuhan University of Technology, Wuhan, China

The El Niño-Southern Oscillation (ENSO) is a quasi-periodic climate type that occurs near the equatorial Pacific Ocean. Extreme periods of this climate type can cause terrible weather and climate anomalies on a global scale. Therefore, it is critical to accurately, quickly, and effectively predict the occurrence of ENSO events. Most existing research methods rely on the powerful data-fitting capability of deep learning which does not fully consider the spatio-temporal evolution of ENSO and its quasi-periodic character, resulting in neural networks with complex structures but a poor prediction. Moreover, due to the large magnitude of ocean climate variability over long intervals, they also ignored nearby prediction results when predicting the Niño 3.4 index for the next month, which led to large errors. To solve these problem, we propose a spatio-temporal transformer network to model the inherent characteristics of the sea surface temperature anomaly map and heat content anomaly map along with the changes in space and time by designing an effective attention mechanism, and innovatively incorporate temporal index into the feature learning procedure to model the influence of seasonal variation on the prediction of the ENSO phenomenon. More importantly, to better conduct long-term prediction, we propose an effective recurrent prediction strategy using previous prediction as prior knowledge to enhance the reliability of long-term prediction. Extensive experimental results show that our model can provide an 18-month valid ENSO prediction, which validates the effectiveness of our method.

KEYWORDS

El Niño southern oscillation, long-term prediction, spatio-temporal modeling, transformer, deep learning

1 Introduction

The EI Nino-Southern Oscillation (ENSO) is one of the recurring interannual variability of ocean-atmosphere interactions phenomenon over the tropical Pacific Ocean and contains three phases (onset, mature and decay) with respect to the changes of sea surface temperature(SST). When the SST are higher than normal in the central and

eastern equatorial Pacific Ocean, it is called El Nino, and when it is lower than normal, it is called La Nina Larkin and Harrison (2002). With wind and SST oscillations, the ENSO has wide influences, for example, the global atmospheric circulation Alexander et al. (2002), crop production Solow et al. (1998), environmental and socioeconomic (McPhaden et al. (2006)), ecology and economy Reyes-Gomez et al. (2013). Therefore, accurate prediction of ENSO occurrence can guide us to take preventive measures and effectively reduce the impact of natural disasters on human society. However, due to the predictability barrier and chaos of climate variability Mu et al. (2019) ENSO prediction remains an extremely challenging task.

In recent years, there are several related indicators to reveal ENSO underlying complex climate change, such as Nino3.4 index and the SST index Yan et al. (2020). All of them utilize the historical SST or Heat Content (HC, Vertical mean ocean temperature above 300 m) to predict whether the ENSO event will happen in the future. Among these indicators, the Nino3.4 index is frequently employed to evaluate phenomenon of ENSO, which calculates mean SST anomaly(SSTA) maps of three consecutive months in an area of 5°N-5°S and 170°W-120°W Ham et al. (2019). The existing ENSO prediction methods can roughly classified into numerical prediction methods (NWP), traditional statistical methods and deep learning methods Ye et al. (2021b). The NWP methods usually adopt the mathematical physics and integrating governing partial differential equations to predict future Nino3.4 index Bauer et al. (2015). Specifically, Zebiak et al. Zebiak and Cane (1987) proposed the first coupled atmosphere-ocean model for forcasting the ENSO phenomenon, and subsequently various models like Intermediate Coupled Model (ICM), Hybrid Coupled Model (HCM) and Coupled General Circulation Model (CGCM), have been proposed to obtain 6-12 months of reliable predictions He et al. (2019).For example, Zhang et al. Zhang and Gao (2016) developed an ICM for enso prediction focusing on thermocline effect on the SST, which reasonably captures the overall warming and cooling trends from 2014-2016. Subsequently, Barnston et al. Barnston et al. (2019) validated the ENSO prediction skill in the North American Multi-Model Ensemble(NMME) and found that NMME can effectively improve the ENSO prediction skill. Johnson et al. Johnson et al. (2019) used the European Centre for Medium-Range Weather Forecasts(ECMWF) to predict ENSO and found that ECMWF has powerful advantages in ENSO prediction, especially in the difficult-to-predict northern spring and summer season. Ren et al. Ren et al. (2019) developed a statistical model to examine the East Pacific (EP) type and Central Pacific (CP) type predictability, and the results showed that ENSO predictability is mainly derived from changes in the upper ocean heat content and surface zonal wind stress in the equatorial Pacific. However, due to weather prediction is highly dependent on initial and boundary conditions, as well as a large variety of physical quantities, which hinder the application of NWP in long-term prediction Ludescher et al. (2021). Furthermore, with the horizontal resolution increasing, the numerical models will lead to an explosion of time costs and computational resources Mu et al. (2019); Ye et al. (2021b). Traditional statistical methods summarized and analyze the shallow patterns in historical data of ENSO, and then, realize the prediction of future ENSO Yan et al. (2020). Concretely, Petrova et al. Petrova et al. (2017) decomposed the time series into dynamic components and captured the dynamic evolution of ENSO to obtain efficient predictions. Subsequently, PETROVA et al. Petrova et al. (2020) added a stochastic periodic component associated with the ENSO time scale, which further improved the prediction. Wang et al. Wang et al. (2020) proposed a nonparametric statistical approach based on simulation prediction to address the limitation of long-term prediction for statistical methods raised by highly non-linear and chaotic dynamics. Rosmiati et al. Rosmiati et al. (2021) proposed the auto regressive ensemble moving average (ARIMA) model to predict the Niño3.4 Index and found that ARIMA was very effective in predicting ENSO events. However, ENSO is non-linear ocean-atmosphere phenomenon over time, traditional statistical methods can not well capture the complex patterns and knowledge to effectively predict the ENSO phenomenon Yan et al. (2020).

As deep learning techniques have developed, researchers have began to design neural networks for predicting weather elements (e.g., rainfall), which can well mine complex and intrinsic correlations, such as artificial neural networks (ANN) Feng et al. (2016), convolutional neural networks (CNN) Ham et al. (2019); Ye et al. (2021b); Patil et al. (2021), long short-term memory networks (LSTM) Broni-Bedaiko et al. (2019), convolutional long short-term memory networks (ConvLSTM) Mu et al. (2019); He et al. (2019); Gupta et al. (2022), CNN-LSTM Zhou and Zhang (2022), graph neural networks (GNN) Cachay et al. (2020), recurrent neural network (RNN) Zhao et al. (2022), transformer Ye et al. (2021a) etc. Feng et al. Feng et al. (2016) propose two methods to predict the existence of ENSO, and the time evolution of ENSO scalar features, which provided a new prediction direction for predicting the occurrence for ENSO events. Broni-Bedaiko et al. Broni-Bedaiko et al. (2019) used the LSTM networks for multi-step advance prediction of ENSO events, which complemented the previous models and predicted the ENSO phenomenon 6, 9, and 12 months in advance. Mu et al. Mu et al. (2019) defined ENSO prediction as a spatio-temporal series prediction issue and used a mixture of ConvLSTM and rolling mechanism to predict the outcome over a longer range of events. The GNN was first used in Cachay et al. (2020) for seasonal prediction, it predict the result in a longer lead time. Zhao et al. Zhao et al. (2022) designed an end-to-end network, named Spatio-Temporal Semantic Network (STSNet), it provided a multiscale receptive domains across spatial and temporal dimensions. The significant breakthrough work is the CNN-based model designed by Ham et al. Ham et al. (2019), which is proficient in predicting ENSO incidents for as long as 1.5 years, significantly higher than most existing methods. Subsequently, Ye et al. Ye et al. (2021b) adapted the different sizes of the convolutional kernel to capture the different scale information and further improved the accuracy than Ham et al. (2019). Patil et al. Patil et al. (2021) trained CNN models using accurate data with the all season correlation skill greater than 0.45 at lead time of 23 months. Another major breakthrough is the combination of the POP analysis procedure with the CNN-LSTM algorithm by Zhou and Zhang (2022), which explores hybrid

modeling by combining physical process analysis methods with neural network and proves its effectiveness. In addition, deep learning in the field of spatio-temporal prediction is now well developed, Li et al. Li et al. (2022) developed an adversarial learning method fully considering the spatial and temporal characteristics of the input data to produce accurate wind field estimates, and Lv et al. Lv et al. (2022) proposed a new generative adversarial network model to simulate the spatial and temporal distribution of pedestrians to generate more reasonable future trajectories, which provides new ideas for ENSO prediction.

Although certain advances have been made in ENSO-related studies, there are still quite limited predictions due to the following reasons: (1) The ENSO phenomenon contains prominent spatio-temporal characteristic, and even if the temperatures of two stations with long time intervals and far apart locations, they may still have complex interactions with different implications for future ENSO prediction. The traditional CNN convolution kernel suffers from the problem of local receptive field, for example, to obtain the SST anomaly relationship between the North Pacific and South Atlantic, it is necessary to stack the deep layers to obtain these two areas, but the amount of information decays as the number of layers increases Ye et al. (2021a). The transformer-based methods explored the attention mechanism to capture the global receptive field. However, these methods mainly model the spatial information, resulting in confusing spatio-temporal features Nie et al. (2022). (2) Due to the variable rate signal and high frequency noise in atmosphere-ocean system, it is a challenge for predicting long-time ENSO in advance. The previous close calendar months have significant effect on the next month prediction, while those with longer intervals have low effect. Existing methods ignore the nearby prediction results when they mine the spatial-temporal patterns in the next time, resulting large errors due to the large magnitude of ocean climate variability over long intervals. (3) The ENSO phenomenon has an obvious statistical characteristic of annual cycle Zhou and Zhang (2022), and how to effectively use this interannual characteristic to capture the correlation between historical and predicted data is the key to improve the prediction of the future trend change in atmosphere-ocean system.

To solve the above limitations, we designed a novel Spatial-temporal Transformer Network for Multi-year ENSO prediction, which is named STTN. First, as the ENSO phenomenon has large-scale and long-term dependencies across both spatial and temporal dimensions, we employed a multi-head spatial-temporal network to adaptively model the variations along with the changes in space and time, which can effectively captures the global and successive characteristics of climate change. Second, we designed an effective recurrent prediction strategy to utilize the previous predictions as prior knowledge for long-term prediction by a single model. To mitigate the negative influence of false predictions, we encoded the contextual information of successive predictions by temporal convolution operation to fully exploit the historical contextual time series. Third, we integrated the month information into the procedures of SSTA and HC anomaly (HCA) maps feature encoding and predictions, which guides the model to better capture the seasonality and periodicity of the ENSO phenomenon.

The main contributions from our work are summarized below:

- We proposed a novel spatial-temporal transformer network to model the variations of SSTA and HCA along with the changes in space and time, which can adaptively captures the inherent characteristics of climatic oscillation.
- We introduced an effective recurrent prediction strategy to treat previous predictions as prior knowledge for long-term predictions and utilize the context of predictions to mitigate the error accumulation during recurrent prediction.
- We integrated the temporal index as position embedding into the feature learning procedure to facilitate mining the influence of seasonal variation on predicting ENSO.
- The extensive experiments indicated that our single model outperforms the state-of-the-art methods with multiple ensemble models, which demonstrates the effectiveness of our method at dynamic prediction.

2 Methodology

2.1 Data processing

The ENSO prediction has been defined as a spatio-temporal prediction issue, where the objective is to use the ENSO historical data $x_{t\text{-}T+1,\ldots,}x_{t\text{-}1,}\ x_t$ to predict the Niño3.4 indexes for the next l months. This process is formulated as:

$$[y_{t+1}, y_{t+2} \ldots y_{t+l}] = \mathrm{F}(x_{t-T+1}, x_{t-1} \ldots x_t) \tag{1}$$

where F denotes the deep learning model, l denotes the lead month, T denotes the length of historical input data. The illustration of our proposed network is illustrated in Figure 1.

The time unit of ENSO historical input data contains T consecutive months, denoted as $x_{ssta} \in \mathbb{R}^{T\times H\times W}$ and $x_{hca} \in \mathbb{R}^{T\times H\times W}$ for SSTA and HCA, respectively. T, H, and W indicate time, height, and width for the input data, respectively. To model the spatial and temporal correlation with a global perspective, we adopt the transformer structure as the backbone of our method. To meet the requirement of transformer structure, we first reshape the SSTA and HCA 2D data into a sequence of flattened 2D patches. Taking x_{ssta} as an example, each grid map is divided into N patches with same size: $x'_{ssta} \in \mathbb{R}^{T\times N\times p_1\times p_2}$, $N{=}H{\times}W/(\mathrm{p}_1{\times}\mathrm{p}_2)$. The p_1 and p_2 is the size of each patch, then each patch is converted into a one-dimensional vector with $p_1{\times}p_2$ dimension. Then, we adopt a linear layer to project these vectors into D dimension. Finally, the features of the SSTA or HCA can be represented as $f_{ssta}{\in}\mathbb{R}^{T\times N\times D}$ and $f_{hca}{\in}\mathbb{R}^{T\times N\times D}$.

2.2 Spatial-temporal position encoding

Due to the complex historical input data with periodic characteristics, we need to assign the position indexes for each patch to let the network know the location and order of each patch, so that the model can explore the correlations among different

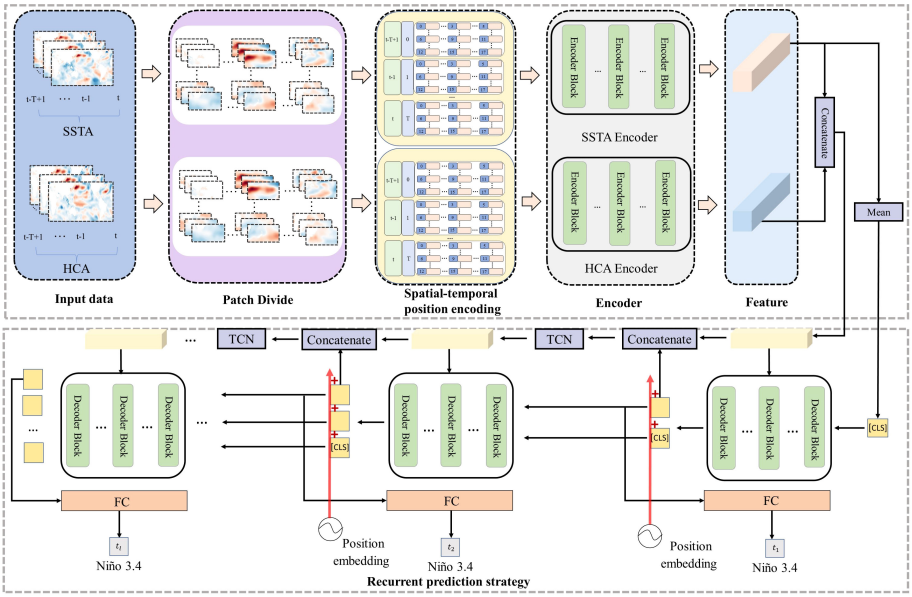


FIGURE 1
The proposed STTN model architecture, which contains Input data, Patch Divide, Spatial-temporal position encoding, Encoder, SSTA and HCA Features, and Recurrent prediction strategy. The SSTA and HCA encoder consist of multiple transformer encoder blocks. The Recurrent prediction strategy predicts the Niño3.4 index according to the time step. Input variables are SSTA (in units of °C) and HCA (in units of °C) from *t*-*T*+1 to *t* (in units of month).The STTN model outputs the Niño3.4 indexes for the next l months.

locations or at different times. To encode the temporal information, we adopt different sine and cosine functions Vaswani et al. (2017), which are periodic and can explore the temporal characteristic of abnormal temperature. Take f_{ssta} as an example:

$$\begin{aligned} PO(i,2j) &= \sin\left(\mathrm{i}/10000^{2j/D}\right) \\ PO(i,2j+1) &= \cos\left(\mathrm{i}/10000^{2j/D}\right) \end{aligned} \tag{2}$$

where i is the time step of the input sequence or the calendar month in the period of C, and j is the index of dimension, $PO \in \mathbb{R}^{T\times D}$. For the location of each patch within space, we learn spatial positional embedding $E \in^{N\times D}$. Finally, the spatio-temporal position is added to the feature f_{ssta} and to obtain the embedding vector $z_{ssta}^{(0)}$.

$$z_{ssta}^{(0)} = Norm(f_{ssta} + E + PO) \tag{3}$$

where *Norm* is the LayerNorm operator, and the embedding vector $z_{hca}^{(0)}$ of HCA can also be obtained by the above process. In addition, the calendar month information and the time step of the input sequence also contributed to the recurrent prediction strategy which will be presented later.

2.3 Spatial-temporal attention module

To better model the spatial and temporal characteristics of ENSO, we adopt a multi-head attention to encode the variability. Without losing generality, we take SSTA data as the input. The encoder structure is shown in Figure 2A, which consists of spatial and temporal attention, multi-layer perceptron, and residual connection to obtain the feature representation. To capture the temporal dynamics, we first use the self-attention mechanism in the time dimension. For example, in the case of temporal attention, exclusively using keys from the same patches but different frames as the query, the query, key, and value vectors in the m-th Encoder block can be computed from the feature vector $z^{(m-1)} \in \mathbb{R}^{N\times T\times D}$ as follows.

$$\begin{aligned} \mathrm{q}_t^{(m,a)} &= W_q^{(m,a)}\mathrm{Norm}(z^{(m-1)}) \in \mathbb{R}^{D_h} \\ \mathrm{k}_t^{(m,a)} &= W_k^{(m,a)}\mathrm{Norm}(z^{(m-1)}) \in \mathbb{R}^{D_h} \\ \mathrm{v}_t^{(m,a)} &= W_v^{(m,a)}\mathrm{Norm}(z^{(m-1)}) \in \mathbb{R}^{D_h} \end{aligned} \tag{4}$$

where $t = 1,\ldots,T$, and Norm is the LayerNorm operation, $a = 1,\ldots,A$ is the index of attention heads, and A is the sum of attention heads, the dimension of the attention head is given as $D_h = D/A$. $W_q^{(m,a)}$, $W_k^{(m,a)}$, $W_v^{(m,a)}$ are the parameters for the projection layers. The weights of temporal patches are obtained by a dot product calculation as follows.

$$a_t^{(m,a)} = \sigma\left(\frac{q_t^{(m,a)^T}}{\sqrt{D_h}}\left\{k_{t'}^{(m,a)}\right\}_{t'=1,\ldots,T}\right) \tag{5}$$

where σ is the softmax activation function and $a_t^{(m,a)} \in \mathbb{R}^{T\times T}$ is the temporal attention layer m in terms of a-th head. The patch representations are calculated by these weights.

$$p_t^{(m,a)} = \sum_{t'=1}^{T} a_{tt'}^{(m,a)} v_{t'}^{(m,a)} \tag{6}$$

Then, these vectors from all the attention heads are concatenated and projected:

$$z_t^{(m)} = W_t \begin{bmatrix} p_t^{(m,1)} \\ \vdots \\ p_t^{(m,A)} \end{bmatrix} \tag{7}$$

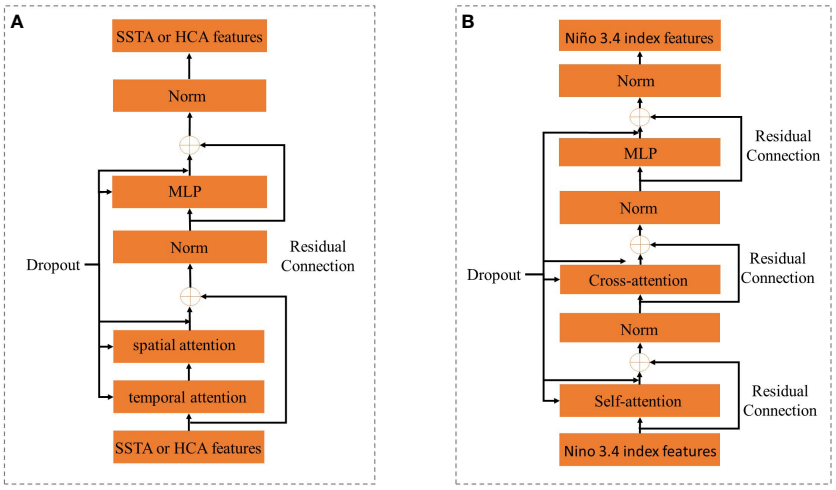


FIGURE 2
The Encoder and Decoder Blocks. The input to the Encoder Block is the SSTA (HCA) feature output by the upper-level block. The input to the Decoder Block is the Niño3.4 index feature output by the upper-level block.

where W_t is the parameter of the linear layer and [] indicates concatenation operation. Further, to capture the spatial dynamics, we use the spatial attention immediately after the temporal attention. The spatial attention calculates the weights in the spatial dimension, exclusively using keys from the same frame as the query. When implementing the spatial attention, we can exchange the spatial and temporal dimensions of $z_t^m \in \mathbb{R}^{N\times T\times D}$, then the query, key, and value vectors can be computed from the feature vector $z_t^m \in \mathbb{R}^{T\times N\times D}$ as follows:

$$\begin{aligned} \mathrm{q}_s^{(m,a)} &= W_Q^{(m,a)}\mathrm{Norm}\,(z_t^m) \in \mathbb{R}^{D_h} \\ \mathrm{k}_s^{(m,a)} &= W_K^{(m,a)}\mathrm{Norm}\,(z_t^m) \in \mathbb{R}^{D_h} \\ \mathrm{v}_s^{(m,a)} &= W_V^{(m,a)}\mathrm{Norm}\,(z_t^m) \in \mathbb{R}^{D_h} \end{aligned} \tag{8}$$

Then, the weight of each space patch also can be computed by the dot product calculation:

$$a_s^{(m,a)} = \sigma\left(\frac{q_s^{(m,a)^T}}{\sqrt{D_h}}\left\{k_{s'}^{(m,a)}\right\}_{s'=1,\ldots,N}\right) \tag{9}$$

where $a_s^{(m,a)} \in \mathbb{R}^{N\times N}$ and σ is the softmax activation function. The encoding of the spatial attention at layer m can be similarly obtained by Eq. 6

$$p_s^{(m,a)} = \sum_{s'=1}^{N} a_{ss'}^{(m,a)} v_{s'}^{(m,a)} \tag{10}$$

Finally, we can also obtain the output $z_s^{(m)}$ of spatial attention as follows:

$$z_s^{(m)} = W_p' \begin{bmatrix} p_s^{(m,1)} \\ \vdots \\ p_s^{(m,A)} \end{bmatrix} \tag{11}$$

where W_p is the parameter of the linear layer and [] indicates the concatenation operation. After using the temporal and spatial attentions, we use the residual connection and multilayer perceptron (MLP) to ensure the stability of the gradient and mine the spatio-temporal features.

$$\begin{aligned} z_s^{(m)} &= \mathrm{Norm}\,(z_s^{(m)} + z^{(m-1)}) \\ z^{(m)} &= \mathrm{Norm}\,(MLP(z_s^{(m)}) + z_s^{(m)}) \end{aligned} \tag{12}$$

After encoding SSTA and HCA data, we get the spatio-temporal features of SSTA and HCA respectively, and in order to perform joint prediction, we concatenate the features of SSTA and HCA to get the feature $Z\in\mathbb{R}^{(2\times T\times N)\times D}$.

2.4 Recurrent prediction strategy

In order to use previous predictions as prior knowledge for long-term prediction, we introduced an effective recurrent prediction strategy (RPS). Specifically, we first utilized the self-attention,cross-attention blocks, MLP layer, and residual connection to construct the decoder of the spatial-temporal characteristics. The structure of the decoder is depicted in Figure 2B. Then, the temporal convolutional block with one-dimensional convolution was adopted to encode the prediction context, which can help reduce the error accumulation in the recurrent prediction process. Finally, the fully connected layer maps the feature vector into the Niño3.4 index to optimize the whole network. It is worth noting that these operations are used in each step of the recurrent prediction. Since the Niño3.4 index is calculated by SSTA, we averaged the features of the SSTA to generate the start character *CLS* Vaswani et al. (2017). When predicting the Niño3.4 index for the l-th lead month, the complete calculation is as follows. First, the output of the decoder before the (l-1) th month is concatenated with *CLS* to generate the input $e^0\in\mathbb{R}^{l\times D}$, which is used for the decoder query. Meanwhile, the time sequence position encoding and calendar month information in the period of C are added to the output of the decoder before the concatenation, and then e^0 is input to the decoder to predict the

Niño3.4 index for the l-th lead month. The process of the decoder is shown below:

$$\begin{aligned} e^{'(m)} &= Norm(SA(e^{m-1}) + e^{m-1}) \\ e^{''(m)} &= Norm(CA(e^{'(m)}, Z) + e^{'(m)}) \\ e^{(m)} &= Norm(MLP(e^{''(m)}) + e^{''(m)}) \end{aligned} \tag{13}$$

where e^{m-1} is output of the m-1 layer decoder block, SA is self-attention. To prevent future information leaks, we use the mask[31] to ensure that the l-th lead month feature can only depend on known outputs smaller than the l feature location in e^{m-1}. CA is cross-attention, and its query/key/value can be computed by $e^{'(m)}$/Z/Z. e^m is the output of decoder for the l-th lead month, then we can get the l-th lead month Niño3.4 index after through a fully connected layer. Moreover, in order to use previous predictions as prior knowledge for long-term projection, we concatenate e_l^m into the input features Z of the CA, l is an index of e^m, and use a one-dimensional convolution with k convolution kernels to mitigate the error accumulation.

3 Experiments

3.1 Dataset and Evaluation metrics

3.1.1 Dataset

Following the existing work Ham et al. (2019), we validate our proposed method on Coupled Model Intercomparison Project Phase 5 (CMIP5, details in Table 1 Ham et al. (2019)) Taylor et al. (2012), Simple Ocean Data Assimilation (SODA) Giese and Ray (2011), and Global Ocean Data Assimilation System (GODAS) Behringer and Xue (2004). These datasets contain the anomaly maps of SST and HC from 180°W-180°E and 55°S-60°N, the spatial resolution of each map is 5° x 5°. The goal of these datasets is to predict the Niño3.4 indexes in the next consecutive months. The details of the data are shown in Table 2. The training dataset includes simulated data from the CMIP5 Taylor et al. (2012) in the period from 1861 to 2004, the validation dataset includes the reanalysis data from the SODA Giese and Ray (2011) in the period from 1871 to 1973, and the test dataset includes the reanalysis data from the GODAS Behringer and Xue (2004) in the period from 1982 to 2017. All methods utilize the same data for training, validation and evaluation. In addition, following the existing work Zhou and Zhang (2022), we also validated our proposed method in Coupled Model Intercomparison Project Phase 6 (CMIP6 Eyring et al. (2016)), SODA, and GODAS. These datasets contain the anomaly maps of SST and HC from 175°W-175°E and 50°S-50°N, the spatial resolution of each map is 5° x 5°, and the details of the data are shown in Table 3. It is worth noting that the dataset in Table 3 was used only for comparison with Zhou and Zhang (2022).

3.1.2 Evaluation metrics

To fairly evaluate the performances of the proposed method and competing methods, we adopted Temporal Anomaly Correlation Coefficient Skill (Corr) and Root Mean Square Error (RMSE) between the predictions and observations with different leading months l, as used in Ham et al. (2019). Corr is a measure of linear correlation between predicted and observed values, and RMSE is the standard deviation of the residuals, which is a standard measure of prediction error between predicted and observed values. In addition to the above metrics for evaluating the performance of ENSO prediction, we also calculated the Mean Absolute Error (MAE) to evaluate the average absolute values. The formulations of Corr, RMSE, and MAE are as follows:

$$\underset{l}{\text{Corr}} = \sum_{m=1}^{12} \frac{\sum_{t=s}^{e}(Y_{t,m} - \overline{Y}_m)(P_{t\,.\,m\,.\,l} - \overline{P}_{m,l})}{\sqrt{\sum_{t=s}^{e}(Y_{t,m} - \overline{Y}_m)^2 \sum_{t=s}^{e}(P_{t\,.\,m\,.\,l} - \overline{P}_{m,l})^2}} \tag{14}$$

$$RMSE_l = \sum_{m=1}^{12} \sqrt{\frac{\sum_{t=s}^{e}(Y_{t,m} - P_{y\cdot m,l})^2}{|e-s|}} \tag{15}$$

$$MAE_l = \sum_{m=1}^{12} \frac{\sum_{t=s}^{e}\left|Y_{t,m} - P_{y\cdot m,l}\right|}{|e-s|} \tag{16}$$

where P is the predicted value, Y is the observed value, $\overline{P}_{m,l}$ is the mean of P, $\overline{Y}_m$ is the mean of Y, m is the calendar month, ranging from 1 to 12. s and e are the start and end years of the data, respectively.

3.1.3 Implementation details

Our approach was implemented on the Pytorch framework, and all experiments were performed on an NVIDIA RTX3090ti with 24 GB of memory. We adopted the strategy of Adaptive moment estimation (Adam) to optimize the network learning. Following the Noam Optimizer Vaswani et al. (2017), we adjusted the learning rate during training. In order to clearly understand the experimental setup, we list the main hyperparameter symbols, descriptions, and the values being set in Table 4, the B_1, B_2, $p_1^{'}$, $p_2^{'}$, p_1, p_2 are set to 160, 80, 8, 12, 10, 14, respectively. The number of layers M of Encoder and Decoder is fixed to 6, the value for attention head A is fixed to 6, and D_1 and D_2 are set to 384 and 768. The convolution kernel of the temporal convolutional network is k=4. The dropout rate d is set to 0.1. The pos in the input sequence of the Encoder is set to 0, 1, 2 and it is set to 3,…26 in the Decoder. The ENSO cycle C is set to 2. For the reproducibility of the experiments, the seeds of CPU and GPU are both 5 when we initialize the parameters, and the GPU seed is 0 when the model is training.

3.2 Comparisons with state-of-the-arts

We compare our method with several representative methods, including numerical prediction and deep learning methods, respectively. The numerical weather prediction contains Scale Inter-action Experiment-Frontier(SINTEX-F) Luo et al. (2008) and the North American MultiModal Ensemble (NMME) Kirtman et al. (2014) with CanCM3, CanCM4, CCSM3, CCSM4, GFDL-aer04, GFDL-FLOR-A06 and GFDL-FLOR-B01. The deep learning method consists of multiple ensemble CNN Ham et al. (2019) and multi scale CNN with parallel deep network(MS-CNN) Ye et al. (2021b), and ensemble model ENSOTR Ye et al. (2021a) with Transformer module. The results are shown in Figure 3. It

TABLE 1 Details of the CMIP5 models used in this study.

CMIP ID	Modeling Group	Integration Period	Number of ensemble Members
BCC-CSM1.1-m	Beijing Climate Center, China Meteorological Administration	JAN1850 - DEC2012	1
CanESM2	Canadian Centre for Climate Modelling and Analysis	JAN1850 - DEC2005	5
CCSM4	National Center for Atmospheric Research	JAN1850 - DEC2005	1
CESM1-CAM5	Community Earth System Model Contributors	JAN1850 - DEC2005	1
CMCC-CM	Centro Euro-Mediterraneo per l Cambiamenti Climatici	JAN1850 - DEC2005	1
CMCC-CMS			1
CNRM-CM5	Centre National de Recherches Meteorologiques/Centre Europeen de Recherche et Formation Avancee en Calcul Scientifique	JAN1850 - DEC2005	5
CSIRO-Mk3-6-0	Commonwealth Scientific and lndustrial Research Organization in collaboration with Queensland Climate Change Centre of Excellence	JAN1850 - DEC2005	5
FIO-ESM	The First Institute of Oceanography, SOA, China	JAN1850 - DEC2005	1
GFDL-ESM2G	NOAA Geophysical Fluid Dynamics Laboratory	JAN1861 - DEC2005	1
GISS-E2-H	NASA Goddard lnstitute for Space Studies	JAN1850 - DEC2005	5
HadGEM2-AO	National lnstitute of Meteorological Research/Korea Meteorological Administration	JAN1860 - DEC2005	1
HadCM3		DEC1859 - DEC2005	1
HadGEM2-CC	Met Office Hadley Centre (additional HadGEM2-ES realizations contributed by Instituto Nacional de Pesquisas Espaciais)	DEC1859 - NOV2005	1
HadGEM2-ES		DEC1859 - NOV2005	4
IPSL-CM5A-MR	Institut Pierre-Simon Laplace	JAN1850 - DEC2005	1
MIROC5	Atmosphere and Ocean Research Institute (The University of Tokyo), National lnstitute for Environmental Studies, and Japan Agency for Marine-Earth Science and Technology	JAN1850 - DEC2012	1
MPI-ESM-LR	Max-Planck-Institut fur Meteorologie (Max Planck lnstitute for Meteorology)	JAN1850 - DEC2005	3
MPI-CGCM3	Meteorological Research lnstitute	JAN1850 - DEC2005	1
NorESM1-M	Norwegian Climate Centre	JAN1850 - DEC2005	1
NorESM1-ME			1

TABLE 2 The training, validation and testing subsets for Niño3.4 index prediction on CMIP5 dataset.

	Data	Models	Type	Period
Training	CMIP5	21	Historical run	1861–2004
Validation	SODA	1	Reanalysis	1871–1973
Testing	GODAS	1	Reanalysis	1982–2017

TABLE 3 The training, validation and testing subsets for Nino3.4 index prediction on CMIP6 dataset.

	Data	Models	Type	Period
Training	CMIP6	23	Historical run	1850–1980
Validation	SODA	1	Reanalysis	1871–1980
Testing	GODAS	1	Reanalysis	1994–2020

display the all-season Corr(ACorr) for three-month-moving-average Niño3.4 index in 1982-2017 and there are several conclusions can been observed:

3.2.1 Numerical prediction vs deep learning

All deep learning methods (e.g. CNN, MS-CNN and Transformer, etc.) outperform the numerical prediction methods (e.g. SINTEX-F and NMME). The main reason is that the numerical prediction methods design mathematical models of the atmosphere and ocean to mine complex variations with complex calculation processes, while the data-driven deep model can automatically explore the variant characteristics of the EI Niño-Southern Oscillation.

3.2.2 CNN-based method vs transformer-based method

The ACorr of single CNN model is above 0.5 for a lead of 13 month prediction Ye et al. (2021b), while the ACorr of multi-scale CNN model is above 0.5 for a lead of 15 month prediction Ye et al. (2021b), which demonstrates that different scales of convolutional kernel sizes utilize multiple receptive fields to better obtain the region correlations. Moreover, the transformer-based methods (e.g. Transformer and ENSOTR) adopt the attention mechanism to conduct spatial interactions and easily obtain global correlations between different regions and outperform the CNN-based methods.

3.2.3 Transformer-based mehtod vs ours

Our proposed method dramatically outperforms the state-of-the-art methods. Specifically, our method without using ensemble multiple models outperforms the ensemble model ENSOTR for all predicted lead months, especially for 3-10 lead months. Comparing to Transformer and ENSOTR, our method not only designs the attention mechanism across both spatial and temporal dimensions but also incorporates the knowledge of prediction and influence of seasonal variation into the learning procedure, which better facilitates the EI Niño prediction.

Figure 3B shows the Corr of the Niño3.4 index variation for each calendar month. The figure shows that our model (right) predicts more months with a Corr of the Niño3.4 index higher than 0.5. In particular, when the target season is May-June-July (MJJ), the SINTEX-F only contains 4 months Ham et al. (2019), the MS-CNN contains 10 months Ye et al. (2021b), and the CNN ensemble model (left) contains 11 months with a correlation coefficient skill higher than 0.5. Our method has 15 months for which the correlation coefficient skill is up to 0.5, which shows that our method can effectively mitigate the drifts of SST and HT due to the springtime equatorial Pacific trade winds. In summary, the ACorr of the Niño3.4 index of our model outperforms all competing methods and can skillfully predict the EI Niño3.4 index over 18 months.

TABLE 4 The hyperparameter symbols, descriptions and values in this study.

Symbol	Description	Value
B_1	batchsize on CMIP5 dataset training	160
B_2	batchsize on CMIP6 dataset training	80
p_1	the height of patch on CMIP5 dataset training	8
P_2	the width of patch on CMIP5 dataset training	12
$p_1^{'}$	the height of patch on CMIP6 dataset training	10
$p_1^{'}$	the width of patch on CMIP6 dataset training	14
M	the number of layers of Encoder and Decoder	6
A	the numbers of attention head	6
D_1	the dimensions of fully connected layer	384
D_2	the dimensions of MLP	768
k	The convolution kernel of temporal convolutiona network	4
d	dropout	0.1

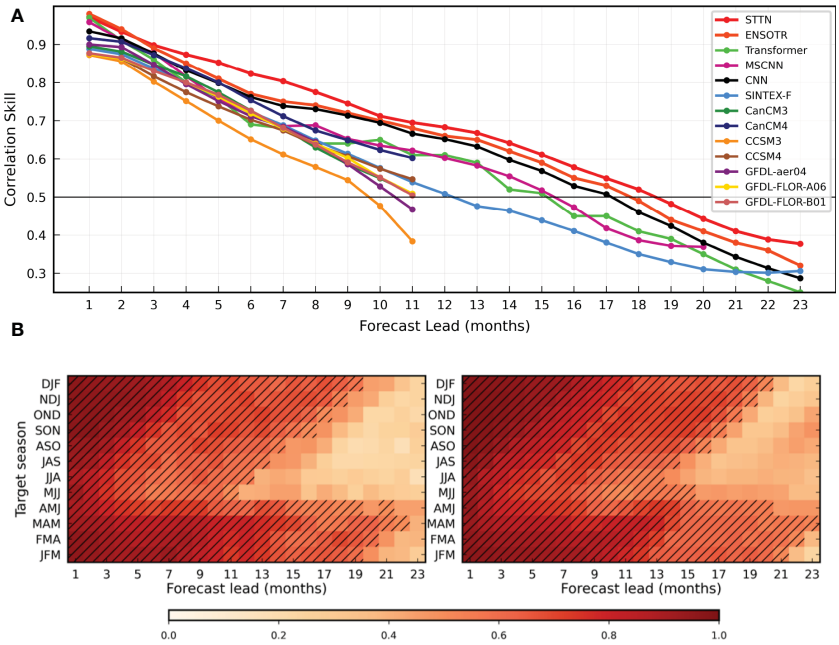


FIGURE 3
ENSO predicts all-season Temporal Anomaly Correlation Coefficient Skill (ACorr) in the STTN model. **(A)** The ACorr of the three-month-moving-averaged Nino3.4 index with Several lead times from ˜ 1982 to 2017 in the STTN model(red), Convolutional Neural Network (CNN) model (black), parallel deep CNNs with heterogeneous Architectures MS-CNN(Light purple), ENSO transformer(ENSOTR)(Orange color), Transformer(Lemon-green), Scale Interaction Experiment-Frontier dynamical prediction system (Sky blue), including additional dynamic prediction systems in the North American Multi-Modal Ensemble (NMME) project (other colors). The ACorr of the Nino3.4 index of every season in the ensemble CNN ˜ model (B.left) and the STTN model (B.right). The light black line indicates that ACorr is equal as 0.5.

3.2.4 Comparison on the CMIP6 dataset

We also compare our method with POP-Net Zhou and Zhang (2022), which is currently the best performing method trained on the CMIP6 dataset. The results are shown in Figure 4. The ACorr of POP-Net model is above 0.5 for a lead of 17 month prediction, while the ACorr of our model is above 0.5 for a lead of 18 month prediction. In general, the ENSO prediction skill of our model is better relative to POP-Net, especially when the lead month is in the range of 12-24. The main reason is that the STTN model can use previous predictions as a priori knowledge for future predictions, which can provide reliable long-term forecasts.

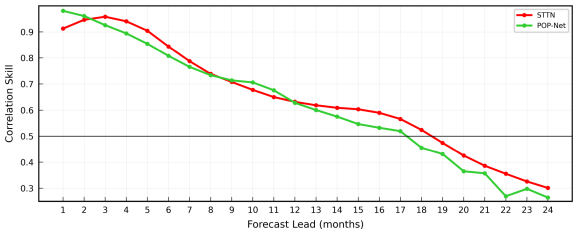


FIGURE 4
ENSO predicts all-season Temporal Anomaly Correlation Coefficient Skill (ACorr) in the STTN model.The ACorr of the three-month-moving-averaged Nino3.4 index index with Several lead times from ˜ 1994 to 2020 in the STTN model(red), POP-Net(Lemon-green).

3.2.5 Comparison of computational resources of different models

Table 5 compares the number of parameters and time cost for the training and testing of the CNN model Ham et al. (2019) and our model. Since the CNN model uses integrated learning, the total number of models is 11040 (23 leadmonths, 12 target months, 4 network settings, and 10 training sessions per model). The number of parameters in the four network settings is 0.12M, 0.18M, 0.21M, and 0.32M, respectively, and the total number of parameters is 2290.8M, which is much larger than our model. In addition, the training and testing time of our model is much lower than that of the CNN model, because STTN only uses the single model instead of the integrated model. The Niño 3.4 index for the next 23 lead months is available in a single run using the STTN model, which indicates that our model can predict the occurrence of El Niño in a more timely and rapid manner.

3.3 Ablation study

In order to verify the importance of our different modules, we performed ablation experiments for each module. To keep the experiment fair, we use the same experimental setting during training as well as testing, including data partition and network hyperparameters. We remove the proposed module from the final network model STTN to demonstrate the effectiveness of using the monthly index of period, the previous prediction as prior knowledge, and TCN, respectively. W/O X indicates the removal of the X module. Figure 5 shows the ACorr,

TABLE 5 Comparison of the computational costs required for different models.

Model	Number of Parameters	Training time cost	Testing time cost
CNN [11]	2290.8M	2700384s	1256.32s
STTN	5.7M	1395.23s	1.10s

RMSR, and MAE when the monthly index of period (w/o p), previous prediction as prior knowledge (w/o F-T N), and TCN (w/o TCN) are removed, respectively. In addition, We also compared the effectiveness of spatio-temporal attention and input data of different lengths.

3.3.1 W/o P

The overall performance of the STTN model decreased after removing the monthly index of period, which indicates that although the neural network can capture the correlation between

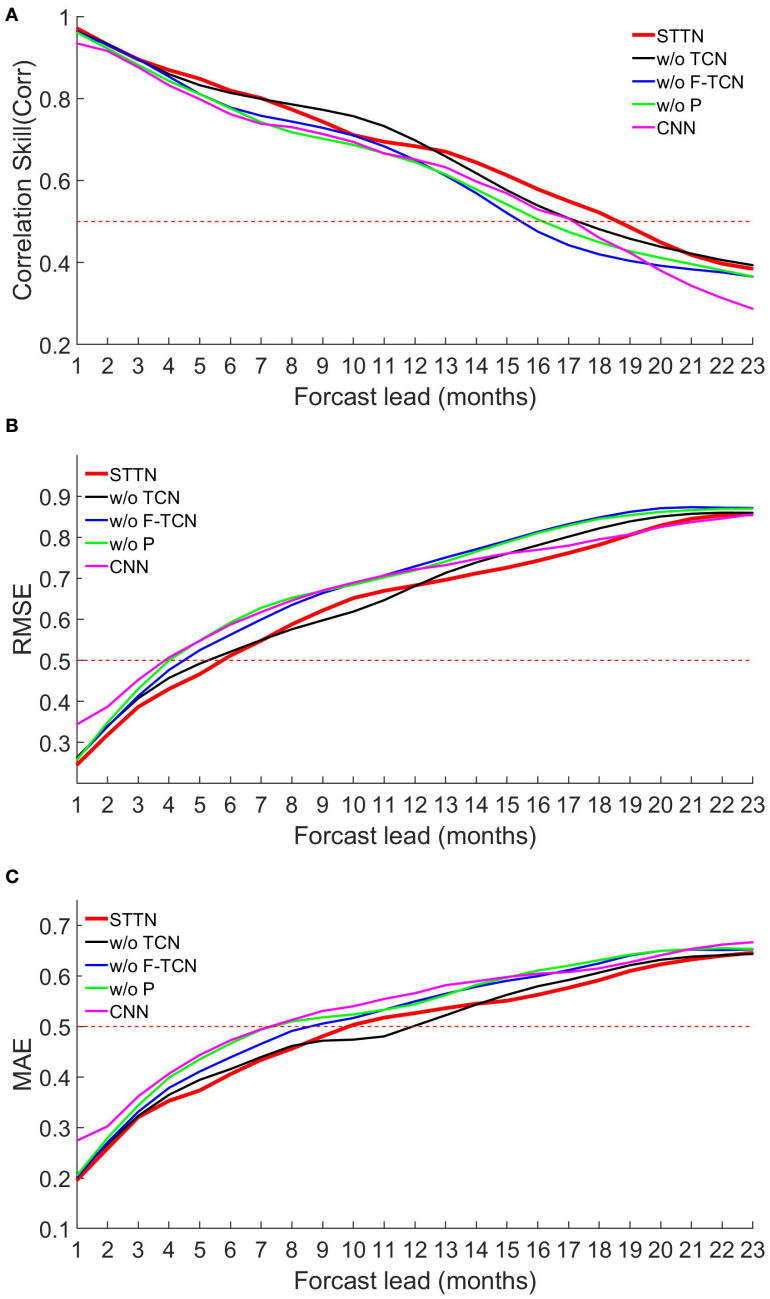


FIGURE 5
Comparison of **(A)** Corr, **(B)** RMSE, and **(C)** MAE between Niño3.4 index predictions and observations obtained with different modules. w/o X indicates removal of the X module. The purple line indicates the result of the CNN model.

data, it cannot capture the period of ENSO. By adding monthly indicators of periodicity, the model can be guided to effectively capture the seasonality and periodicity of the El Niño phenomenon, reducing the complexity of the model in extracting valid features from the input data and helping the model to accurately predict the Niño3.4 index.

3.3.2 W/o F-TCN

After removing the previous predictions as prior knowledge, the ACorr between the predicted and observed Niño3.4 index decreased sharply, especially in the long-term prediction, which indicates that the model does not predict the trend of evolution of El Niño over the next 23 months well when considering the input data alone. As shown in Figures 4B, C, where the MAE and RMSE increase after removing the previous prediction, it indicates that the previous predictions can compensate over long intervals and provide reliable long-term predictions.

3.3.3 W/o TCN

With the removal of the TCN module, we observed a low degradation in the performance of the model, which indicates that the cycle and future features are very important information. Compared to STTN, the model relies more on thepredicted Niño3.4 index series after lead month 12, which suggests that the temporal semantics are significant in the later stage for Niño3.4 index prediction.

3.3.4 Effectiveness of spatio-temporal attention

We compared the performance of the models using spatio-temporal attention and without using spatio-temporal attention. Figure 6A-C plots the ACorr, RMSR, and MAE of the prediction results. We first observed that the model with spatio-temporal attention performs better than the model without spatio-temporal attention. The spatio-temporal attention semantically learns more separable features and effectively reduces the spatio-temporal chaos,

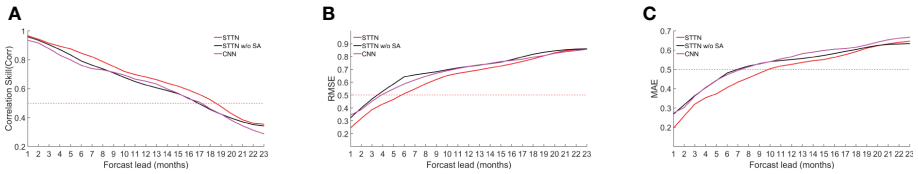


FIGURE 6
Comparison of **(A)** Corr, **(B)** RMSE, and **(C)** MAE between Niño3.4 index predictions and observations obtained using spatio-temporal attention or attention. The purple line indicates the result of the CNN model.

TABLE 6 The RMSE and MAE between Niño3.4 index predictions and observations obtained using different modules and the CNN model.

Model	RMSE	MAE
STTN-w/o C	0.6883	0.5264
STTN-w/o F-TCN	0.6849	0.5178
STTN-w/o TCN	0.65360	0.4949
STTN-w/o SA	0.6941	0.5246
CNN	0.6797	0.5350
STTN	**0.6404**	**0.4930**

The best results are in bold.

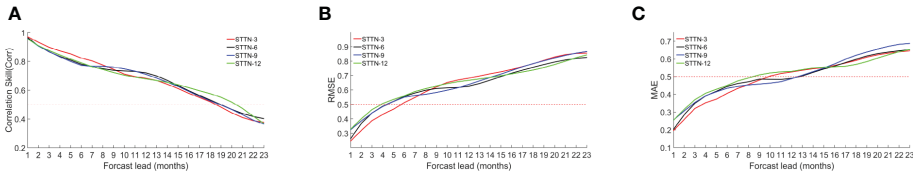


FIGURE 7
Comparison of **(A)** Corr, **(B)** RMSE, and **(C)** MAE between Niño3.4 index predictions and observations for different lengths of input data. The STTN-3, STTN-6, STTN-9, STTN-12 indicate that the length of input data is 3, 6, 9, 12 months, respectively.

allowing the model to better fit the ENSO phenomenon. As can be seen from Table 6, these modules all favor ENSO prediction, and removing any of the modules would harm the performance.

3.3.5 Compare input data of different lengths

We compared the performance of different lengths of input data. Figure 7A-C plots the ACorr, RMSR, and MAE of the predicted results. We observed that the best performance is achieved when the input data length is 3 in lead months 1-8, better performance is achieved when the input data length is 6 or 9 in lead months 8-15, and relatively better performance is achieved when the input data length is 12 in lead months 15-23, so we can conclude that: (1) the early prediction may simply require the SSTA and HCA data that are close in time to the predicted month, and the earlier month may cause noise in the input data; and (2) longer-term predictions require longer inputs, which we speculate may be due to the longer inputs containing more physical laws of ENSO as a result of the westward shift within the ocean.

3.4 Case study

To clearly show the difference between the observed and predicted results from 1982 to 2017, we visualized the Niño3.4 index on the GODAS dataset for 1, 3, 6, 9, 12, and 15 lead months ahead, as shown in Figure 8. From the results, we found that the Niño3.4 indexes at 1-, 3-, 6-, and 9-lead months are accurately predicted and obtain a correlation coefficient skill of 0.97, 0.91, 0.82, and 0.74, respectively. When the lead month increased, the correlation coefficient skill decreased due to the absence of evidence for a long time series and the complex climate variation. Nonetheless, the correlation coefficient is 0.61 and over 0.5 when predicting the index for 15 lead months, which verifies the effectiveness of our method to predict the multi-year ENSO trend.

To explore the seasonal impacts, we show the predicted Niño3.4 index of averaging the December-January-February(DJF) season of 1, 6, 12 and 18 lead months in Figure 9. It can be observed that our method successfully predicts the amplitude of the Niño3.4 index at 6 lead months in advance. Even when we increase the lead time up to 18 months, the trend of our predicted results still fits the curve well when a strong El Niño or La Niña occurs. Moreover, we visualize the predicted results of a typical Super El Niño during (A) 1982-1983, and (C) 2015-2016 as well as a Super La Niña during (B) 1988-1989 in Figure 10. The predictions are the continuous outputs of our method from 1 to 23 lead months, and we can see that our model can successfully predict the evolution of these strong EI Niño phenomena and the results are consistent with the observed results even for longer lead times.

As both the SSTA and HCA influence the ENSO phenomenon, we visualize the contributions of these two factors in Figure 11. This figure shows that when we input three consecutive months during

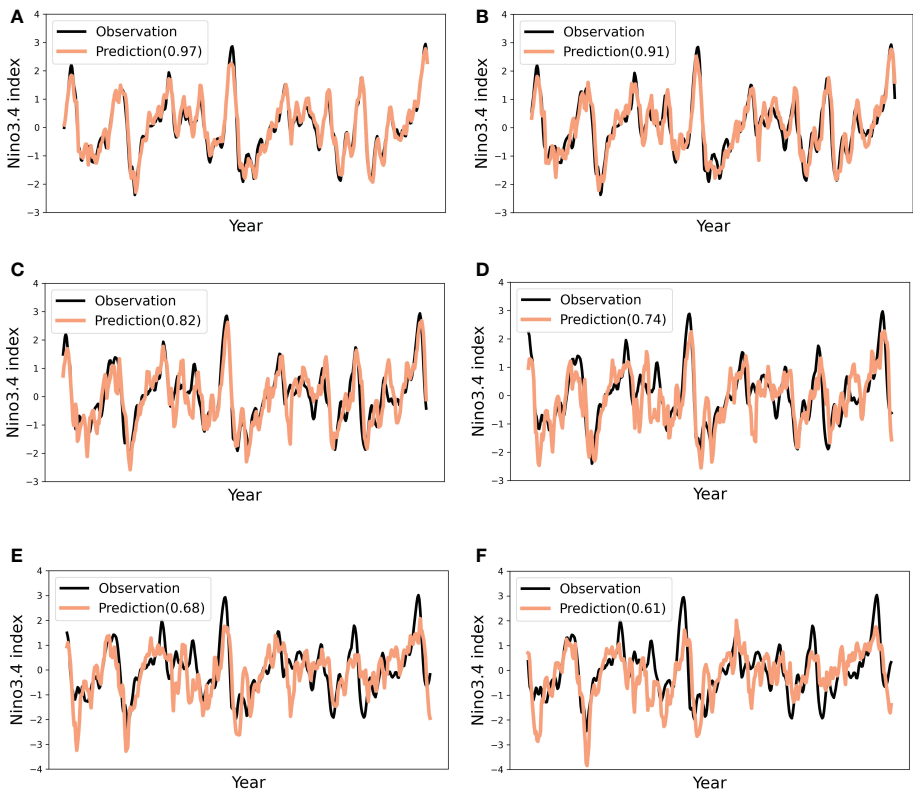


FIGURE 8
The Niño3.4 index of STTN model predictions and observations from 1984 to 2007 with **(A)** 1, **(B)** 3, **(C)** 6, **(D)** 9, **(E)** 12, and **(F)** 15 of lead months,respectively.

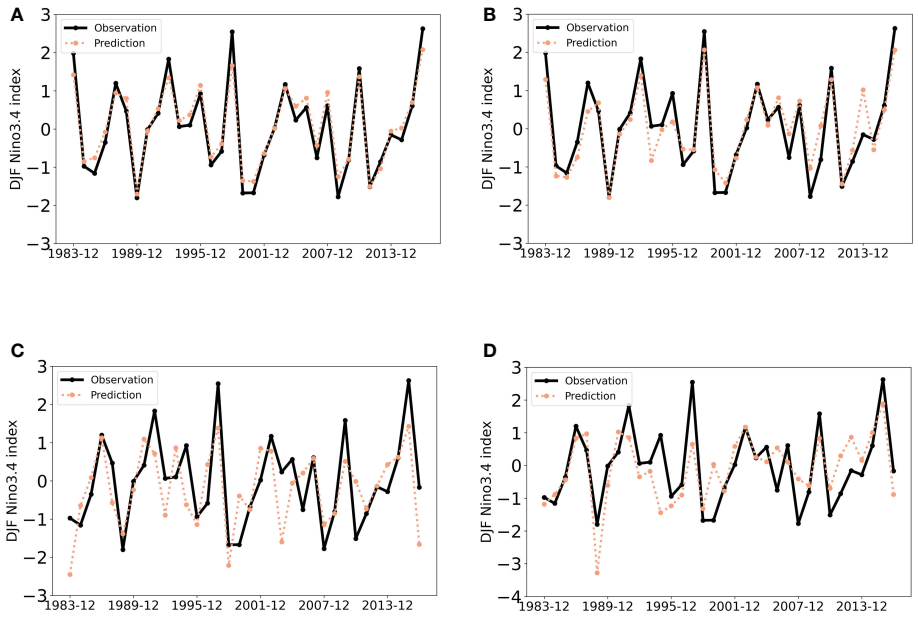


FIGURE 9
Predicted and observed values of Niño3.4 index for the December-January-February season, with **(A)** 1, **(B)** 6, **(C)** 12, **(D)** 18 months for lead months.

the 1997-1998 Super El Niño event, they have different weightings to predict the Niño3.4 index in the next 23 months, which can help us understand how our method can predict El Niño for such a long time. The first row indicates the heat map of SSTA and another row indicates the heat map of HCA. Three columns indicate the time series from December 1997 to February 1998. The darker color represents the more important. From the figure we have the following observations:

- SSTA and HCA show different contributions in both the spatial and temporal dimensions. With the increasing of time, their importance in different spatial locations gradually increase.
- SSTA plays a more important role than HCA at earlier times (first two columns) in predicting the Niño3.4 index. The third column shows that the contributions of SSTA and HCA close to the predicted future are almost equal, which

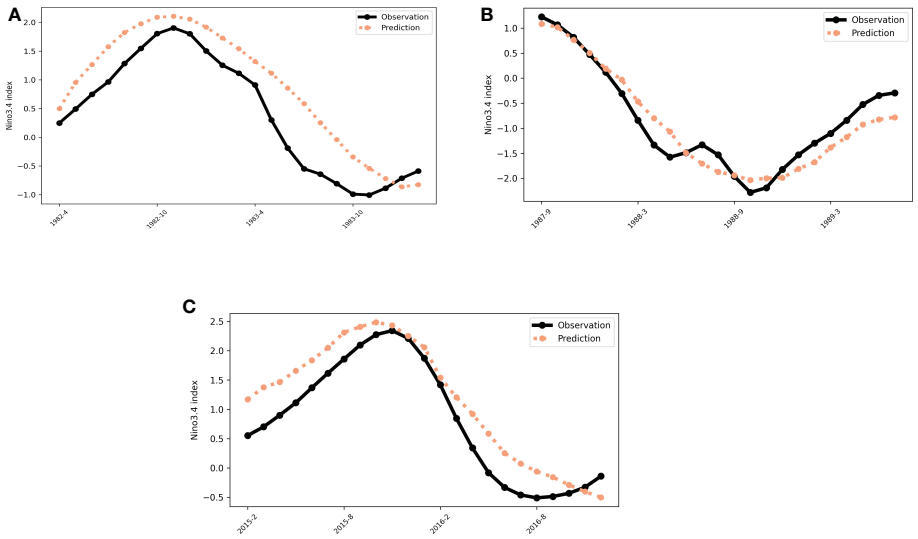


FIGURE 10
The 23 consecutive months output of STTN model in Super El Niño event at **(A)** 1982-1983, **(C)** 2015-2016 and Super La Nina at **(B)** 1988-1989.

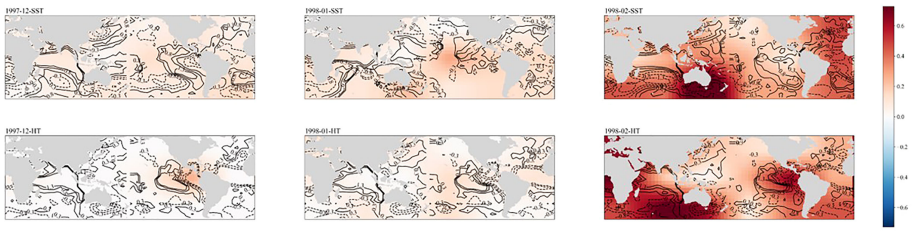


FIGURE 11
The heat map of the contribution of SSTA (in units of °C) and HCA (in units o °C) data to the prediction of the STTN model for the 1997/1998 Super El Niño event for the following 23 consecutive months (The dashed and solid line distributions indicate negative and positive values of SST or HC anomalies). The SSTA and HCA input data are from 1997-December, 1998-January, and 1998-February, respectively.

demonstrates that our method takes full advantage of these two inputs and their complementary relationship.

- The global heat map induces a similar observation to Ham et al. (2019) that the anomalies over the tropical western Pacific, Indian Ocean, and subtropical Atlantic are the main regions to accurately predict the 1997/98 El Niño phenomenon.
- With the change over time (from first column to third column), the contributions of the western part of the map are increasing due to the westward movement that occurs within the ocean.

4 Conclusion

In this paper, we propose a novel spatial-temporal transformer network for multi-year ENSO prediction. Motivated by the attention mechanism, we designed a spatial-temporal attention mechanism to model the contributions of different ocean locations with change over time. For long-term prediction, this article proposes utilizing the accurate previous prediction as prior knowledge and fusing the seasonal variation during the encoding of the temporal information to facilitate the ENSO prediction. Moreover, we use a single model instead of a multi-model architecture to reduce computational resources, which is more convenient for predicting ENSO with different lead times. Extensive experiments using the model on the Coupled Model Intercomparison Project phase 5 (CMIP5) and the Coupled Model Intercomparison Project phase 6 (CMIP6) have shown that our method can provide a more accurate prediction over the existing methods, which verifies the effectiveness of the spatial-temporal attention mechanism, the prior knowledge of previous prediction and the temporal index for modeling the seasonal variation. In the future, we will add more variables and fully explore the relationship among their sea-air interactions to facilitate the reliability of multi-year ENSO prediction.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://doi.org/10.5281/zenodo.3244463.

Author contributions

DS contributed to conceptualization and editing, and supervision. XS data processing, modeling, and writing of the original draft. WLi performed methodology and writing review, A-AL performed validation and review. TR and WLiu, performed validation and investigation. ZS optimized the model framework. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by the National Key Research and Development Program of China (2021YFF0704000) and the National Natural Science Foundation of China (U22A2068).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alexander, M. A., Bladé, I., Newman, M., Lanzante, J. R., Lau, N.-C., and Scott, J. D. (2002). The atmospheric bridge: The influence of enso teleconnections on air–sea interaction over the global oceans. *J. Climate* 15, 2205–2231. doi: 10.1175/1520-0442(2002)015<2205:TABTIO>2.0.CO;2

Barnston, A. G., Tippett, M. K., Ranganathan, M., and L'Heureux, M. L. (2019). Deterministic skill of enso predictions from the north american multimodel ensemble. *Climate Dynamics* 53, 7215–7234. doi: 10.1007/s00382-017-3603-3

Bauer, P., Thorpe, A., and Brunet, G. (2015). The quiet revolution of numerical weather prediction. *Nature* 525, 47–55. doi: 10.1038/nature14956

Behringer, D., and Xue, Y. (2004). Eighth symposium on integrated observing and assimilation systems for atmosphere, oceans, and land surface. *AMS 84th Annual Meeting* Seattle, Washington: Washington State Convention and Trade Center, 11–15.

Broni-Bedaiko, C., Katsriku, F. A., Unemi, T., Atsumi, M., Abdulai, J.-D., Shinomiya, N., et al. (2019). El Niño-southern oscillation forecasting using complex networks analysis of lstm neural networks. *Artif. Life Robot* 24, 445–451. doi: 10.1007/s10015-019-00540-2

Cachay, S. R., Erickson, E., Bucker, A. F. C., Pokropek, E., Potosnak, W., Osei, S., et al. (2020). Graph neural networks for improved el n/no forecasting. doi: 10.48550/arXiv.2012.01598

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., et al. (2016). Overview of the coupled model intercomparison project phase 6 (cmip6) experimental design and organization. *Geosci. Model. Dev.* 9, 1937–1958. doi: 10.5194/gmd-9-1937-2016

Feng, Q. Y., Vasile, R., Segond, M., Gozolchiani, A., Wang, Y., Abel, M., et al. (2016). Climatelearn: A machine-learning approach for climate prediction using network measures. *Geosci. Model. Dev. Discussions*, 1–18. doi: 10.5194/gmd-2015-273

Giese, B. S., and Ray, S. (2011). El Niño variability in simple ocean data assimilation (soda), 1871–2008. *J. Geophysical Res.: Oceans* 116. doi: 10.1029/2010JC006695

Gupta, M., Kodamana, H., and Sandeep, S. (2022). Prediction of enso beyond spring predictability barrier using deep convolutional lstm networks. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2020.3032353

Ham, Y.-G., Kim, J.-H., and Luo, J.-J. (2019). Deep learning for multi-year enso forecasts. *Nature* 573, 568–572. doi: 10.1038/s41586-019-1559-7

He, D., Lin, P., Liu, H., Ding, L., and Jiang, J. (2019). A deep learning enso forecasting model[C]//PRICAI 2019: Trends in Artificial Intelligence. *16th Pacific Rim International Conference on Artificial Intelligence*, August 26–30, 2019, Proceedings, Part II 16. (Cuvu, Yanuca Island, Fiji: Springer International Publishing), 12–23.

Johnson, S. J., Stockdale, T. N., Ferranti, L., Balmaseda, M. A., Molteni, F., Magnusson, L., et al. (2019). Seas5: the new ecmwf seasonal forecast system. *Geosci. Model. Dev.* 12, 1087–1117. doi: 10.5194/gmd-12-1087-2019

Kirtman, B. P., Min, D., Infanti, J. M., Kinter, J. L., Paolino, D. A., Zhang, Q., et al. (2014). The north american multimodel ensemble: phase-1 seasonal-to-interannual prediction; phase-2 toward developing intraseasonal prediction. *Bull. Am. Meteorol. Soc.* 95, 585–601. doi: 10.1175/BAMS-D-12-00050.1

Larkin, N. K., and Harrison, D. (2002). Enso warm (el niño) and cold (la niña) event life cycles: Ocean surface anomaly patterns, their symmetries, asymmetries, and implications. *J. Climate* 15, 1118–1140. doi: 10.1175/1520-0442(2002)015<1118:EWENOA>2.0.CO;2

Li, Y., Huang, W., Lyu, X., Liu, S., Zhao, Z., and Ren, P. (2022). An adversarial learning approach to forecasted wind field correction with an application to oil spill drift prediction. *Int. J. Appl. Earth Observation Geoinform.* 112, 102924. doi: 10.1016/j.jag.2022.102924

Ludescher, J., Martin, M., Boers, N., Bunde, A., Ciemer, C., Fan, J., et al. (2021). Network-based forecasting of climate phenomena. *Proc. Natl. Acad. Sci.* 118, e1922872118. doi: 10.1073/pnas.1922872118

Luo, J.-J., Masson, S., Behera, S. K., and Yamagata, T. (2008). Extended enso predictions using a fully coupled ocean–atmosphere model. *J. Climate* 21, 84–93. doi: 10.1175/2007JCLI1412.1

Lv, Z., Huang, X., and Cao, W. (2022). An improved gan with transformers for pedestrian trajectory prediction models. *Int. J. Intelligent Syst.* 37, 4417–4436. doi: 10.1002/int.22724

McPhaden, M. J., Zebiak, S. E., and Glantz, M. H. (2006). Enso as an integrating concept in earth science. *science* 314, 1740–1745. doi: 10.1126/science.1132588

Mu, B., Peng, C., Yuan, S., and Chen, L. (2019). ENSO Forecasting over multiple time horizons using ConvLSTM network and rolling mechanism. *(Budapest, Hungary: International Joint Conference on Neural Networks (IJCNN)*. 1–8. doi: 10.1109/IJCNN.2019.8851967

Nie, J., Huang, L., Wang, Z., Sun, Z., Zhong, G., Wang, X., et al. (2022). Marine oriented multimodal intelligent computing: challenges, progress and prospects (in chinese). *J. Image Graphics* 27, 2589–2610. doi: 10.11834/jig.211267

Patil, K., Doi, T., Oettli, P., Jayanthi, V. R., and Behera, S. (2021). *AGU Fall Meeting 2021, (New Orleans, LA.)*, A13I–A108.

Petrova, D., Ballester, J., Koopman, S. J., and Rodó, X. (2020). Multiyear statistical prediction of enso enhanced by the tropical pacific observing system. *J. Climate* 33, 163–174. doi: 10.1175/JCLI-D-18-0877.1

Petrova, D., Koopman, S. J., Ballester, J., and Rodó, X. (2017). Improving the long-lead predictability of el niño using a novel forecasting scheme based on a dynamic components model. *Climate Dynamics* 48, 1249–1276. doi: 10.1007/s00382-016-3139-y

Ren, H.-L., Zuo, J., and Deng, Y. (2019). Statistical predictability of niño indices for two types of enso. *Climate Dynamics* 52, 5361–5382. doi: 10.1007/s00382-018-4453-3

Reyes-Gomez, V., Díaz, S., Brito-Castillo, L., and Núñez-López, D. (2013). Enso drought effects and their impact in the ecology and economy of the state of chihuahua, mexico. *WIT Trans. State-of-the-art Sci. Eng.* 64. doi: 10.2495/978-1-84564-756-8/007

Rosmiati, R., Liliasari, S., Tjasyono, B., and Ramalis, T. (2021). Development of arima technique in determining the ocean climate prediction skills for pre-service teacher. *Journal of physics: Conference series*, vol. 1731. (Bengkulu, Indonesia: Mathematics and Science Education International Seminar (MASEIS)), 012072.

Solow, A. R., Adams, R. F., Bryant, K. J., Legler, D. M., O'brien, J. J., McCarl, B. A., et al. (1998). The value of improved enso prediction to us agriculture. *Climatic Change* 39, 47–60. doi: 10.1023/A:1005342500057

Taylor, K. E., Stouffer, R. J., and Meehl, G. A. (2012). An overview of cmip5 and the experiment design. *Bull. Am. Meteorol. Soc.* 93, 485–498. doi: 10.1175/BAMS-D-11-00094.1

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst.* 30. doi: 10.1145/3295222.3295349

Wang, X., Slawinska, J., and Giannakis, D. (2020). Extended-range statistical enso prediction through operator-theoretic techniques for nonlinear dynamics. *Sci. Rep.* 10, 1–15. doi: 10.1038/s41598-020-59128-7

Yan, J., Mu, L., Wang, L., Ranjan, R., and Zomaya, A. Y. (2020). Temporal convolutional networks for the advance prediction of enso. *Sci. Rep.* 10, 1–15. doi: 10.1038/s41598-020-65070-5

Ye, F., Hu, J., Huang, T.-Q., You, L.-J., Weng, B., and Gao, J.-Y. (2021a). Transformer for ei niño-southern oscillation prediction. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2021.3100485

Ye, M., Nie, J., Liu, A., Wang, Z., Huang, L., Tian, H., et al. (2021b). Multi-year enso forecasts using parallel convolutional neural networks with heterogeneous architecture. *Front. Mar. Sci.* 1092. doi: 10.3389/fmars.2021.717184

Zebiak, S. E., and Cane, M. A. (1987). A model el niñ–southern oscillation. *Monthly Weather Rev.* 115, 2262–2278. doi: 10.1175/1520-0493(1987)115<2262:AMENO>2.0.CO;2

Zhang, R.-H., and Gao, C. (2016). The iocas intermediate coupled model (iocas icm) and its real-time predictions of the 2015–2016 el niño event. *Sci. Bull.* 61, 1061–1070. doi: 10.1007/s11434-016-1064-4

Zhao, J., Luo, H., Sang, W., and Sun, K. (2022). Spatiotemporal semantic network for enso forecasting over long time horizon. *Appl. Intell.* 53, 6464–6480. doi: 10.1007/s10489-022-03861-1

Zhou, L., and Zhang, R.-H. (2022). A hybrid neural network model for enso prediction in combination with principal oscillation pattern analyses. *Adv. Atmospheric Sci.* 39, 889–902. doi: 10.1007/s00376-021-1368-4

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 16 March 2023
DOI 10.3389/fmars.2023.1149186

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Zhineng Chen,
Fudan University, China
Suja Cherukullapurath Mana,
Sathyabama Institute of Science and
Technology, India

*CORRESPONDENCE
Xiaodong Wang
wangxiaodong@ouc.edu.cn

SPECIALTY SECTION
This article was submitted to
Ocean Observation,
a section of the journal
Frontiers in Marine Science

RECEIVED 21 January 2023
ACCEPTED 03 March 2023
PUBLISHED 16 March 2023

CITATION
Zhai J, Han L, Xiao Y, Yan M, Wang Y
and Wang X (2023) Few-shot fine-
grained fish species classification *via*
sandwich attention CovaMNet.
Front. Mar. Sci. 10:1149186.
doi: 10.3389/fmars.2023.1149186

COPYRIGHT
© 2023 Zhai, Han, Xiao, Yan, Wang and
Wang. This is an open-access article
distributed under the terms of the Creative
Commons Attribution License (CC BY). The
use, distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Few-shot fine-grained fish species classification *via* sandwich attention CovaMNet

Jiping Zhai[1], Lu Han[1], Ying Xiao[2], Mai Yan[1],
Yueyue Wang[3] and Xiaodong Wang[4]*

[1]College of Electronic Engineering, Ocean University of China, Qingdao, Shandong, China, [2]School of Science, The Hong Kong University of Science and Technology, Hong Kong, Hong Kong SAR, China, [3]Computing Center, Ocean University of China, Qingdao, Shandong, China, [4]College of Computer Science and Technology, Ocean University of China, Qingdao, Shandong, China

The task of accurately classifying marine fish species is of great importance to marine ecosystem investigations, but previously used methods were extremely labor-intensive. Computer vision approaches have the advantages of being long-term, non-destructive, non-contact and low-cost, making them ideal for this task. Due to the unique nature of the marine environment, marine fish data is difficult to collect and often of poor quality, and learning how to identify additional categories from a small sample of images is a very difficult task, meanwhile fish classification is also a fine-grained problem. Most of the existing solutions dealing with few-shot classification mainly focus on the improvement of the metric-based approaches. For few-shot classification tasks, the features extracted by CNN are sufficient for the metric-based model to make a decision, while for few-shot fine-grained classification with small inter-class differences, the CNN features might be insufficient and feature enhancement is essential. This paper proposes a novel attention network named Sandwich Attention Covariance Metric Network (SACovaMNet), which adds a new sandwich-shaped attention module to the CovaMNet based on metric learning, strengthening the CNN's ability to perform feature extraction on few-shot fine-grained fish images in a more detailed and comprehensive manner. This new model can not only capture the classification objects from the global perspective, but also extract the local subtle differences. By solving the problem of feature enhancement, this new model can accurately classify few-shot fine-grained marine fish images. Experiments demonstrate that this method outperforms state-of-the-art solutions on few-shot fine-grained fish species classification.

KEYWORDS

fish species classification, computer vision, few-shot learning, fine-grained image classification, sandwich attention

1 Introduction

Fish species classification is critical to industry and food production as well as conservation and management of marine fisheries. However, most marine fish classification solutions still require manual classification by humans (Alsmadi et al., 2019). As fish classification is a fine-grained problem, the manual classification process is time-consuming and requires a lot of labor and material resources. Due to the dynamic changes of the marine environment, the requirements for shooting equipment are high, which means that the number of underwater images we can obtain is small. Therefore, few-shot fine-grained fish species classification has become a difficult problem to solve. At the same time, due to the absorption and scattering of light in seawater (McGlamery, 1980), as well as other impurities in seawater, most of the collected underwater fish data have poor image quality and complex background problems, which makes the task of few-shot fine-grained fish species classification even more difficult. With the rapid development of computer vision, more and more deep learning methods have appeared in our production, life and work, so the classification of marine species based on deep learning is very necessary (Zhao et al., 2021; Alsmadi and Almarashdeh, 2022; Li et al., 2022).

Few-shot learning is an emerging but important method which attempts to learn new categories from a few labeled examples (Hou et al., 2019). Commonly used methods to solve few-shot image classification mainly include transfer learning (Luo et al., 2017; Peng et al., 2019), meta-learning (Finn et al., 2017; Ren et al., 2018; Lee et al., 2019; He et al., 2023) and metric learning (Vinyals et al., 2016; Snell et al., 2017; Sung et al., 2018; Li et al., 2023). The first two categories focus on finding a suitable initialization parameter model for few-shot learning networks, then using prior knowledge extracted from other tasks to prevent overfitting and improve generalization capabilities. And the last category pays attention to finding a superior similarity metric function to replace the fully connected classification layer with a large amount of parameters, where most existing methods use Euclidean distance and cosine similarity as metric function to classify images. Methods based on metric learning have achieved state-of-the-art performance in the few-shot classification field due to the strong ability of discrimination. Most of the current few-shot image classification methods focus on common classification tasks, that is, the features between categories have obvious differences. However, for fish images, the difference between sample image categories is small, which obviously makes this a fine-grained image classification problem (Zhao et al., 2021), and unfortunately the above classification methods do not take into account the difficulties raised by fine-grained classification.

For few-shot fine-grained image classification, most of the currently available methods take one of two approaches, they either attempt to make the network with a more advanced feature vector measurement module (Vinyals et al., 2016; Sung et al., 2018; Li et al., 2019) or they rely on feature reconstruction (Zhang et al., 2020; Wertheimer et al., 2021). However, they ignore the issue where fine-grained images have much higher requirements for the capabilities of feature extraction modules than general classification methods. Since the images have similar global features in different categories of fine-grained images, and only have significant differences in some subtle features, the extracted feature vectors also have a certain degree of similarity (Wei et al., 2021), which puts too much pressure on the feature measurement module. Due to the small number of samples, few-shot learning is prone to overfitting (Chen et al., 2019), and using a large feature extraction module is not a perfect solution, but through extensive research, the Attention Mechanism (AM) has been used in underwater image enhancement and underwater image dehazing (Shi et al., 2022; Liu P. et al., 2022), it was concluded that an AM may be a better solution for few-shot fish image classification.

Considering the above problems, this paper proposes a novel AM network, named Sandwich Attention CovaMNet (SACovaMNet for short), which can effectively solve the classification problem of few-shot fine-grained fish images, and enable the CNN to more carefully and comprehensively classify marine fish. This new SACovaMNet enables the CNN to extract features from fine-grained images of marine fish in a more detailed and comprehensive manner, capturing recognition objects globally as well as extracting nuances between classes of fish samples locally, thus improving classification accuracy. The main contributions of this work are summarized as follows: 1) To solve the few-shot fine-grained fish species classification problem caused by the small number of fish images and minor differences between classes, we carefully designed a Sandwich Attention module that combines local attention and global attention on the basis of the few-shot model CovaMNet to build our SACovaMNet, which enables CovaMNet to more comprehensively extract features from fine-grained images of marine fish and expand the distance between prototype feature vectors of different categories; 2) Aiming at the problem of missing feature information in the fine-grained image of the CBAM, we improved the CBAM module so that it can weigh the feature map more completely; 3) Exhaustive experiments were conducted based on three fine-grained datasets of marine fish organisms, and experimental results demonstrate that the proposed method outperforms the state-of-the-art solutions.

The rest of this paper is as follows. Section 2 is a review of the related works for few-shot fine-grained image classification. Section 3 introduces the proposed method SACovaMNet. And Section 4 shows the experimental results. Finally, a conclusion is made in Section 5.

2 Related work

Deep learning performs very well when the amount of training data is large, but conversely training the network to perform better becomes problematic when the amount of training data is small. In recent years, few-shot learning (Chen et al., 2019) has been proposed to solve this problem. It was found that few-shot learning is better for the problem of classifying marine fish with sparse samples, and a brief review of the relevant aspects of the problem-solving approach will be given.

2.1 Fish species classification

The fish species classification task is different from general classification tasks, it is a typical fine-grained classification task (Zhao et al., 2021). In recent years, many methods for fish species classification have been proposed, and fish classification models based on biological characteristics (Kartika and Herumurti, 2016; Tharwat et al., 2018) and deep learning models (Chen et al., 2017; Zhao et al., 2021) are more popular. Kartika and Herumurti (2016) proposed a K-means segmentation background and HSV color space feature extraction method, which effectively extracted the color features of koi carp, and finally adopted NBM and SVM methods for identification and classification. Tharwat et al. (2018) took a different approach, using the fusion of Weber Local Descriptor (WLD) features and color features, and also used the LDA algorithm to reduce the dimension of the feature vector and increase the discrimination between different categories (fish species), and finally used the AdaBoost classifier for classification. Unfortunately, methods based on biometric feature extraction cannot handle complex backgrounds or a large number of images, however, deep learning can better solve this problem and achieve more accurate classification results. Rathi et al. (2017) performed classification by pre-processing images using Gaussian blur, morphological operations, Otsu's thresholding, and pyramid mean translation, and further fed the enhanced images to a convolutional neural network for classification. Prasetyo et al. (2022) proposed Multi-Level Residual (MLR) as a new residual network strategy by combining the low-level features of the initial block with the high-level features of the last block using Depthwise Separable Convolution (DSC). They used VGGNet as the backbone of the new CNN architecture by removing the fifth block and replacing it with components such as MLR, Asymmetric Convolution (AC), Batch Normalization (BN), and residual features.

Unfortunately, in reality, due to the complexity of the underwater environment (Shevchenko et al., 2018), it is impossible to obtain enough samples for traditional deep learning training. Guo et al. (2020) believed that the classic CNN model required a large amount of high-quality data to obtain excellent results. For few-shot fish images, it is difficult to obtain data diversity through image augmentation, so a generative network is used to generate realistic fake images with a small amount of training data, and the classification accuracy can be improved by making the datasets diverse and rich. However, the training method based on the generative network is complicated, so the proposed method considers building a few-shot learning method to solve this problem.

2.2 Few-shot learning

2.2.1 Meta-learning

Meta-learning (Hochreiter et al., 2001) is, as the name suggests, learning to learn; the algorithm sets up a meta-learner component and a task-specific learner component, with the training unit being the task, allowing information to cross between tasks. Meta-learning is a popular approach to tackle few-shot problems. MAML (Finn et al., 2017) proposed an algorithm for meta-learning that is model-agnostic, and trained a model's parameters such that a small number of gradient updates will lead to rapid learning on a new task. Reptile (Nichol et al., 2018) removed the re-initialization of each task in order to simplify the update process for MAML, making it a more natural choice in some settings. LEO (Rusu et al., 2019) learnt a low-dimensional latent embedding of model parameters and performed optimization-based meta-learning in this space. While meta-learning has had some success with few-shot problems, it is difficult to train due to its use of complex memory addressing structures (Li et al., 2019), therefore the proposed approach utilizes only a single CNN framework baseline which can be end-to-end trained from scratch.

2.2.2 Transfer learning

Transfer learning (Zhuang et al., 2021) is to transfer the learned model parameters from one model to a new model or task in order to achieve better training results. For datasets with fewer samples, first the model is trained on a dataset with a large number of similar data domains, and then fine-tuned, usually with good results. Compared with the complex training mode of meta-learning, transfer learning can perform simple end-to-end training. Luo et al. (2017) proposed a framework to learn representations that are transferable across different domains and tasks in a label-efficient manner. This method combats domain shift with a domain-adversarial loss and uses a metric learning-based method to generalize embeddings to new tasks. Peng et al. (2019) used the graph convolutional neural network to construct a mapping network between semantic knowledge and visual features, combined image features and semantic features through the fusion of classifier weights, and supplemented semantic features as *a priori* knowledge to a few-shot classifier.

2.2.3 Metric learning

Metric-based learning methods learn a set of item functions (embedding functions) and metrics to measure the similarity between query and sample images and classify them in a feed-forward manner. The main difference between metric-based learning methods is how they learn the metrics, hence metric learning is often referred to as similarity learning (Li et al., 2020). Matching Networks (Vinyals et al., 2016) constructed an end-to-end network architecture that uses cosine similarity to calculate distances. After training, the matching network was able to generate reasonable test labels for unobserved categories without any fine-tuning of the network. In contrast, Prototypical Networks (Snell et al., 2017) mapped the sample data in each category into a space and extracted their means to represent them as protoforms of that class, using Euclidean distance as the distance metric, they are trained so that protoforms of the same class are represented as the closest distance and that inter-class protoforms are represented as the farther distance.

2.3 Fine-grained image classification

2.3.1 Fine-grained image classification

Fine-grained image classification aims to distinguish subcategories, such as birds or dog breeds. Fish image classification also belongs to fine-grained image classification. Compared with general classification tasks, fine-grained image classification is challenging due to high intra-class and low inter-class variance (Zhao et al., 2017). Zhang et al. (2014) proposed a model utilizing deep convolutional features computed on bottom-up region proposals, which learns whole-object and part detectors, enforces learned geometric constraints between them, and predicts a fine-grained category from a pose-normalized representation. Li et al. (2021) proposed a so-called Bi-Similarity Network (BSNet) that consists of a single embedding module and a bi-similarity module of two similarity measures. After the support images and the query images pass through the convolution-based embedding module, the bi-similarity module learns feature maps according to two similarity measures of diverse characteristics.

2.3.2 Few-shot fine-grained image classification

With the development of deep learning, fine-grained image classification has achieved remarkable achievements, but largely relies on a large number of labeled samples. However, in practical applications in some fields, it is difficult to obtain such a large amount of labeled fine-grained data. Therefore, few-shot fine-grained images classification is getting more and more attention (Liu Y. et al., 2022). CovaMNet (Li et al., 2019) proposed a deep covariance metric to measure the consistency of distributions between query samples and new concepts, and used the second-order statistics of concept representation and verified that it is more suitable to represent a concept beyond the first-order statistics, it can naturally capture the underlying distribution information of each concept (or category). Wertheimer et al. (2021) introduced a novel mechanism by regressing directly from support features to query features in closed form, without introducing any new modules or large-scale learnable parameters. Lee et al. (2022) proposed Task Discrepancy Maximization (TDM), which is a feature alignment method, to define the class-wise channel importance, and to localize the class-wise discriminative regions by highlighting channels encoding distinct information of the class. The AM can be used to make the feature vector reweight once before entering the measurement module to ensure that the feature vector pays more attention to the differences between categories, so as to solve the problem of small differences between few-shot fine-grained image samples.

2.3.3 Attention mechanism

Transformer (Vaswani et al., 2017) first achieved excellent results in natural language processing (NLP), and then researchers applied it to the field of vision (Vision Transformer, ViT) (Dosovitskiy et al., 2021; Guo et al., 2022). Dosovitskiy et al. (2021) is believed that the biggest reason for the promising results of Vision Transformer is that it uses a Multi-Headed Self-Attentive (MHSA) module and thus introduces a global attention mechanism, which has powerful representation capabilities. However, due to the image processing method of Vision Transformer, the training time and inference speed will increase quadratically when processing large scale images. To solve this problem, Srinivas et al. (2021) proposed a botnet combining CNN and transformer, in which the 3×3 convolutional layers in the bottleneck are replaced with MHSA, making the botnet achieve state-of-the-art in classification, target detection and segmentation, whilst the training time and inference speed were significantly reduced relative to (Dosovitskiy et al., 2021).

2.4 Comparison to our approach

Compared with other meta-learning based few-shot classification methods, our method SACovaMNet adopts the metric learning architecture and is based on a simple CNN network construction, which can be trained easily in an end-to-end manner from scratch. We use a second-order measurement algorithm that can compare the similarity in more detail, which improves the feature measurement capability of fine-grained images compared to other first-order metric methods. Additionally, our self-designed Sandwich Attention module strengthens the feature extraction ability of our method for fine-grained images, making our method more suitable for the few-shot fine-grained fish species classification.

3 Methodology

The proposed method utilizes episodic training as the training method, as many researchers have demonstrated it to be simple and effective for few-shot problems (Li et al., 2019). The model structure is shown in Figure 1. After the support images and the query images pass through the weight-sharing feature extraction module at the same time to obtain the feature map, the feature map then passes through the Sandwich Attention module to finally obtain the $H \times W \times C$ feature map. The measurement module uses the second-order covariance metric to measure the correlation between query features and support features.

3.1 Baseline

Various metric-based networks have achieved excellent performance in recent few-shot learning studies (Li et al., 2020). Most of the current metric learning algorithms are first-order metric methods such as Euclidean distance or cosine similarity distance. Generally speaking, before the feature map enters these measurement modules, the dimensions of the feature map need to be reduced. Obviously, there will be a large information loss due to this process, especially the spatial information of the feature map. For fish samples especially captured in situ, since the difference between categories is very small, it is very easy to lose key information in pooling and dimensional reduction, so the above approach is unacceptable in fine-grained fish image classification.

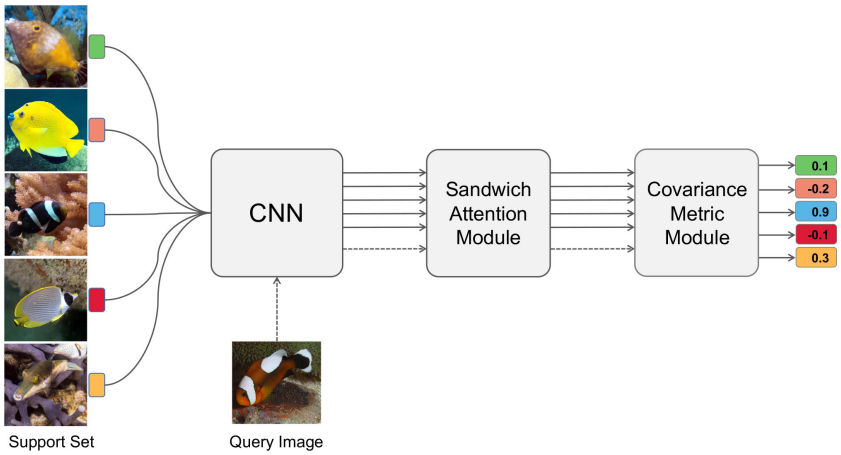


FIGURE 1
Architecture of the proposed SACovaMNet model. The support images and the query image are simultaneously passed through a weight-sharing CNN network to obtain the feature map, and the output feature map is then fed into our Sandwich Attention module to produce the feature map ($H \times W \times C$), which is finally passed through the second-order covariance metric module for similarity calculation.

Recently, (Li et al. 2019) proposed a method based on the second-order local covariance metric. The covariance matrix is the original second-order statistic of the sample set. Since the number of images in each category is very small under the few-shot settings, it is impossible to accurately learn the covariance matrix to describe the data distribution. So the baseline introduces local covariance, expressed as follows:

$$\Sigma_c^{local} = \frac{1}{MK-1}\sum_{i=1}^{K}(\boldsymbol{X}_i - \boldsymbol{\tau})(\boldsymbol{X}_i - \boldsymbol{\tau})^{\top}, \quad (1)$$

where Σ_c^{local} represents the local covariance representation of the c-th class, K is the total number of samples of the c-th class, usually is set as 1 or 5, and X_i is the input sample image, M represents the M local depths of the sample, and τ is an average vector matrix.

The covariance measure is to measure the relationship between a sample and a category, and the measure function named Covariance Metric is as follows:

$$f(\boldsymbol{x}, \Sigma) = \boldsymbol{x}^{\top}\Sigma x\,. \quad (2)$$

The above mentioned Covariance Metric can directly describe the underlying distribution of a concept, and it can fully take into account the local similarity information of the feature map. Since the fish images are fine-grained dataset, and one of the key issues for the classification is to distinguish the local subtle differences between fish categories so as to achieve the more accurate classification. The proposed method has opted to use CovaMNet (Li et al., 2019) which has achieved promisingresults in a series of experimental settings meeting the requirements.

The whole network framework is simple and compact due to it being based on a single end-to-end CNN, a local covariance representation to represent the underlying distribution of each category, and a new covariance metric that is embedded into the network to measure the relationship between query images and categories. The 5-way 1-shot and 5-way 5-shot episodic training mechanism are considered to measure the few-shot classification method under different few-shot situations.

3.2 Sandwich attention

Although the baseline solves some problems in fish classification to a certain extent, the measurement method can only solve the issues in the process of comparing the similarity of feature maps. However, by analyzing the fish image datasets, it was found that most of the images collected in real time cannot correctly reflect the feature information of fish samples due to a variety of problems. In the face of complex fish images, it is expected that feature maps will better reflect the differences between different categories, thereby improving the accuracy of classification, so it was decided to leverage the attention, with a novel attention module designed as shown in Figure 2.

Firstly, in most fish images, the object to be classified is usually only part of the whole image, and there is a lot of interference from the background and other creatures on the seabed, which is also reflected in the feature map extracted by the backbone, making the

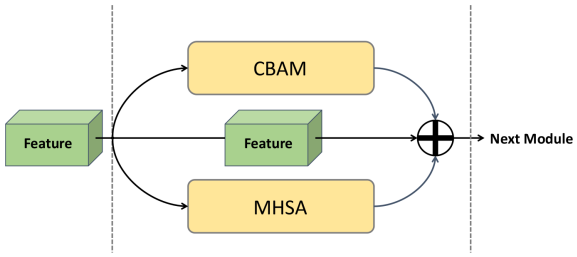


FIGURE 2
Architecture of the proposed attention mechanism, which has been named Sandwich Attention due to it being shaped like a sandwich.

feature map full of useless spatial information. If a manual process was used to increase the proportion of objects identified by manual culling, this would increase the human and financial investment. Therefore it is believed that spatial attention is the most "cost effective" approach to this problem. To this end, a Convolutional Block Attention Module (CBAM) (Woo et al., 2018) module was added to the network, so that the network can correctly locate the position and key feature information of the recognized categories. There are two main tandem sub-modules in CBAM, the channel attention module and the spatial attention module, which perform channel and spatial attention respectively.

In the channel attention module in Figure 3A, the input feature map F ($H\times W\times C$) is subjected to global max pooling and global average pooling to obtain two $1\times1\times C$ feature maps, which are then fed into a two-layer neural network (MLP). Then, the features output by MLP are summed based on element-wise, and activated by sigmoid to generate the final channel attention feature. In the spatial attention module in Figure 3A, the output channel attention and the input feature map F are multiplied element-wise to generate the input of spatial attention module. Next, channel-based global maximum pooling and global average pooling are performed, and then the two feature maps are channel-based splicing operations, one $H\times W\times1$ feature map is obtained through a convolution operation. Finally, the spatial attention feature is generated through the sigmoid function.

At the same time, as fish images are inherently fine-grained, and the difficulty with fine-grained image classification is that the differences between recognized objects are very small and only vary in subtle ways, so the difficulty lies in making the network more accurate in classifying fine-grained images in a few-shot setting. With the rise of ViT in recent years, it is believed that the biggest reason for the promising results achieved by Vision Transformer is because of its powerful representation capabilities using a Multi-Headed Self-Attention module (MHSA) and introducing a global attention mechanism. In Srinivas et al. (2021), the proposed MHSA also introduces Relative Position Encodings, as shown in Figure 3B, thus taking into account the relative distances between features at different locations and being able to effectively relate cross-object

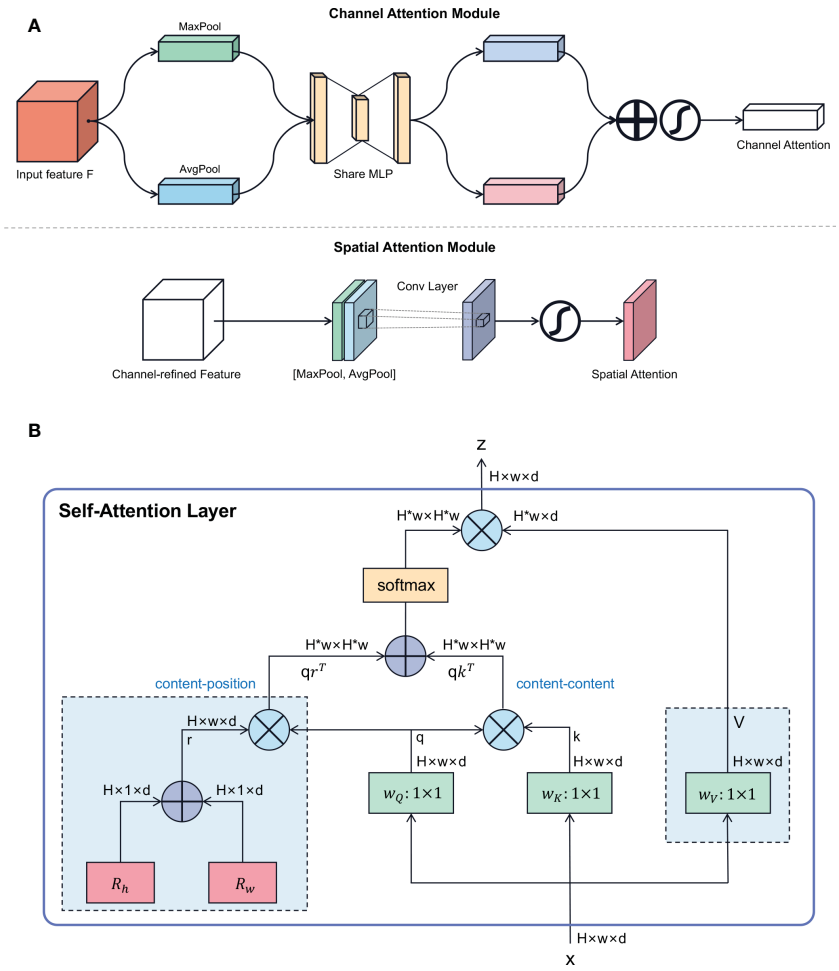


FIGURE 3
The details of AM modules employed in our SACovaMNet model. **(A)** Schematic diagram of each attention sub-module of CBAM (Woo et al., 2018). **(B)** Network structure of multi-head self-attention (MHSA) (Srinivas et al., 2021).

information to location awareness, so this attention mechanism is used in the proposed model.

Based on the above thinking, both MHSA and CBAM were fused into the proposed network. To demonstrate that this approach works and the use of the attentional connectivity, ablation experiments were also conducted in Section 4.4. The final network is based on a simple end-to-end framework using a single CNN with a compact training simple network structure, and the experimental results are presented in Section 4.3.

3.3 Improved CBAM

Although the new model can achieve promising classification results on few-shot fish datasets, fish classification is more difficult due to the difference between fish datasets and general datasets, so it is believed that while CBAM can be applied to fish classification it is still not a perfect solution. More specifically, it is thought that the application of CBAM in fish species classification still has the following problems: 1) The channel attention of CBAM uses global pooling to process the feature map, which obviously does not take into account the importance of different spatial regions of the feature map, resulting in a deviation in the weight calculation of the channel, which is very important for classification, especially that, the difficult fish classification task will obviously have a greater impact; 2) The CBAM uses the feature map of channel attention after global average pooling and maximum pooling to calculate the channel weight through weight-sharing MLP, obviously, there are some differences in the feature map information saved by these two different pooling methods, and using the same MLP cannot fully mine all the information it contains.

Based on the above considerations, we improved the channel attention module of the CBAM module, as shown in Figure 4, both adaptive average pooling and maximum pooling were performed on the feature map (64×21×21) output by the CNN, and it was divided into 7×7 spatial areas, then the MLP module was removed from the CBAM, and two small CNN networks were employed to perform weight calculations respectively, in which the convolution kernel of the first layer of CNN has a large receptive field convolution kernel of 7×7, the second layer is a CNN for dimensionality reduction, the third layer is a Rectified Linear Unit (ReLU) activation function, and the fourth layer is a CNN for dimensionality increase, so we call it DualPath Channel Attention CBAM (DPCACBAM). The importance of different regions of the feature map is calculated not only to ensure that the contribution of different spatial regions of the feature map can be comprehensively considered in the channel attention, but also to fully mine the hidden information in the feature map.

4 Experiments

In this section, extensive experiments were conducted on three fish datasets under corresponding few-shot settings to evaluate the proposed SACovaMNet.

4.1 Datasets

4.1.1 WildFish

This dataset was first proposed in Zhuang et al. (2018), which is a large-scale benchmark dataset for wild fish identification. And it is the largest wild fish recognition image dataset, which contains 1000 fish categories and 54,459 unconstrained images, according to our statistics, the number of images per category varies between 30 and 167. In this work, we randomly split the dataset by categories, where 550, 150, and 300 categories are used for training, validation, and testing, respectively.

4.1.2 Fishclassifierfinal

This dataset is a dataset on the Kaggle website[1], which contains 30 kinds of fish. The dataset has been divided into a train set and a test set. We merge the images of the same fish, and the number of fish images in each category is about 300. We randomly split the dataset by category, where 17, 6, and 7 categories are used for training, validation, and testing, respectively.

4.1.3 QUT fish dataset

This dataset is a dataset also published on the Kaggle website (Anantharajah, 2014), which contains about 4,000 images of 468 fish species. After we classify the given raw images, according to our statistics, the number of each category is between 3 and 26. In this paper, we randomly split this dataset by the number of categories, where 280, 80, and 123 categories are used for training, validation, and testing, respectively.

4.2 Experimental settings

The 5-way 1-shot and 5-way 5-shot classification experiments were conducted on WildFish and fishclassifierfinal datasets. During the training process, episodic training mechanism was used to learn the model parameters, and a total of 250,000 episodes were trained. Each episode contained a query set and a support set. For the 5-way 1-shot classification task, 5 different categories of images were required. Each category of images needed 1 support image and 15 query images. For the 5-way 5-shot classification task, 5 different categories of images were required, and each category of images needed 5 support images and 15 query images. The optimization algorithm Adam (Kingma and Ba, 2014) was used, the initial learning rate was set to 0.0001, and every 10,000 episodes the learning rate would be reduced. During the testing process, 600 episodes were randomly constructed from the testing set, and the top-1 mean accuracy and 95% confidence intervals (model's skill having a 95% probability to correctly generalize) were calculated. Note that the proposed SACovaMNet model was trained from scratch in an end-to-end manner and did not require fine-tuning.

1 https://www.kaggle.com/datasets/khaledelsayedibrahim/fishclassifierfinal

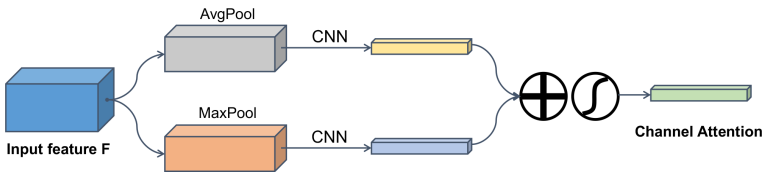


FIGURE 4
Architecture of the proposed DPCACBAM. The input feature map is subjected to local average pooling and maximum pooling, and then the features obtained after passing through two CNN networks are summed element-wise, and finally a channel attentionfeature is generated through a sigmoid.

For QUT fish dataset, due to the small sample size, only the 5-way 1-shot classification experiment was conducted. In the episodic training mechanism, in each category of each episode, there was 1 support image and 2 query images. Other experimental settings remained unchanged.

In order to evaluate the performance of our model on the fish datasets, a selection of state-of-the-art methods commonly used in few-shot fine-grained images were considered for comparison, including baseline CovaMNet (Li et al., 2019), Matching Nets (Vinyals et al., 2016), Prototypical Nets (Snell et al., 2017), MAML (Finn et al., 2017), FRN (Wertheimer et al., 2021), and TDM (Lee et al., 2022). MAML and FRN use the method of meta-learning, Matching Nets, Prototypical Nets and CovaMNet use the method of metric learning, and TDM uses a transferable attention module. We use the TDM method with both FRN and Prototypical Net. For these comparative models, their experimental setup followed the settings from their original work. The SACovaMNet model employed a four-layer convolutional network with a kernel size 64 of each convolutional layer as an embedding module.

4.3 Comparison with state-of-the-arts

The experimental results are shown in Table 1, where, the second column indicates whether the method needs to be fine-tuned; the third and the fourth columns indicate the 5-way 1-shot and the 5-way 5-shot classification accuracies on the WildFish dataset, with 95% confidence intervals; the fifth and the sixth columns represent the 5-way 1-shot and the 5-way 5-shot classification accuracies on the fishclassifierfinal dataset, with 95% confidence intervals; the seventh column represent the 5-way 1-shot classification accuracies on the QUT fish dataset, with 95% confidence intervals. SACovaMNet indicates the method proposed in Section 3.2, and SACovaMNet* indicates the method proposed in Section 3.3. From Table 1, it can be seen that the baseline is more suitable for the fish datasets than other methods, which appears to prove that it was the correct choice for the baseline method to utilize the second-order covariance metric measure. Experimental results have shown that the proposed method outperforms state-of-the-art methods with higher accuracies in all

TABLE 1 The 5-way 1-shot and the 5-way 5-shot classification accuracies on the three datasets, *i.e.*, WildFish, fishclassifierfinal, and QUT fish dataset, with 95% confidence intervals.

Model	Fine-tuning	5-Way Accuracy(%)				
		WildFish		fishclassifierfinal		QUT fish dataset
		1-shot	5-shot	1-shot	5-shot	1-shot
Matching Nets (2016)	N	49.37	56.76	39.84	43.64	60.40
Prototypical Nets (2017)	N	49.81	79.87	51.55	75.49	67.11
MAML (2017)	Y	61.93	76.40	47.73	64.45	74.06
CovaMNet (2019)	N	70.87	84.33	54.54	68.52	66.86
FRN (2021)	N	64.12	80.81	45.42	66.41	61.05
FRN+TDM (2022)	N	43.71	81.66	41.92	69.03	37.03
ProtoNet+TDM (2022)	N	60.23	78.79	52.51	73.03	61.05
SACovaMNet	N	71.44	85.88	58.89	69.01	68.85
SACovaMNet*	N	72.68	86.12	59.28	73.82	70.52

cases. Matching Nets and Prototypical Nets are the earliest few-shot learning methods, and the network structure is simple, so the performance in few-shot fine-grained image classification is not satisfactory; and MAML uses a strategy of meta-learning and fine-tuning, so the effect has been improved. CovaMNet does not adopt the common first-order metric, but uses the second-order metric method, because the details of fine-grained images are preserved, resulting in higher accuracy. FRN achieves better classification results by reconstructing the feature space. The effect of TDM on FRN is not as good as that on Prototypical Nets. This is because FRN itself has more parameters than Prototypical Nets. After adding TDM, overfitting occurs when the number of samples is set to be very small, resulting in unsatisfactory results. Compared to the meta-learning-based MAML that needs to be fine-tuned, our method not only has a simple network structure, but also has a simple training process and short training time, additionally in this case it also achieves high accuracy. And the recent TDM has poor performance mainly because there are very few training samples, with the unsatisfactory results especially on the QUT fish dataset. Compared with other methods, the proposed method demonstrates state-of-the-art capabilities, which validates that the novel AM module, namely Sandwich Attention, can better solve the problem of few-shot fine-grained fish image classification.

4.4 Ablation study

We then conducted ablation study to experimentally demonstrate the effectiveness of our different design choices. For this ablation study, the three datasets mentioned in 4.1 were used and the same convolutional layers as the baseline architecture were also employed. The experimental settings were consistent with those in 4.2. The proposed module design process was divided into two parts, the first part to be examined was to add effective attention to solve the problem which was encountered on the fish datasets, and the second part considered how to incorporate attention modules that were effective for problem solving. The details of each experiment are explained below.

The first line of experimental results in Table 2 is the 5-way 1-shot and 5-way 5-shot classification accuracies obtained by the baseline method CovaMNet on three datasets (i.e., WildFish, fishclassifierfinal, QUT fish dataset); the second line of results shows where after the features were extracted through the convolutional layer of the baseline, the features were passed through the CBAM (Woo et al., 2018) module to obtain the accuracy results of the 5-way 1-shot and 5-way 5-shot; the third line of results shows where the feature was extracted by the convolutional layer on the baseline, and then was passed through the CBAM module (Woo et al., 2018) and the MHSA module (Srinivas et al., 2021), and the features obtained through the two AM modules were paralleled before finally being sent to the classification network to obtain the 5-way 1-shot and 5-way 5-shot accuracy results; the results in the fourth line show where the features were passed through the CBAM module (Woo et al., 2018) and the MHSA module (Srinivas et al., 2021) after the features were extracted in the convolutional layer on the baseline, the three features obtained by the two AM modules and the features obtained by the original extraction were paralleled and then sent to the classification network to obtain the 5-way 1-shot and 5-way 5-shot accuracy results.

Through the comparison of experimental results, it can be found that the original feature map extracted by the convolutional layer has been paralleled with the CBAM and MHSA modules, forming our Sandwich Attention module, such a network structure can allow the network to more comprehensively consider the importance of different regions and channels of the fish image feature map, weight the feature map more accurately, parallel connection with the feature map can effectively ensure the integrity of the original information, so that our experimental results are significantly higher than our baseline.

4.5 Results visualization

For qualitative analysis, the results are presented in the form of t-SNE diagram (Van der Maaten and Hinton, 2008), which is a machine learning algorithm for nonlinear dimensionality reduction, and usually reduces high-dimensional data to 2 dimensions or 3 dimensions for visualization. Here we show the output visualization results of the baseline CovaMNet, SACovaMNet mentioned in 3.2, and SACovaMNet* mentioned in 3.3, on the fishclassifierfinal dataset for 5-way 5-shot classification

TABLE 2 Ablation study on different choices and connections of AM modules, in terms of the 5-way 1-shot and 5-way 5-shot classification accuracies on the three datasets, *i.e.*, WildFish, fishclassifierfinal, and QUT fish dataset, with 95% confidence intervals.

	5-Way Accuracy(%)				
	WildFish		fishclassifierfinal		QUT fish dataset
	1-shot	5-shot	1-shot	5-shot	1-shot
Baseline	70.87	84.33	54.54	68.52	66.86
CBAM	73.63	85.41	57.23	68.52	68.06
CBAM+MHSA	72.97	85.29	57.61	68.21	67.04
CBAM+feature+MHSA (Ours)	71.44	85.88	58.89	69.01	68.85

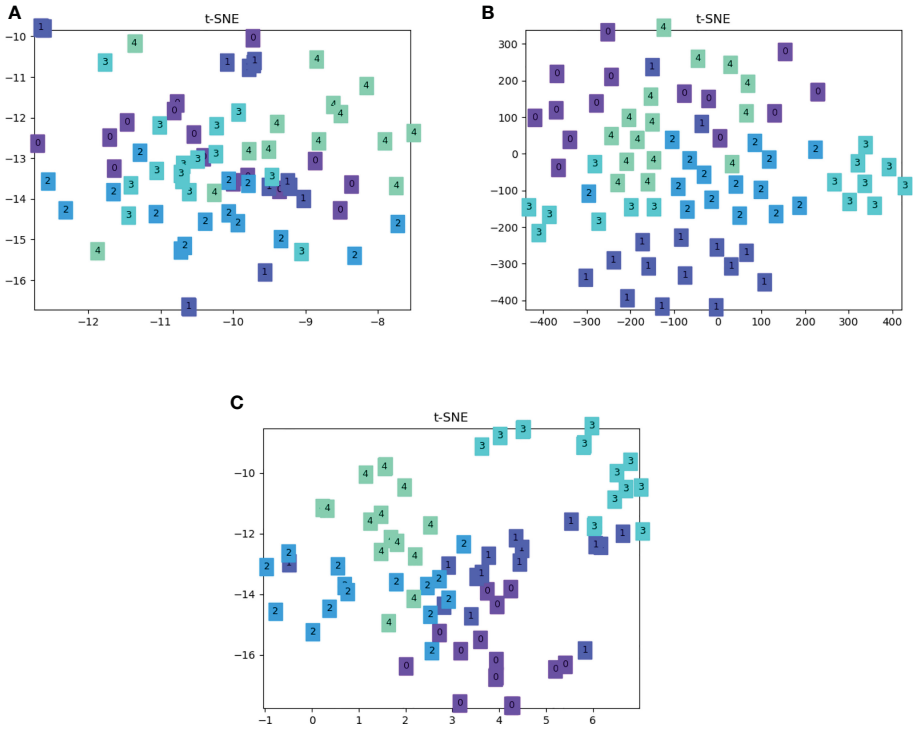


FIGURE 5
Visualization comparison of the t-SNE on baseline CovaMNet, SACovaMNet, and SACova-MNet*. The same color represents one category. **(A)** Visualization of t-SNE on baseline CovaMNet. **(B)** Visualization of t-SNE on SACova-Mnet. **(C)** Visualization of t-SNE on SACova-MNet*.

tasks. The same color in the figure represents the data of the same category. It can be seen from Figure 5A that there is a problem of overlap between different categories, and the boundary of each category is unclear, which will lead to poor classification effects. In Figure 5B, the situation where there is overlap between different categories is reduced, however the data between the same category is relatively scattered. In comparison, the clustering effect in Figure 5C is better, and the boundaries between categories are clearer. The results indicate that our method can make the classification more accurate.

5 Conclusion

In this paper, an approach called SACovaMNet was proposed for few-shot fine-grained marine fish species classification to address the problems caused by a lack of marine fish data and difficulties in classification. The proposed SACovaMNet can extract fish features in detail by fusing CBAM and MHSA in the case of few-shot settings. At the same time, DPCACBAM is proposed to correctly locate the identified objects and key feature information to improve the accuracy of the fine-grained classification, while also applying a second-order covariance metric for similarity comparison that fully takes into account the local similarity information of the feature maps. Based on extensive experiments, the proposed method is shown to be superior to the state-of-the-art methods and the training process is much simpler, providing a basis for research in marine life conservation and marine production.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

JZ, LH, YW, and XW designed the study and wrote the draft of the manuscript with contributions from YX and MY. YX and MY collected the marine fish image datasets. YW and XW devised the method. JZ and LH performed the experiments. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation of China (No. 32073029) and the Key Project of Shandong Provincial Natural Science Foundation (No. ZR2020KC027).

Acknowledgments

We thank the Intelligent Information Sensing and Processing Lab at Ocean University of China for their computing servers and collaboration during experiments. We also thank Leon Bevan Bullock for his suggestions on manuscript writing. We kindly thank the Editor Dr. Hongsheng Bi for his efforts to handle this manuscript

and all the reviewers for their constructive suggestions that helped us to improve our present manuscript.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alsmadi, M. K., and Almarashdeh, I. (2022). A survey on fish classification techniques. *J. King Saud Univ. - Comput. Inf. Sci.* 34, 1625–1638. doi: 10.1016/j.jksuci.2020.07.005

Alsmadi, M. K., Tayfour, M., Alkhasawneh, R. A., Badawi, U., Almarashdeh, I., and Haddad, F. (2019). Robust feature extraction methods for general fish classification. *Int. J. Electrical Comput. Eng.* 9, 5192–5204. doi: 10.11591/ijece.v9i6

Anantharajah, K., Ge, Z., McCool, C., Denman, S., Fookes, C. B., Corke, P., et al. (2014). "Local inter-session variability modelling for object classification," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*. 309–316.

Chen, W.-Y., Liu, Y.-C., Kira, Z., Wang, Y.-C. F., and Huang, J.-B. (2019). "A closer look at few-shot classification," in *Proceedings of the International Conference on Learning Representations*. 1–17.

Chen, G., Sun, P., and Shang, Y. (2017). "Automatic fish classification system using deep learning," in *Proceedings of the International Conference on Tools for Artificial Intelligence*. 24–29.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proceedings of the International Conference on Learning Representations*. 1–22.

Finn, C., Abbeel, P., and Levine, S. (2017). "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the International Conference on Machine Learning*.

Guo, Z., Gu, Z., Zheng, B., Dong, J., and Zheng, H. (2022). Transformer for image harmonization and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1–19. doi: 10.1109/TPAMI.2022.3207091

Guo, Z., Zhang, L., Jiang, Y., Niu, W., Gu, Z., Zheng, H., et al. (2020). "Few-shot fish image generation and classification," in *Proceedings of the Global Oceans 2020: Singapore–US Gulf Coast*. 1–6.

He, J., Kortylewski, A., and Yuille, A. (2023). "CORL: Compositional representation learning for few-shot classification," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3890–3899.

Hochreiter, S., Younger, A. S., and Conwell, P. R. (2001). "Learning to learn using gradient descent," in *Proceedings of the International Conference on Artificial Neural Networks*. 87–94.

Hou, R., Chang, H., MA, B., Shan, S., and Chen, X. (2019). "Cross attention network for few-shot classification," in *Proceedings of the Advances in Neural Information Processing Systems*. 4005–4016.

Kartika, D. S. Y., and Herumurti, D. (2016). "Koi fish classification based on HSV color space," in *Proceedings of the International Conference on Information & Communication Technology and Systems*. 96–100.

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv*. doi: 10.48550/arXiv.1412.6980

Lee, K., Maji, S., Ravichandran, A., and Soatto, S. (2019). "Meta-learning with differentiable convex optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10657–10665.

Lee, S., Moon, W., and Heo, J.-P. (2022). "Task discrepancy maximization for fine-grained few-shot classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5331–5340.

Li, X., Li, Y., Zheng, Y., Zhu, R., Ma, Z., Xue, J.-H., et al. (2023). ReNAP: Relation network with adaptiveprototypical learning for few-shot classification. *Neurocomputing* 520, 356–364. doi: 10.1016/j.neucom.2022.11.082

Li, X., Wu, J., Sun, Z., Ma, Z., Cao, J., and Xue, J.-H. (2021). BSNet: Bi-similarity network for few-shot fine-grained image classification. *IEEE Trans. Image Process.* 30, 1318–1331. doi: 10.1109/TIP.2020.3043128

Li, J., Xu, W., Deng, L., Xiao, Y., Han, Z., and Zheng, H. (2022). Deep learning for visual recognition and detection of aquatic animals: A review. *Rev. Aquac.* 15, 1–25. doi: 10.1111/raq.12726

Li, W., Xu, J., Huo, J., Wang, L., Gao, Y., and Luo, J. (2019). "Distribution consistency based covariance metric networks for few-shot learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*. 8642–8649.

Li, X., Yu, L., Fu, C.-W., Fang, M., and Heng, P.-A. (2020). Revisiting metric learning for few-shot image classification. *Neurocomputing* 406, 49–58. doi: 10.1016/j.neucom.2020.04.040

Liu, Y., Bai, Y., Che, X., and He, J. (2022). "Few-shot fine-grained image classification: A survey," in *Proceedings of the International Conference on Natural Language Processing*. 201–211.

Liu, P., Zhang, C., Qi, H., Wang, G., and Zheng, H. (2022). Multi-attention DenseNet: A scattering medium imaging optimization framework for visual data pre-processing of autonomous driving systems. *IEEE Trans. Intelligent Transport. Syst.* 23, 25396–25407. doi: 10.1109/TITS.2022.3145815

Luo, Z., Zou, Y., Hoffman, J., and Fei-Fei, L. F. (2017). "Label efficient learning of transferable representations acrosss domains and tasks," in *Proceedings of the Advances in Neural Information Processing Systems*. 165–177.

McGlamery, B. L. (1980). "A computer model for underwater camera systems," in *Proceedings of the Ocean Optics VI*. 221–231.

Nichol, A., Achiam, J., and Schulman, J. (2018). On first-order meta-learning algorithms. *arXiv*. doi: 10.48550/arXiv.1803.02999

Peng, Z., Li, Z., Zhang, J., Li, Y., Qi, G.-J., and Tang, J. (2019). "Few-shot image recognition with knowledge transfer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 441–449.

Prasetyo, E., Suciati, N., and Fatichah, C. (2022). Multi-level residual network VGGNet for fish species classification. *J. King Saud University-Computing Inf. Sci.* 204, 5286–5295. doi: 10.1016/j.jksuci.2021.05.015

Rathi, D., Jain, S., and Indu, S. (2017). "Underwater fish species classification using convolutional neural network and deep learning," in *Proceedings of the International Conference on Advances in Pattern Recognition*. 1–6.

Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J. B., et al. (2018). "Meta-learning for semi-supervised few-shot classification," in *Proceedings of the International Conference on Learning Representations*. 1–15.

Rusu, A. A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., et al. (2019). "Meta-learning with latent embedding optimization," in *Proceedings of the International Conference on Learning Representations*. 1–17.

Shevchenko, V., Eerola, T., and Kaarna, A. (2018). "Fish detection from low visibility underwater videos," in *Proceedings of the International Conference on Pattern Recognition*. 1971–1976.

Shi, Z., Guan, C., Li, Q., Liang, J., Cao, L., Zheng, H., et al. (2022). Detecting marine organisms *via* joint attention-relation learning for marine video surveillance. *IEEE J. Ocean. Eng.* 47, 959–974. doi: 10.1109/JOE.2022.3162864

Snell, J., Swersky, K., and Zemel, R. (2017). "Prototypical networks for few-shot learning," in *Proceedings of the Advances in Neural Information Processing Systems*. 4077–4087.

Srinivas, A., Lin, T.-Y., Parmar, N., Shlens, J., Abbeel, P., and Vaswani, A. (2021). "Bottleneck transformers for visual recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16519–16529.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. (2018). "Learning to compare: Relation network for few-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1199–1208.

Tharwat, A., Hemedan, A. A., Hassanien, A. E., and Gabel, T. (2018). A biometric-based model for fish species classification. *Fish. Res.* 204, 324–336. doi: 10.1016/j.fishres.2018.03.008

Van der Maaten, L., and Hinton, G. (2008). Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9, 2579–2605.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). "Attention is all you need," in *Proceedings of the Advances in Neural Information Processing Systems*. 5998–6008.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., and Wierstra, D. (2016). "Matching networks for one shot learning," in *Proceedings of the Advances in Neural Information Processing Systems*. 3630–3638.

Wei, X.-S., Song, Y.-Z., Mac Aodha, O., Wu, J., Peng, Y., Tang, J., et al. (2021). Fine-grained image analysis with deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 8927–8948. doi: 10.1109/TPAMI.2021.3126648

Wertheimer, D., Tang, L., and Hariharan, B. (2021). "Few-shot classification with feature map reconstruction networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8012–8021.

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). "CBAM: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision*. 3–19.

Zhang, C., Cai, Y., Lin, G., and Shen, C. (2020). "DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12203–12213.

Zhang, N., Donahue, J., Girshick, R., and Darrell, T. (2014). "Part-based r-CNNs for fine-grained category detection," in *Proceedings of the European Conference on Computer Vision*. 834–849.

Zhao, B., Feng, J., Wu, X., and Yan, S. (2017). A survey on deep learning-based fine-grained object classification and semantic segmentation. *Int. J. Automation Comput.* 14, 119–135. doi: 10.1007/s11633-017-1053-3

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., et al. (2021). Application of machine learning in intelligent fish aquaculture: A review. *Aquaculture* 540, 736724. doi: 10.1016/j.aquaculture.2021.736724

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., et al. (2021). A comprehensive survey on transfer learning. *Proc. IEEE* 109, 43–76. doi: 10.1109/JPROC.2020.3004555

Zhuang, P., Wang, Y., and Qiao, Y. (2018). "WildFish: A large benchmark for fish recognition in the wild," in *Proceedings of the ACM International Conference on Multimedia*. 1301–1309.

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 16 March 2023
DOI 10.3389/fmars.2023.1077623

Check for updates

OPEN ACCESS

EDITED BY
Mark C. Benfield,
Louisiana State University, United States

REVIEWED BY
Bruno Buongiorno Nardelli,
National Research Council (CNR), Italy
Robert J. Frouin,
University of California, San Diego,
United States

*CORRESPONDENCE
Joana Roussillon
joana.roussillon@gmail.com

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 23 October 2022
ACCEPTED 24 February 2023
PUBLISHED 16 March 2023

CITATION
Roussillon J, Fablet R, Gorgues T, Drumetz L, Littaye J and Martinez E (2023) A Multi-Mode Convolutional Neural Network to reconstruct satellite-derived chlorophyll-a time series in the global ocean from physical drivers.
Front. Mar. Sci. 10:1077623.
doi: 10.3389/fmars.2023.1077623

COPYRIGHT
© 2023 Roussillon, Fablet, Gorgues, Drumetz, Littaye and Martinez. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

A Multi-Mode Convolutional Neural Network to reconstruct satellite-derived chlorophyll-a time series in the global ocean from physical drivers

Joana Roussillon[1]*, Ronan Fablet[2], Thomas Gorgues[1], Lucas Drumetz[2], Jean Littaye[1] and Elodie Martinez[1]

[1]Laboratoire d'Océanographie Physique et Spatiale, CNRS/IFREMER/IRD/UBO, Institut Universitaire Européen de la Mer, Plouzané, France, [2]IMT Atlantique, UMR CNRS LabSTICC, Technopole Brest Iroise, Brest, France

Time series of satellite-derived chlorophyll-a concentration (Chl, a proxy of phytoplankton biomass), continuously generated since 1997, are still too short to investigate the low-frequency variability of phytoplankton biomass (e.g. decadal variability). Machine learning models such as Support Vector Regression (SVR) or Multi-Layer Perceptron (MLP) have recently proven to be an alternative approach to mechanistic ones to reconstruct Chl synoptic past time-series before the satellite era from physical predictors. Nevertheless, the relationships between phytoplankton and its physical surrounding environment were implicitly considered homogeneous in space, and training such models on a global scale does not allow one to consider known regional mechanisms. Indeed, the global ocean is commonly partitioned into biogeochemical provinces (BGCPs) into which phytoplankton growth is supposed to be governed by regionally-"homogeneous" processes. The time-evolving nature of those provinces prevents imposing *a priori* spatially-fixed boundary constraints to restrict the learning phase. Here, we propose to use a multi-mode Convolutional Neural Network (CNN), which can spatially learn and combine different modes, to globally account for interregional variabilities. Each mode is associated with a CNN submodel, standing for a mode-specific response of phytoplankton biomass to the physical forcing. Beyond improving performance reconstruction, we show that the different modes appear regionally consistent with the ocean dynamics and that they may help to get new insights into physical-biogeochemical processes controlling phytoplankton spatio-temporal variability at global scale.

KEYWORDS

Convolutional Neural Networks, attention mechanisms, satellite ocean color, phytoplankton physical drivers, biogeochemical regions, neural networks interpretability, time-series regression, global scale

1 Introduction

Phytoplankton, the microalgae that populate the upper sunlit layers of the ocean, plays a key role in the global carbon cycle and fuels the oceanic food web. It accounts for half of the total carbon fixation in the global biosphere through photosynthesis (Mélin and Hoepffner, 2011) and conditions the oceanic protein production on which ~3,3 billion people rely for their alimentation (FAO, 2020). Thus, understanding and monitoring phytoplankton biomass past and current spatio-temporal variability is of crucial importance to predict and thus anticipate its future evolution in the context of climate change.

Ocean color satellite observations allow documentation of its synoptic variations. Global surface chlorophyll-a concentrations (Chl, a proxy of phytoplankton biomass) can be retrieved from space since the launch of the "Coastal Zone Color Scanner" (CZCS) which has operated from 1978 to 1986. At the end of 1997, the launch of the SeaWiFS sensor, followed by others, was the beginning of 25 years of continuous observations. Although ocean color remote sensing products present a number of uncertainties [due among others to radiometric properties and stability of the sensor, the conditions in the atmosphere or water, the design of the algorithm or the irregular spatio-temporal sampling of the ocean, (Gregg and Casey, 2007; IOCCG, 2019)], radiometric observations have allowed one to point out regional seasonal and interannual phytoplankton variability and to provide new insights about mechanisms driving its spatio-temporal variations (e.g., Longhurst, 1995; McClain et al., 2004; Messié and Chavez, 2012; Racault et al., 2017). However, available ocean color time-series remain too short to inform without ambiguity the basin-scale phytoplankton response to natural decadal climate cycles (Martinez et al., 2009; d'Ortenzio et al., 2012), as well as to derive reliable anthropogenic induced long-term trends for which at least 30-40 years of homogeneous observations would be required (Henson et al., 2010). Some *in-situ* biogeochemical observatories have locally collected long-term time series, but the network coverage is far too sparse to study basin-scale evolutions (Henson et al., 2016). Moreover, if coupled physical-biogeochemical models are able to reproduce the main past global Chl interannual variations, large discrepancies are reported regarding decadal variabilities (Henson et al., 2009b; Patara et al., 2011).

In that context, data-driven methods have appeared to be relevant alternative approaches to reconstruct long-term, continuous and homogeneous phytoplankton time-series based on satellite observations (Schollaert Uz et al., 2017; Martinez et al., 2020a; Martinez et al., 2020b). Phytoplankton growth is limited by light and nutrient availability (e.g., nitrogen, phosphorus, iron). Thus, along with a variety of other biological factors influenced by temperature and/or seascape connectivity [*e.g.* phytoplankton physiology (Grimaud et al., 2017) and ecology (Boyd et al., 2010; Winder and Sommer, 2012)], the spatio-temporal distribution of surface phytoplankton on a global scale is strongly shaped by changes in the supply of nutrients to the sunlit upper ocean through vertical exchange. Phytoplankton changes can also be related to other known processes as the predation by grazers, such as zooplankton (the so-called "top-down control") whose variability can also be related to their physical environment (e.g., temperature; Beaugrand et al., 2002). Consequently, as physical ocean and atmospheric dynamics largely drive global phytoplankton variability (Wilson and Adamec, 2002; Wilson and Coles, 2005; Kahru et al., 2010; Feng et al., 2015), statistical relationships can be determined between some physical predictors and Chl. Once such statistical relationships are established and validated, they provide new means to retrieve past and future Chl based on physical data from satellites (with a longer time period than for Chl) and/or numerical model simulations.

Schollaert Uz et al. (2017) were the first to use this approach in the tropical Pacific Ocean ([20°S-20°N]) with a linear canonical correlation analysis applied on Sea Surface Temperature (SST) and Sea Surface Height (SSH) vs. Chl. They reproduced most of the Chl variability within 10° around the equator over 1958-2008, and evidenced decadal variations corresponding to the Pacific Decadal Oscillation (PDO). Martinez et al. (2020a) extended such an approach to the global ocean using a Support Vector Regression (SVR) model relying on a larger number of surface oceanic and atmospheric predictors from numerical models. Given their capacity to model complex non-linear relationships between data (Hornik et al., 1989), dense neural network models (namely Multi-Layer Perceptrons, MLPs) have been successfully applied in geoscience and biogeochemical oceanography to regress some variables from predictors (Long et al., 2014; Sauzède et al., 2016; Sammartino et al., 2020). Thus, in a second study, Martinez et al. (2020b) extended their work to satellite observations and showed that an MLP outperforms the SVR to retrieve both Chl spatial and temporal patterns. However, in these two studies, the considered point-wise machine learning models explicitly relied on spatial coordinates (periodized longitude and latitude) and temporal information (periodized month) as predictors. This may impede the ability of neural networks to capture changes in the boundaries of biogeochemical provinces (BGCPs) that are naturally time-evolving (Oliver and Irwin, 2008; Devred et al., 2009; Reygondeau et al., 2013). In addition, these results remained hard to interpret in terms of processes involved in the Chl reconstruction and variability, whereas data-driven approaches have great potential to discover new patterns, structure and relationships in scientific datasets (Bergen et al., 2019). Understanding what drives neural network output is also essential to ensure they behave appropriately to the field of application (Xie et al., 2020) so as to enhance the degree of confidence that can be placed in them.

Besides MLPs, other deep learning schemes, in particular Convolutional Neural Networks (CNNs), have shown a much greater ability to decompose and represent the space-time variations. We may cite numerous successful applications in Earth science forecasting (Haidar and Verma, 2018; Ham et al., 2019; Pan et al., 2019; Chattopadhyay et al., 2020; Weyn et al., 2020) and reconstruction (Cooke and Scott, 2019; Sun et al., 2019; Ai et al., 2020; Kim et al., 2020; Jeon et al., 2021; Meng et al., 2021; Pyo et al., 2021) problems, including studies focusing on Chl data (Yu et al., 2020; Ye et al., 2021). CNNs assume translation equivariance of the input data (Goodfellow et al., 2016), so that they cannot learn region-specific representations when trained over the whole ocean (Cachay et al., 2020). On the other hand, the *a priori* definition of BGCPs to train region-specific CNN models

are not fully relevant due to their time-evolving nature, especially as they are expected to be impacted by climate changes (Polovina et al., 2008; Irwin and Olivier, 2009; Reygondeau et al., 2020). By contrast, attention mechanisms (Chen et al., 2017; Jetley et al., 2018) provide a generic approach to account for different modes of variability within CNNs. For instance, Pyo et al. (2021) inserted such attention blocks into a CNN and improved both performance and interpretability to predict cyanobacteria cells from spatialized water quality predictors.

Here, we introduce a regular CNN, then a CNN with attention mechanisms, referred to as a Multi-Mode Convolutional Neural Network (CNN_{MM}), to reconstruct phytoplankton dynamics from physical predictors. The statistical models are trained between ocean color observations vs. physical variables from satellite observations and reanalysis outputs. The study is conducted from 1998 to 2015. We demonstrate that the CNN_{MM} scheme outperforms the state-of-the-art MLP data-driven approach and illustrate its relevance to analyze the space-time variabilities of physics-driven phytoplankton dynamics.

2 Material and methods

2.1 Chl observations, physical predictors and climate index

The different datasets used in this study are briefly described here. They comprise the same products as those used in Martinez et al. (2020b), complemented with bathymetry data.

Several ocean color sensors embedded on different satellite platforms have been operating since 1997. However, their limited lifespan and differences in calibration lead to inter-sensor bias and make them irrelevant for decadal time-scales studies. In order to provide more homogeneous data, the European Space Agency (ESA) has produced the Ocean Color Climate Change Initiative (OC-CCI) Chl products, hereafter referred to as $Chl_{OC\text{-}CCI}$. Radiometric observations from the Sea-viewing Wide Field-of-View Sensor (SeaWIFS, 1997-2010), the Moderate Resolution Imaging Spectroradiometer (MODIS, 2002-present), the MEdium Resolution Imaging Spectrometer (MERIS, 2002-2012) and the Visible and Infrared Imaging Radiometer Suite (VIIRS, 2012-ongoing) were consistently reprocessed to produce a global longer-term and "bias-corrected" ocean-color time series (Sathyendranath et al., 2019). Level 3 products from v4.2 were downloaded at https://oceancolor.gsfc.nasa.gov/l3/, with a monthly temporal resolution on a 1° grid and over 50°N-50°S to reduce the number of missing data due to cloud cover and/or permanent night in wintertime at high latitudes. Even though the OC-CCI Chl products benefit from merged data from multiple satellite missions to provide a better spatial and temporal coverage and a more consistent long-term time series, it is worth noting that these data still present some uncertainties. Indeed, with a global uncertainty of about 30% for derived Chl (IOCCG, 2019; Sathyendranath et al., 2019), reported accuracies may vary significantly regionally (Szeto et al., 2011) and seasonally (Bisson et al., 2021). Thus, one should be aware that the satellite-derived Chl used in this study may not always properly describe the *in-situ* Chl variability. Yet, satellite-derived Chl, with the spatio-temporal resolution chosen in this study, are still commonly used to study global intra-annual to longer timescale variations in phytoplankton biomass.

Short-Wave radiations (SW), referred to total solar irradiance with wavelengths in the range of 300-3000 nm, are considered as a proxy of Photosynthetically Active Radiation (PAR, 400-700 nm) used for phytoplankton growth. SW are here preferred to PAR as they are available over the historical period (e.g. from the 50's) from ocean and atmosphere numerical model outputs, that do not include irradiance in the photosynthetic range, bearing in mind that the model developed in this study is meant to be later used to reconstruct phytoplankton past long-term time series. The reanalysis daily product NCEP/NCAR (Kalnay et al., 1996) delivered by the National Oceanic and Atmospheric Administration (NOAA) with a resolution of 2°x2° is used in this study and available at https://psl.noaa.gov/data/gridded/data.ncep.reanalysis.derived.html.

SST is usually considered as a good proxy of ocean vertical mixing, being itself related to nutrient availability in the upper ocean (e.g., Wilson and Coles, 2005; Behrenfeld et al., 2006; Martinez et al., 2009; d'Ortenzio et al., 2012). Moreover, SST can impact phytoplankton metabolic rates (Lewandowska et al., 2014). The monthly 1°x1° SST of the Reyn_SmithOIv2 dataset produced at NOAA using both *in situ* and satellite data (Reynolds et al., 2002) was downloaded at http://iridl.ldeo.columbia.edu/.

Sea Level Anomaly (SLA) variability has been shown to be a proxy for the thermocline/pycnocline/nutricline depth variability in most parts of the global ocean (Wilson and Adamec, 2002). The Ssalto/Duacs merged satellite altimetry product of CNES/SALP project is used here. It consists in a weekly product with a 1/3°x1/3° spatial resolution and was retrieved at https://resources.marine.copernicus.eu (accessed on December 2020).

Zonal and meridional surface currents (U and V, respectively) could supply nutrients from remote regions through lateral advection (Messié and Chavez, 2012). The Ocean Surface Current Analysis Real-time (OSCAR) unfiltered product (ESR, 2009) is used here to depict global ocean surface currents. It was generated by NASA Earth Space Research (ESR) at a 1/3° x 1/3° resolution every 5-days from 1993. Horizontal velocities are computed from satellite-sensed SSH gradients, surface vector winds and SST fields with simplified physics. This product allows detection of eddies that range from 100 to 300 km (Dohan, 2017). The data is available from the NASA Physical Oceanography data center at https://podaac.jpl.nasa.gov/dataset/OSCAR_L4_OC_third-deg.

Zonal and meridional surface wind stress (Uera and Vera, respectively) exhibits global large-scale correlation patterns with Chl (Kahru et al., 2010). In the open ocean, increased winds contribute to deepen the mixed layer and thus to either reduce phytoplankton light exposition in subpolar regimes or to increase nutrients availability in subtropical regions. They account for one part of the interannual and decadal mixed layer depth (MLD) variability, that is reflected on phytoplankton bloom timing and magnitude variations (Henson et al., 2009a; Kahru et al., 2010; Martinez et al., 2011). Monthly global atmospheric reanalysis computed by the ECMWF was used. The ERA-Interim 4 product

was downloaded with a spatial resolution of 0.25° x 0.25° at: https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era-interim.

The General Bathymetric Chart of the Oceans (GEBCO) produced under the auspices of the International Hydrographic Organization and the Intergovernmental Oceanographic Commission of UNESCO is used. It consists in a continuous, global terrain model for ocean and land, with a spatial resolution of 15 arc seconds. The GEBCO_2020 product was downloaded at https://www.gebco.net/data_and_products/gridded_bathymetry_data/gebco_2020/.

The monthly Multivariate El Ninõ Southern Oscillation Index (MEI) is provided by the National Oceanic and Atmospheric Administration (NOAA) website at https://psl.noaa.gov/enso/mei/.

The choice of the 8 physical predictors (SW, SST, SLA, U, V, Uera, Vera, Bathy) is motivated by our will to use the most realistic environmental conditions, that only observations allow, to learn relationships with Chl. Among routinely measured oceanic properties, we chose to rely on surface ones only (except for the bathymetry), for which observations are much less scarce at global and interannual scales than the ones below the surface. These variables have also been selected as they are known to be proxies of dynamical processes which drive the variability of phytoplankton to the first order. In addition, deep neural networks are expected to derive other related quantities (e.g., wind curl, eddy kinetic energy, etc) on their own through operations (squares, cubes, gradients, etc), although some subjective choices of predictors can sometimes help the network to identify meaningful relationships.

Moreover, monthly physical fields are used in this study to predict simultaneous monthly Chl, without considering any time-lag. This choice is motivated by the rapid response of phytoplankton growth to changes in physical forcing, with an associated average turnover time of global oceanic plant biomass on the order of a week or less (Falkowski et al., 1998). It is also consistent with the strong large-scale correlation patterns that were previously reported in the literature between environmental forcing and synchronous phytoplankton biomass at monthly timescales (Wilson and Adamec, 2002; Wilson and Coles, 2005; Feng et al., 2015; Schollaert Uz et al., 2017).

2.2 Data pre-processing

The eight physical predictors' datasets are extracted over [1998-2015] and resampled to the same spatio-temporal resolution as Chl, *i.e.* monthly on a 1°x1° grid between 50°N and 50°S. Some missing values (NaN: Not a Number) remained in the different datasets such as on land for oceanic variables. As CNNs cannot account for NaN values for the input predictors, a gap-filling scheme is applied. A classic zero-filling strategy is discarded as it may lead to spurious results especially in coastal areas. Alternatively, we extrapolate missing data using the heat diffusion equation (see Equation 1), that is widely used in the field of computer vision (Aubert et al., 2006):

$$\begin{cases} \frac{\partial u}{\partial t}(t,x) - \Delta u(t,x) = 0, \ \{\ t \in \mathbb{N} \quad t \leq 1000\}, \ x \in \mathbb{R}^2 \\ u(0,x) = \ u_0(x) \end{cases} \quad (Eq.\ 1)$$

where u_0 is the field with a zero-filling scheme for missing data, u the interpolated field, t the iteration step and x the space coordinates. This diffusion is applied to all the input fields involving missing data (as illustrated in Figure S1) but is not needed for the output field (Chl).

Given the well-known log-normal distribution of Chl data, Chl is logarithmically transformed prior to being used in the machine learning schemes. Back-transformation is applied afterwards to the reconstructed log(Chl) (where log stands for the natural logarithm, to the base e) to retrieve Chl fields that can be validated against Chl satellite observations. As classically done in deep learning approaches to stabilize training, we normalize each variable by subtracting its mean from the original values and dividing by its standard deviation over [1998-2015].

2.3 Deep learning schemes

In this study, we explore three different neural architectures: the baseline MLP considered in Martinez et al. (2020b), a basic CNN and the proposed multi-mode CNN. According to our choice of not considering time-lags, those three models have in common to only rely on instantaneous relationships. We detail below these three architectures.

2.3.1 Baseline MLP

We implement the same MLP as in Martinez et al. (2020b). The MLP is composed of seven dense layers (see Table 1) with LeakyReLU activations. We refer the reader to Martinez et al. (2020b) for more details about its architecture. It involves 1,800,000 parameters. We may point out that the MLP applies pixel-wise, that is to say to a vector of input data, corresponding to a predefined set of features defined at each space-time location. Similarly to (Martinez et al., 2020b), the feature vector comprises the following 12 variables: SLA, SST, Uera, Vera, U, V, SW, sin(lat),

TABLE 1 Summary of the models' architectures. CNN_{MM8} corresponds to the multi-mode CNN composed of an attention-based module *W* and 8 CNNs submodels M_i trained in parallel.

Model		Layers	Number of neurons/filters	Number of parameters
MLP		7 dense layers	12:1000:1000:500:500:120:120	~1 800 000
CNN_1		5 convolutional layers	9:16:32:64:128	~100 000
CNN_{MM8}	**W**	3 convolutional layers	9:16:32	~7 000
	M_i	5 convolutional layers	9:16:32:64:128	~100 000

sin(lon), cos(lon), sin(month), cos(month). Cosine and sine of longitude are used to account for periodicity (longitude 0° = longitude 360°), and sine of latitude is used to keep the same ranges of values between longitude and latitude predictors. In a similar manner, months are periodized using sine and cosine of month to account for seasonal similarities (month 1, *i.e.* January, is seasonally related to month 12, *i.e.* December).

2.3.2 Baseline CNN

CNNs, and their variants such as convolutional ResNets (He et al., 2016) and Unets (Ronneberger et al., 2015) are state-of-the-art architectures for a variety of image processing and computer vision applications. They offer a new way of processing multidimensional data by extracting patterns using convolution. Here, we consider a basic CNN architecture composed of a sequence of five 2 dimensional convolutional layers with 3x3 kernel sizes, stride and padding 1x1, and with ReLU activations. We report the details of the mono-mode CNN (hereafter referred to as CNN_1) architecture in Table 1. Overall, it involves ~100,000 parameters. Contrary to the MLP, the CNN applies directly to the concatenation of the 2D fields predictors.

2.3.3 Multi-mode CNN

The proposed multi-mode architecture aims at better accounting for the space-time variabilities of the relationship between plankton dynamics and the physical forcing. Modular neural networks were proposed in the 80's (Micheli-Tzanakou, 1987; Anzai and Shimada, 1988) with the aim of enabling decomposing complex tasks into more practicable sub-parts (Auda and Kamel, 1999; Azam, 2000). They rely on the idea that the combination of several estimators can lead to better results than when using only one. More recently, attention-based mechanisms (Chen et al., 2017; Kirsch et al., 2018) provide means to implement this general concept. As sketched in Figure 1, the proposed architecture applies in parallel *i* CNNs (referred to as M_i). These *i* CNNs have the same architecture than the baseline CNN introduced above, and only differ from one another in the way their respective weights are optimized during training. As such, for a given set of 2D fields predictors, we are provided with *i* outputs with the same size than the target Chl field. We then compute a pixel-wise weighted average of these *i* outputs according to weights computed by the attention-based network W (this product is hereafter referred as "mode"). W is also a CNN with the same architecture than the baseline one, but with 3 convolutional layers only. This CNN also uses as inputs the multivariate 2D fields formed by the physical forcing. Importantly, the last layer of this CNN is a softmax layer, so that the weights are positive and sum to one for each pixel. The key features of this multi-mode CNN architecture are three-fold: (1) it can explicitly account for regional physics-driven variabilities, (2) there is no need to *a priori* delineate BGCPs boundaries, (3) the learnt attention-based module defines the space-time activation domain of each mode, which may improve the interpretability of the network. As summarized in Table 1, the multi-mode CNN for an 8-modes configuration (referred to as CNN_{MM8}) comprises ~807,000 parameters (8*100 000 + 7000).

2.4 Learning settings

For evaluation purposes, the whole database is split into three independent datasets to train, validate and test the deep-learning schemes. We consider non-overlapping time periods for each dataset as sketched in Figure 2: the training is performed over [2003-2010], the validation dataset covers [1998-2001] to monitor the generalization performance of the models during the training phase and select models' parameters through sensitivity tests, and

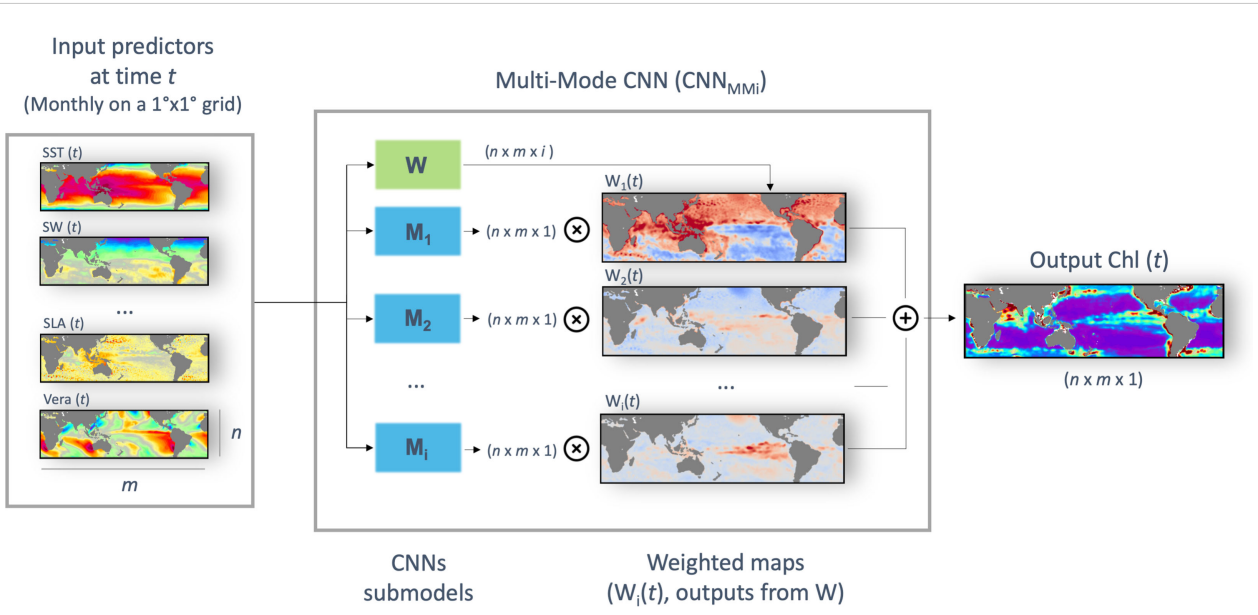


FIGURE 1
Diagram of the CNN_{MMi} architecture. For a given set of input 2D predictors, each of size *n* x *m*, *i* outputs of size *n* x *m* x *1* are computed from the *i* CNN submodels (M_i). Those are spatially weighted according to the *i* dynamic probability maps outputted at each time *t* from the W spatial attention module, and summed to obtain the output Chl 2D field.

reconstructed Chl are compared to satellite Chl over [2012-2015] (i.e., the test time-period). Years 2002 and 2011 are discarded so that the training, test and validation datasets are not auto-correlated. This configuration delivers long-enough test time periods to assess the seasonal and interannual timescales of interest (i.e., El Niño Southern Oscillation - ENSO). It also defines time periods during which the number of ocean color sensors remains the same in the OC-CCI dataset (Sathyendranath et al., 2019) to avoid confusions between possible Chl variations due to switch in sensors or occurring in nature (Gregg et al., 2017).

We train all models using a Mean Squared Error (MSE) loss and Adam optimizer (Kingma and Ba, 2014). The MLP is trained over 200 epochs with a learning rate of 10^{-4} and a dropout of 0.15. The CNNs and CNN_{MMi} are trained over 500 epochs with an initial learning rate of 0.001 that is decreased to 0.0001 at the 400th epoch to stabilize the training. Dropout values of 0.15 and 0.35 are used for the CNN_1 and CNN_{MMi}, respectively, to prevent overfitting (Srivastava et al., 2014)(see respective learning curves in Figure S3). Hyperparameters settings were chosen according to sensitivity tests summarized in (Supplemental Table S1).

During each training run, we assess the score of the trained model on the validation dataset at the end of each epoch and save the one with the best score. We implement all models using Python with the Pytorch library. We run numerical experiments with a GPU NVIDIA Tesla T4 with 32Go of RAM. As recommended by many ethics' guidelines for developers (Vinuesa et al., 2020; Ryan and Stahl, 2021; Taddeo et al., 2021), we also report the carbon footprint of the training phase of each model using the Carbontracker Python library (Anthony et al., 2020). Our computing server is located in France, with a detected averaged carbon intensity of 294.21 gCO2/kWh.

2.5 Evaluation framework

We consider the following three quantitative metrics for evaluation purposes: the root-mean-square error (RMSE, Eq. 2), the coefficient of determination (R^2, Eq. 3) and the linear regression slope are used to compare the reconstructed log(Chl) times series vs. OC-CCI satellite observations:

$$RMSE = \sqrt{\frac{\sum(\log(Chl) - \log(Chl_{OC-CCI}))^2}{N}} \qquad (Eq.\ 2)$$

$$R^2 = \left(\frac{\sum(\log(Chl) - \overline{\log(Chl)})(\log(Chl_{OC-CCI}) - \overline{\log(Chl_{OC-CCI})})}{N * \sigma_{\log(Chl)} * \sigma_{\log(Chl_{OC-CCI})}}\right)^2 \qquad (Eq.\ 3)$$

with N the number of samples, σ the standard deviation and the horizontal bar the time average, both calculated over the considered time period.

Global map of correlation and of normalized RMSE (NRMSE, Eq. 4) of Chl times series vs. OC-CCI satellite observations are also used to assess regional discrepancies:

$$NRMSE = \frac{\sqrt{\sum(Chl - Chl_{OC-CCI})^2}}{\sqrt{\sum(Chl_{OC-CCI})^2}} \qquad (Eq.\ 4)$$

To estimate the model's ability to reproduce seasonal and interannual variabilities, an Empirical Orthogonal Function (EOF) analysis is performed as follows. First, the annual (monthly) Chl_{OC-CCI} average is removed from the initial time series to obtain the seasonal (interannual) Chl anomalies which are then normalized with respect to their standard deviations. We project the reconstructed Chl time series onto these seasonal and interannual Chl_{OC-CCI} spatial patterns and the resulting seasonal and interannual temporal patterns (i.e. the principal components, PCs) are compared to those of Chl_{OC-CCI} using Pearson correlation.

For each pixel, the percentage of variance explained by each of the i modes of the multi-mode CNN_{MMi} is derived to assess their relative importance. It relies on (1) successively reconstructing Chl while putting the probability weights of the corresponding mode to zero, and (2) calculating the difference in RMSE that is observed compared to when Chl is inferred with the full model.

From the obtained i percentages of variance P_k, we further compute, for each pixel, the following entropy-based metric H:

$$H = -\sum_{k=1}^{i} P_k * log_2(P_k) \qquad (Eq.\ 5)$$

It allows us to evaluate to which extent the reconstruction at a given pixel truly results from a multi-mode relationship (large entropy values) or from a single-mode one (low entropy values).

Finally, we also assess the relative importance of each physical predictor to reconstruct Chl using a perturbation-based method as in Kim et al. (2020). From a given CNN_{MMi}, the difference of the RMSE of the predicted Chl when using the initial data vs. randomly shuffled data (both in time and space) for each predictor individually is computed. RMSE differences are normalized so

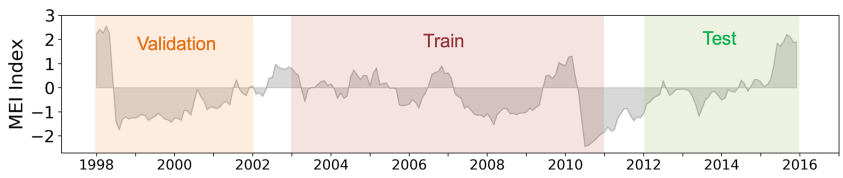


FIGURE 2
Time-series of the MEI. The validation, training and test time periods used to compare the implemented regression models' performances are indicated as orange, red and green filled sections, respectively.

that the relative importance of all the predictors sums up to one for each pixel.

3 Results and discussion

3.1 Performance of the mono-mode CNN vs. MLP baseline

The reconstructed Chl from both the mono-mode CNN_1 and the state-of-the-art MLP are compared to satellite Chl over the [2012-2015] test period to assess the added value of convolutions. When the 12 predictors [namely SLA, SST, Uera, Vera, U, V, SW, sin(lat), sin(lon), cos(lon), sin(month), cos(month)] are used, performances obtained with the MLP and CNN_1 remain close (Table 2). However, the CNN_1 contains almost twenty times less parameters than the state-of-the-art MLP (~100 000 vs. ~1 800 000, respectively), is ten times faster to compute and more than ten times more energy-efficient, supporting that convolutions are better suited to reconstruct Chl.

To avoid learning constraints of time and space, the models are trained removing the spatial coordinates, i.e. on 9 predictors. Results further stress the relevance of convolutional architectures to reconstruct $Chl_{OC\text{-}CCI}$. Indeed, whereas the MLP highly drops in performance (R^2 down to 0.59 and RMSE up to 0.5), the CNN_1 still presents satisfactory scores (R^2 = 0.80 and RMSE = 0.35) (Table 2). These results averaged at global scale are consistent over the three oceanic basins with a higher R^2 between $Chl_{OC\text{-}CCI}$ and CNN_1 than with MLP by 0.23 and 0.24 respectively in the Indian and Pacific oceans and by 0.14 in the Atlantic Ocean (Figure 3 lower row vs. upper row).

Interestingly, removing the temporal predictors (i.e., sin(month) and cos(month)) does not reduce the CNN_1 performance and even tends to slightly improve it (slope of 0.81 vs. 0.77, and interannual correlation coefficient of 0.96 vs 0.94, Table 2). It suggests that temporal predictors only bring redundant information already included into the seasonally-fluctuating physical fields provided as predictors. This result also suggests that the network benefits from being no longer monthly constrained when interannual time-series are considered. Indeed, learning on periodized months may force the network to learn static seasonal phytoplankton bloom characteristics (e.g., start, duration and amplitude) over several years. Thus, it would impede to correctly account for interannual delays in bloom timing or difference in the length of the growing period (Henson et al., 2009a) that can for instance reach ~10 weeks for major ENSO events (Racault et al., 2012) and that would be otherwise considered through other physical fields such as SST.

The CNN_1 is further improved by the addition of two other predictors: the bathymetry and a continental mask. The bathymetry is considered as it would participate to distinguish open ocean ecosystems from coastal ones, where specific processes can occur (shelf break fronts, tidal mixing, river discharge, coastal upwelling, etc) and where the water-leaving radiance measured by ocean color sensors may only partially represent Chl (inorganic particles dominate over phytoplankton concentration). Moreover, as being more spatially resolved than OSCAR data, it is also expected to bring additional information about the ocean circulation (especially concerning the fine-scale dynamic) that is regionally related to the seafloor topography (Gille et al., 2004; Bryan, 2016). The binary continental mask (0 on ocean and 1 on land) is also added because the oceanic predictors are filled over land with data through diffusion (see the data section) inducing that no information on the exact boundary between ocean and land are no longer available. Doing so, results are slightly improved (R^2 = 0.84, RMSE = 0.31 and slope = 0.85) and the CNN_1 better captures Chl spatial structure in some places as observed over the tropical Atlantic Ocean (Supplemental Figure S2).

3.2 Chl reconstruction improvement from mono-mode CNN_1 to multi-mode CNN_{MM8}

Given the overall good performance of the CNN_1, we chose this model as a basis to document the impact of multi-modality. With

TABLE 2 Global performance metrics obtained with the state-of-the-art MLP, CNN_1 and CNN_{MM8} over the [2012-2015] test period.

Predictors		Global scatterplot			Corr. Seas. PC	Corr. Inter. PC	N param	Time computation	Km travelled by car
	Model	R^2	RMSE	Slope					
12	MLP	0.85	0.30	0.84	0.99	0.97	1 840 000	50h13	13.5
	CNN_1	0.86	0.30	0.87	0.99	0.98	99 889	5h	0.95
9 (without sin(lat), cos(lon), sin (lon))	MLP	0.59	0.50	0.57	0.97	0.85	1 836 000	50h13	13.4
	CNN_1	0.80	0.35	0.77	0.99	0.94	99 457	4h53	0.93
7 (without sin(month), cos (month))	CNN_1	0.80	0.35	0.81	0.98	0.96	99 169	4h52	0.92
9 (+ bathymetry + continental binary mask)	CNN_1	0.84	0.31	0.85	0.99	0.95	99 457	4h54	1.04
	CNN_{MM8}	**0.87**	**0.28**	**0.90**	**1.00**	**0.96**	803 920	39h	8.9

The R^2, RMSE and slope metrics are calculated between the reconstructed log(Chl) and satellite log($Chl_{OC\text{-}CCI}$). Correlations of seasonal and interannual 1st principal components from EOF analysis are calculated between the reconstructed Chl and satellite $Chl_{OC\text{-}CCI}$. The number of parameters used and the computation time of the training phase (performed over [2003-2010]) are reported, as well as the associated carbon footprints in equivalent km traveled by car. Performance metrics of the proposed multi-mode approach are highlighted in bold.

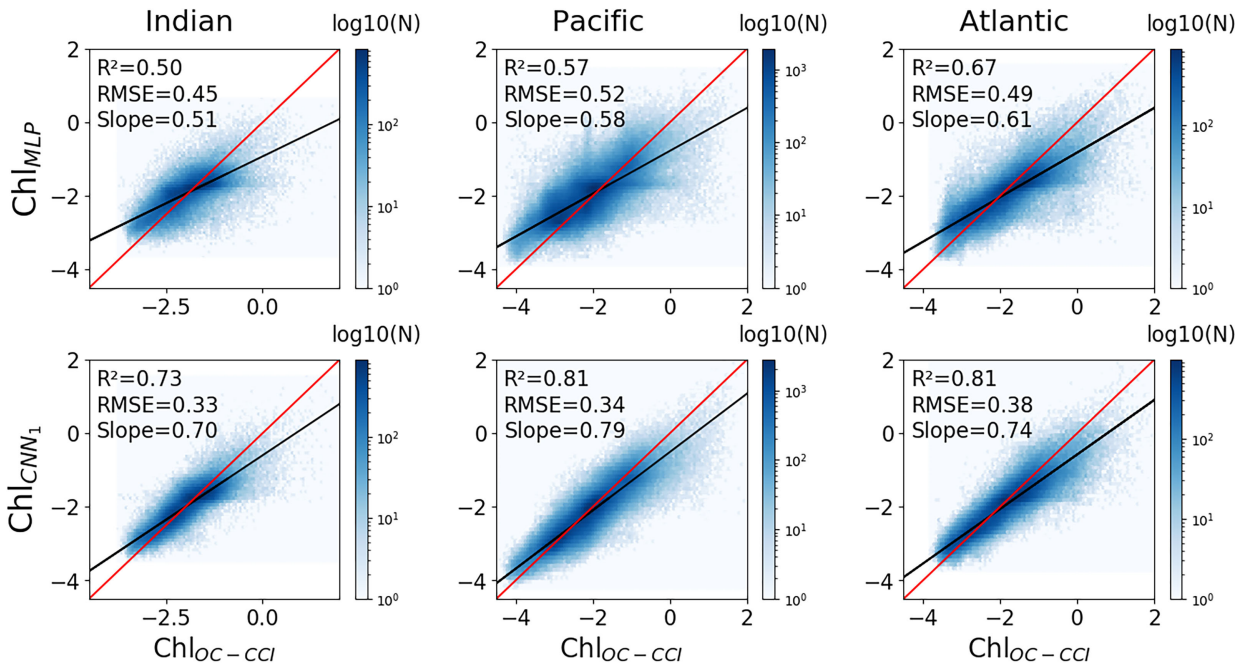


FIGURE 3
Scatterplots of reconstructed log(Chl) from the MLP (upper row) and CNN_1 (lower row) vs. satellite $Chl_{OC\text{-}CCI}$, when explicit geographic predictors (i.e., sin(lat), cos(lon), sin(lon)) are removed from the training phase. Columns correspond to different oceanic basins (left: Indian Ocean, middle: Pacific Ocean, right: Atlantic Ocean). The log of $Chl_{OC\text{-}CCI}$ vs. reconstructed log of Chl regression lines are plotted in black and the 1:1 regression lines are plotted in red. Plots are color-coded according to the density of observations.

the same 9 predictors, performances of the proposed multi-mode CNN_{MMi} schemes are investigated from 1 to 15 modes. R^2 increases from 0.81 up to 0.87 and RMSE decreases from 0.32 down to 0.27 from one to four modes (Figure 4, see Table S2 for details). For both metrics, a plateau is reached from the fourth mode for R^2 and the eighth mode for RMSE. Overall, the CNN_{MM8} model seems to be the best trade-off between performance and computational complexity. Thus, the CNN_{MM8} is investigated hereafter and compared to CNN_1 to further discuss the advantages of the multi-modality.

Time averaged satellite $Chl_{OC\text{-}CCI}$ over the [2012-2015] test period compares reasonably well with that reconstructed from CNN_{MM8} (Figures 5A vs. 5B). The CNN_{MM8} correctly represents the main spatial patterns with, for instance, higher Chl at high latitudes and along the equatorial and eastern boundary upwelling, as well as in the Arabian Sea. The CNN_{MM8} also captures low Chl in the subtropical gyres delimited by the 0.07 $mg.m^{-3}$ mean Chl isocontour. The correlation map computed between $Chl_{OC\text{-}CCI}$ and CNN_{MM8} shows values higher than 0.8 over large parts of the global ocean and especially in the subtropical areas (Figure 5C). Conversely, low correlation values, associated in most cases to high NRMSE (Figure 5D), can be observed at higher latitudes than 40° and in the eastern and tropical part of the Pacific Ocean oligotrophic gyres. This can be due to several factors. In some places, the spatio-temporal resolution (i.e., monthly on a 1° grid) used in the present study may be too coarse to capture the overall Chl variability. In particular, this would mainly explain the lack of correlation observed in the tropical southeastern and northwestern Pacific where the dominating timescales of Chl variability have been very recently reported to be below 30 days (Jönsson et al., 2023; see their Figure 7B). This may also explain part of the Chl underestimation observed in highly energetic areas with mesoscale and sub-mesoscale eddies (<100 km scales) that may impact phytoplankton along dynamical fronts (Lévy et al., 2018). This component of the ocean dynamics might not be sufficiently resolved here, as along the Gulf Stream, the Kuroshio and Agulhas currents and in subantarctic waters along the Antarctic Circumpolar Current (Frenger et al., 2018). In addition, the list of predictors that we used is not exhaustive and variables representative of some biogeochemical and physical mechanisms may be missing. For

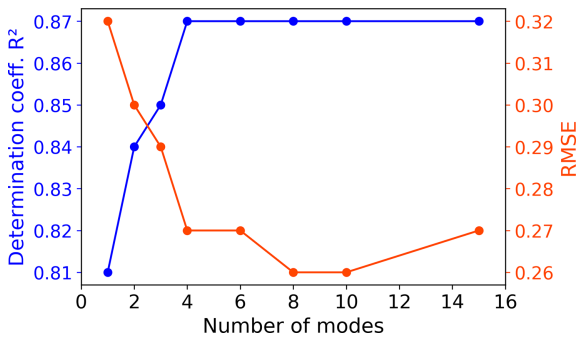


FIGURE 4
Performance evolution according to the number of modes of the CNN_{MMi} models. Metrics are computed over the [1998-2001] validation period during which model parameters are assessed.

instance, terrigenous inputs at the mouths of large rivers (driven by precipitations) supply nutrient rich waters which are not considered in our predictors. In addition, ocean color observations in these regions may rather reflect suspended particles and colored dissolved organic matter (respectively SPM and CDOM) rather than Chl. This could explain the high NRMSE values observed along the Amazon, the Congo and Kunene rivers. More generally, the predictors we used in this study cannot account for some ocean color sources of uncertainties (e.g., atmospheric conditions, solar zenith angle, properties of the sensors, etc), so potentially biased Chl values cannot be fully reproduced by the networks. Moreover, biological effects such as zooplankton grazing (the so-called top-down control), which are not directly accounted for by any of our predictors, may also regionally inhibit the signature of phytoplankton growth on satellite observations, especially at high latitudes. Proxy of iron supply in the open ocean from other external sources, such as dust deposition or hydrothermal vents, are also missing among our predictors. This can limit the ability of our network to distinguish areas of different nutrient (co-)limitations (Moore et al., 2013) and to account for phytoplankton responses driven by the dynamics of these sources, especially in iron-limited High Nutrient Low Chlorophyll (HNLC) regions. As such, one part of the low correlations observed in the eastern tropical Pacific could come from the role played by dust deposition in altering the timing and amplitude of ENSO-related phytoplankton response (Lim et al., 2022a). This could also partly explain low correlations values observed in the northwestern Pacific (Meng et al., 2022), or high NRMSE values observed in the northern Arabian Sea where dust deposition would play a key role in controlling phytoplankton bloom amplitude (Guieu et al., 2019).

Mean difference maps between CNN_{MM8} and CNN_1 in terms of correlation and NRMSE with $Chl_{OC\text{-}CCI}$ illustrate that the CNN_{MM8} improves correlations over most of the global ocean (in red in Figure 5E). Differences higher than 0.3, and that can exceed 0.6, appear in the tropical zone between 20°S and 20°N (Figure 5E) where Chl are not well reconstructed with the CNN_1 (Figure S4). Subtropical areas that already show high correlations with the CNN_1 model led to lower differences (yet show no degradation) in the correlation scores. The analysis is a bit more contrasted for the NRMSE metrics. NRMSE values are also improved by the CNN_{MM8} over most of the global ocean (in red in Figure 5F). However, the NRMSE is deteriorated (in blue) in several regions of the ocean, reaching values up to 0.5 around the Amazon River plume, and up to 0.3 at the mouths of the Congo and Kunene rivers off the coast of Angola, although correlations are improved when multi-modality is introduced. The use of a multi-mode CNN, whose learning is expected to be more regionally focused than a CNN_1, might increase the NRMSE in these regions, where Chl variability might rather reflect SPM and CDOM variability whose related predictors are missing.

To illustrate the ability of the CNN_{MM8} to better capture regional processes than CNN_1, the improvement in reconstructed Chl for specific regions when the CNN_1 is trained regionally vs. the CNN_1 and CNN_{MM8} trained at global scale is investigated. The CNN_1 trained regionally is expected to better learn regional processes than the CNN_1 trained over the global ocean (Fourrier et al., 2020). Table 3 shows the performance metrics obtained for two different BGCPs, a productive vs. an oligotrophic region: the Ninõ 3.4 region [5°N–5°S; 120°W–170°W] and the ultra-oligotrophic part of the South Pacific Subtropical Gyre (SPSG, [20°S–30°S; 95°W–145°W]). They present contrasting responses to the regional learning process. The Niño 3.4 region displays a significant potential for performance improvement as shown by the improvement between the globally and locally learnt CNN_1, which means that the relationships learnt at global scale are different than

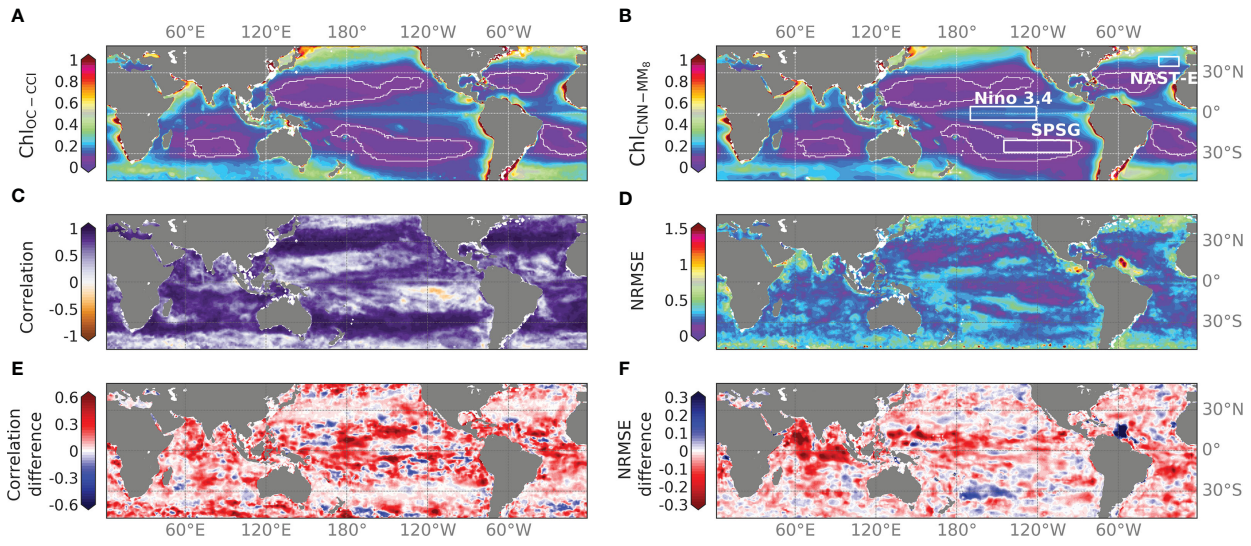


FIGURE 5
Time averaged **(A)** $Chl_{OC\text{-}CCI}$ and **(B)** CNN_{MM8} (in $mg.m^{-3}$) over [2012-2015]. Oligotrophic gyre boundaries are delimited by the 0.07 $mg.m^{-3}$ mean Chl isocontour superposed in white. **(C)** Correlation and **(D)** NRMSE of $Chl_{OC\text{-}CCI}$ vs. CNN_{MM8} over the same time-period. **(E)** Correlation and **(F)** NRMSE differences between CNN_{MM8} and Chl_{CNN1} over the same time period. NB: the colorbar is reversed for the NRMSE difference when compared to the correlation difference to highlight in red where the Chl reconstruction with CNN_{MM8} is improved.

TABLE 3 Performance metrics obtained between spatially averaged reconstructed Chl and $Chl_{OC\text{-}CCI}$ over two contrasted BGCPs: the Ninõ 3.4 region ([5°N–5°S, 120°W–170°W]) and the South Pacific Subtropical Gyre (SPSG, [20°S–30°S; 95°W–145°W]), when the CNN_1 is either trained at global scale or regionally, and the CNN_{MM8} is trained globally.

Model	Ninõ 3.4		SPSG	
	R^2	NRMSE	R^2	NRMSE
Globally learned CNN_1	0.28	0.17	0.85	**0.14**
Regionally learned CNN_1	0.48	0.12	**0.92**	0.17
Globally learned CNN_{MM8}	**0.68**	**0.11**	0.90	**0.14**

Best performances are highlighted in bold.

those learnt at regional scale. Here, the CNN_{MM8} reaches those performances and even outperforms the regional CNN_1, confirming the hypothesis of a better ability of the multi-mode CNN to reconstruct regional Chl. Contrastingly, in the SPSG region the reconstruction of Chl is already well performed by the globally and locally learnt CNN1 with R^2 = 0.85 vs. 0.92, respectively, leaving little room for improvement by the CNN_{MM8}. However, the CNN_{MM8} allows reduction of the NRMSE. Thus, in both regions, the CNN_{MM8} outperforms the regionally trained CNN_1 due to its ability to switch between different modes while it is less prone to overfitting.

Using the proposed EOF-based analysis, the ability of the CNN_{MM8} to retrieve the satellite-derived Chl spatio-temporal variability is investigated. The first EOF modes calculated on the seasonal and interannual $Chl_{OC\text{-}CCI}$ signal over [2012-2015] are presented in Figure 6 (upper and lower row, respectively). They respectively account for 33.2% and 13.2% of the total variance. Regarding the seasonal variability, the observed spatial patterns depict a clear contrast between the two hemispheres (Figure 6A), reflecting their opposite seasonal cycles. Consistently, the associated PC time-series depicts a sinusoidal signal with a one-year period (black line in Figure 6B). This seasonal variability is well reproduced by both the CNN_1 and CNN_{MM8} models, with correlations of their projected PCs with those of $Chl_{OC\text{-}CCI}$ of 0.99 and 1.00, respectively (Figure 6B). Even though the amplitude of the $Chl_{OC\text{-}CCI}$ PC was already very well captured by the monomode model, the multi-mode one still allows the correction of the slight underestimation that was observed otherwise.

Regarding the interannual variability, the first $Chl_{OC\text{-}CCI}$ EOF mode illustrates the strong spatio-temporal signature of ENSO events observed in the Pacific Ocean (Figure 6C), with opposite Chl responses to ENSO-related physical anomalies observed in the eastern Pacific compared to the western Pacific (Chavez et al., 1999). The temporal evolution of this first interannual $Chl_{OC\text{-}CCI}$ PC is highly related to the MEI (r=0.75, p<0.001) which reaches its maximum during the strong 2015/2016 El Niño event (Figure 6D). Here again, the interannual signal is well represented by CNN_1 and CNN_{MM8} with high correlation coefficients of their PCs with those of $Chl_{OC\text{-}CCI}$ (0.95 and 0.96, respectively), although the amplitudes are underestimated. These results stress the ability of the learning-based schemes to inform about the seasonal and interannual variability while it is not explicitly constrained during the training phase. Indeed, neither the training loss nor the architecture exploits time-related information. The underestimation of the interannual signal may be related to processes not considered, either related to the predictors (e.g. rivers inputs of nutrients, dust, land wildfire …) or to unresolved spatio-temporal scales. For instance, some discrepancies in the patterns of respective interannual EOF modes 1 can be observed in the Indian ocean and in the north Atlantic ocean (Figures S5B, D, F) where atmospheric dust inputs are most important (Jickells et al., 2005). Other sources of error can arise from differences between the training and the test periods chosen for this study. Beyond differences in the amplitude of ENSO events observed during those periods, different types of ENSO [Eastern Pacific (EP) versus Central Pacific (CP)] have also been reported. Thus, our training period [2003-2011] mainly hosts CP events, whereas the strong 2015/2016 El Niño event is usually classified as an EP event, with different processes and related impact on primary production (Radenac et al., 2012; Racault et al., 2017). Finally, delayed effects of climate modes have been very recently shown to influenced Chl in large parts of the ocean (see Figure 6 of Lim et al., 2022b), and especially in the eastern tropical Pacific one, whereas time-lags are not considered into our model.

3.3 Emergence of coherent spatio-temporal distribution of modes

The main advantage of the multi-mode CNN is the ability of its different sub-models to regionally specialize during the training phase. The training of the network benefits from all the sub-models that activate differently in various parts of the ocean. Maps of the percentage of variance explained by each mode of the CNN_{MM8} are computed over [2012-2015] (Figures 7A–H). This resulting regionalization, even if presenting some slight modifications of their spatial imprints, are quite consistent from one run to another. These percentages can regionally exceed 30% of the total variance for some modes in specific regions, such as in the three oligotrophic gyres of the southern hemisphere (mode 1), and, to a lesser extent, in those of the northern one (modes 2 and 3). These high variances which predominate for specific modes correspond to low values of entropy (the lower the entropy is, the more a specific mode dominates the signal: purple areas in Figure 7I). The percentages of variance of the remaining oceanic regions are distributed in a more balanced way between a larger number of modes (higher entropy, Figure 7I), but still present some regional variations.

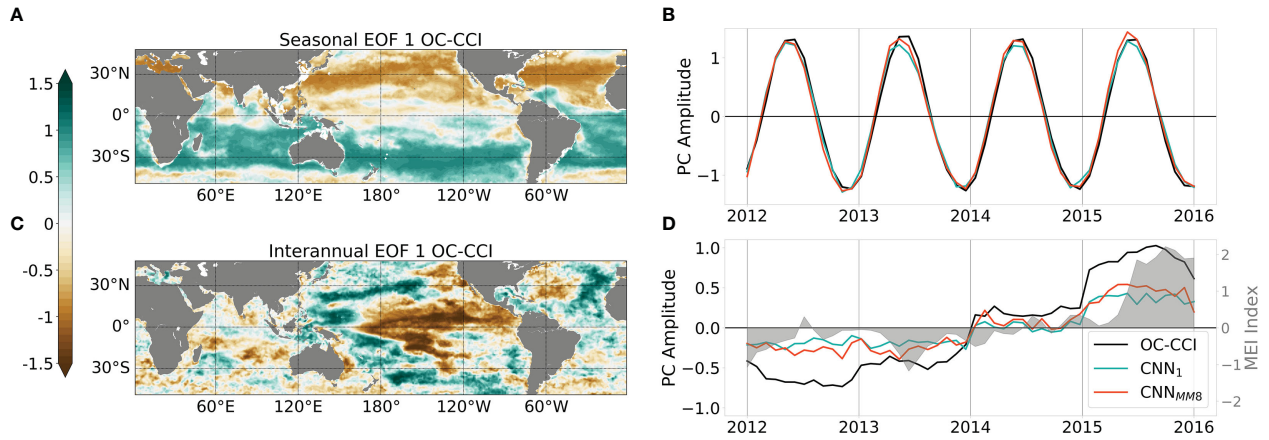


FIGURE 6
(A) Spatial pattern and **(B)** associated principal component (PC, as the black line) of the EOF first mode calculated on seasonal ChlOC-CCI over [2012-2015]. ChlCNN1 and ChlCNN-MM8 PCs obtained from the projection on ChlOC-CCI EOF spatial pattern are reported as the green and orange lines, respectively. **(C, D)** same as **(A, B)** but for the interannual signal. In **(D)**, the MEI is reported as the grey shaded area.

The eight sub-models thus depict some clear, coherent and non-random spatial patterns. Figure 7J synthetizes areas over which the different modes dominate, depicting for each pixel the mode that presents the maximum of explained variance. At first glance, there is a zonal spatial distribution of the modes in agreement with the original BGCPs distribution from Longhurst (1995). It partly results from latitudinal variations in physical forcing and leads to distinguishing what is called the "westerly winds domain" from the "trade wind domain" in the open ocean, whose seasonal changes in MLD are driven by different processes. The first one is reported to extend from the equator to ~30° of latitude, whereas the second one corresponds to mid-latitude areas. From the trained CNN_{MM8}, mode 7 mostly activates at higher latitudes than ~30°N/°S (in blue in Figure 7J), whereas mode 6 mainly activates at low-latitude. The first mode highly matches the three southern hemisphere oligotrophic gyres whereas the second and third modes coincide with the two gyres of the northern hemisphere. The spatial distribution of the three remaining modes (i.e., 4, 5 and 8) fits regions with specific oceanographic dynamics. Indeed, mode 4 (in red in Figure 7J) principally corresponds to areas of wind-induced coastal upwellings, as the Peru, Canary and Benguela areas and to a lesser extent to the California one, as well as to the Pacific and Atlantic equatorial upwelling. Mode 5 (in orange, Figure 7J) seems to stand for the mid-latitude highly dynamical parts of the ocean, that is to say the Gulf Stream and the Kuroshio currents. Finally, mode 8 (in yellow, Figure 7J) potentially highlights the Pacific frontal areas such as the Transition Zone Chlorophyll Front (Polovina et al., 2001) or the boundary between the equatorial Pacific high nutrient low chlorophyll (HNLC) area and the subtropical gyres.

Another feature of the multi-mode CNN is the ability of the sub-models to be variably activated over time. When considering two BGCPs with contrasting entropy such as the eastern North Atlantic Subtropical Gyre (NAST-E, [35°N–42°N; 15°W–30°W]) and the already mentioned SPSG (delimited on Figure 7J) it appears that the percentages of variance explained by each mode, spatially averaged over their respective areas, present contrasted temporal patterns (Figure 8). For instance, clear seasonal patterns emerge in the NAST-E (Figure 8A). The activation of the 7th mode (in light blue) occurs seasonally (with some inter-annual variability) with a maximum in November/December and a smaller secondary peak in April before starting to decrease. Then, the percentage of variance explained by modes 3 and 6 (in garnet and gray, respectively) increases in turn, followed by mode 2 (in green). The strong seasonal cycle observed here is consistent with the seasonal phytoplankton blooms reported in the North Atlantic. Contrastingly, in the SPSG one mode totally prevails over the others and does not display a clear seasonal cycle (purple line in Figure 8B). This strong dominance is also highlighted in Figures 7A, J and in the entropy map (Figure 7I). These results show that the learned modes vary in space but also in time and that they can be variably activated according to the variations of the physical predictors.

3.4 Predictors' relative importance in Chl reconstruction according to the modes

Using the perturbation-based method described in the last paragraph of **Section 2.5**, here we provide an insight in the relative importance of the physical predictors to reconstruct satellite-derived Chl. Histograms in Figure 9 show the normalized distribution along with the relative importance of each predictor in areas characterized by one dominant mode (i.e. the areas reaching the 90th percentile of variance, delimited by the green lines in Figures 7A–H), over the [2012-2015] test period. Those histograms illustrate that the different modes specialize by learning specific relationships between Chl and the physical predictors, implying possible different physical-biogeochemical interactions and dominant mechanisms. This is especially obvious with regards to the SST for which the eight histograms display various distributions. Those appear to be in general agreement with

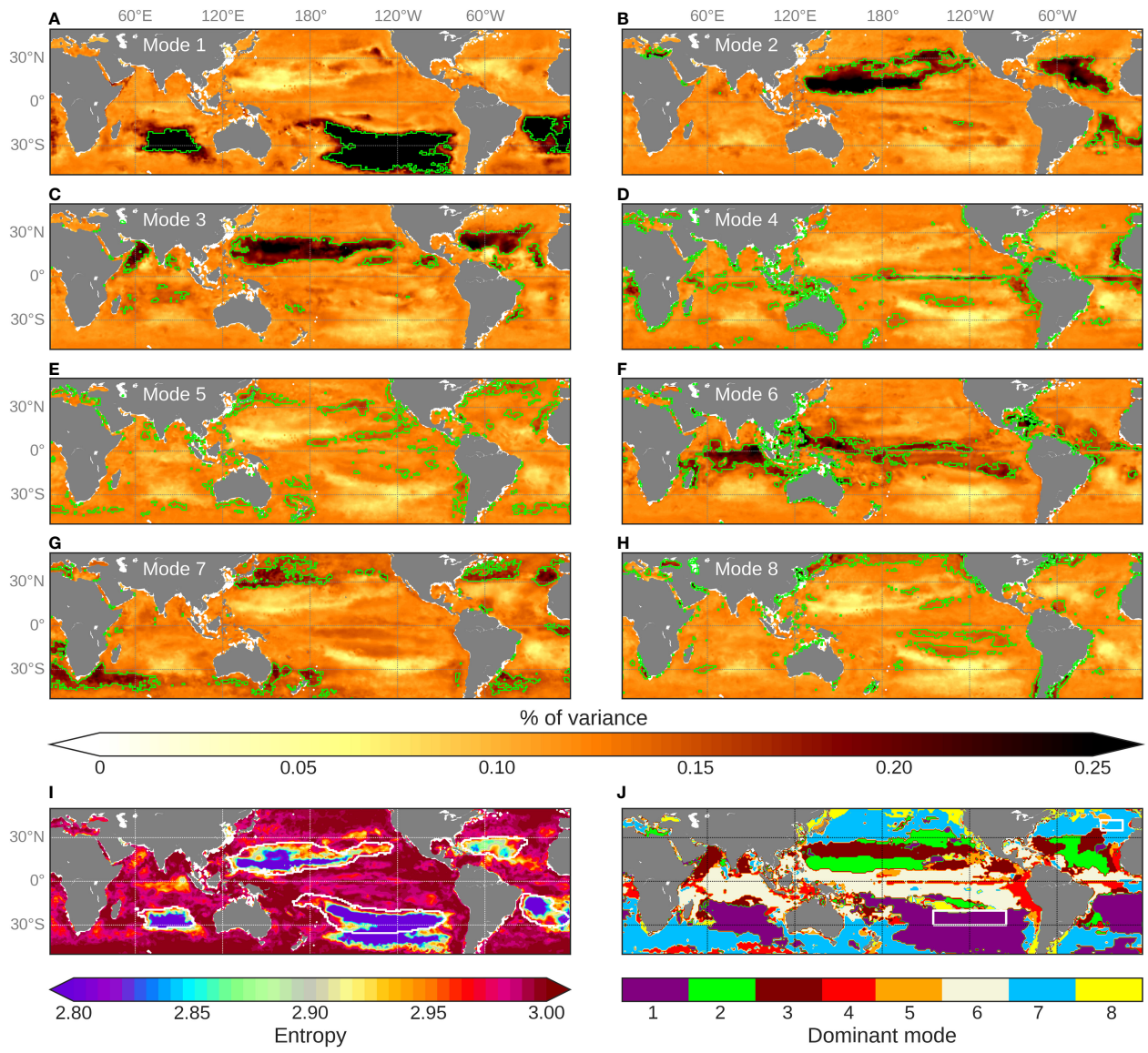


FIGURE 7
(A–H) Percentages of variance explained by each of the 8 modes of CNNMM8. Isolines of percentile-90 of the values are superposed in green. **(I)** Entropy characteristics computed from the above percentage of variance. Oligotrophic areas (with mean Chl < 0.07 mg.m-3 calculated over [2012-2015]) are delineated as white isocontours. **(J)** Spatial distribution of the modes explaining the highest percentage of variance.

known physical-biogeochemical processes, but also highlight some unexpected while plausible relationships.

As expected and already noted in a previous machine learning based study (Martinez et al., 2020b), SST has the strongest relative importance when compared to other physical predictors in all mode-associated regions (NB: the scale on the x-axis in Figure 9 differs for SST when compared to the other predictors). However, this relative importance is particularly striking in the subtropical gyres of the northern (modes 2 and 3) and southern (mode 1) hemispheres, and in some of the equatorial areas (mode 6) where most of the pixels reach a relative value higher than 0.5. Yet, a maximum/peak of occurrence around 0.7 is reached only for modes 2, 3 and 6. In those latter regions representing (i) the northern subtropical gyres and (ii) the equatorward boundaries of the northern and southern boundaries subtropical gyres, the strong dominance of SST as a predictor is consistent with the SST-Chl inverse relationship reported at global scale in literature (e.g., Behrenfeld et al., 2006; Martinez et al., 2009). Indeed, in the permanently stratified ocean which is nutrient-limited, SST variability is a proxy of vertical mixing variability and thus of the potential uplift of nutrients within the euphotic zone (Signorini et al., 2015).

Interestingly, this statement slightly differs for the southern oligotrophic gyres (mode 1) where the SST importance is weaker (but still dominant) than in the northern gyres (modes 2 & 3). On the contrary, other predictors such as SLA and surface currents (U, V) seem to have a greater relative importance in the southern gyres than in the northern ones. Counterintuitively, it suggests that some

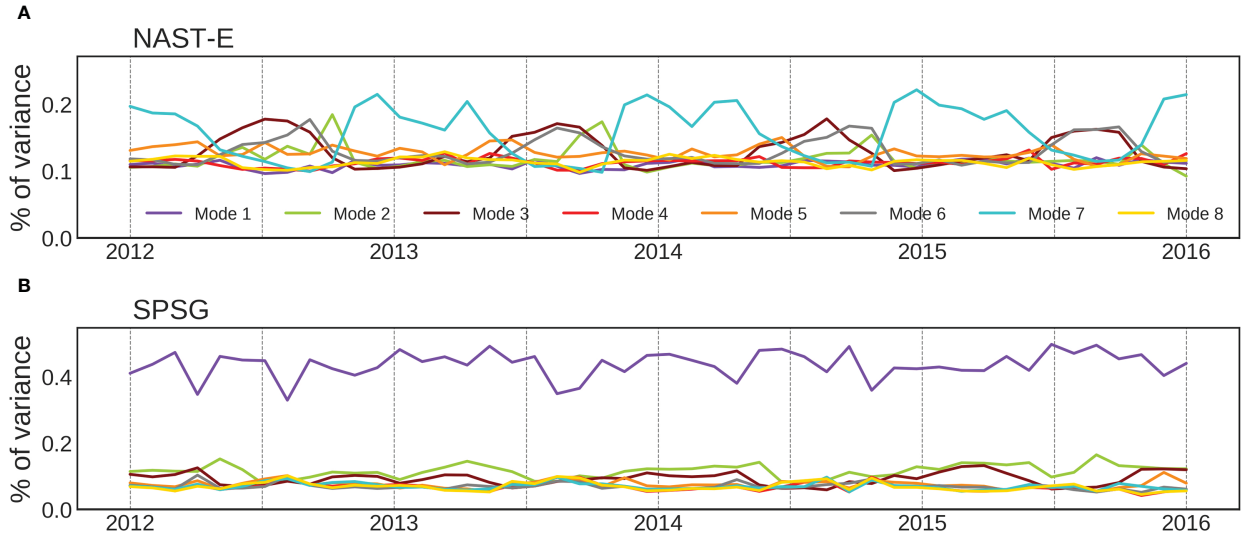


FIGURE 8
Temporal evolution of the percentages of variance explained by the 8 modes of the CNN_{MM8} in two BGCPs with contrasting entropy: **(A)** in the eastern North Atlantic Subtropical Gyre (NAST-E, [35°N–42°N; 15°W–30°W]) and **(B)** in the SPSG ([20°S–30°S; 95°W–145°W]). The colors correspond to the modes as in Figure 6J.

of the mechanisms that are at play in the oligotrophic gyres would be different between the two hemispheres. One hypothetic explanation may come from the possible stronger iron limitation in the southern hemisphere (Moore et al., 2001), resulting in a decoupling between the vertical inputs of macro-nutrients (e.g. NO3, PO4) and the phytoplankton local growth, thus minimizing the imprint of the SST-Chl inverse relationship characteristic of the northern hemisphere gyres. Considering the lack of vertical inputs of the limiting nutrients, lateral transport of tracers (nutrients and phytoplankton) near transition zones surrounding the gyres may thus be of greater relative importance in the southern hemisphere, which is consistent with the greater importance of the SLA and currents. The double peak SST distribution in mode 1 could then be interpreted as one characteristic of the gyres [related to the vertical input of nutrients and common to the southern and northern gyres (Signorini et al., 2015)] and one related to the lateral transport of tracers mostly at play in the southern gyres. Consistently, for both mode 1 and mode 3, the smaller relative importance of SST occurs where the relative importance of surface currents (U, V) is larger (not shown), the first peak of low SST importance observed for mode 3 corresponding to the Arabian Sea area.

For the other modes (4, 5, 7 and 8) which correspond to more productive regions (e.g. mode 4 highlights equatorial and coastal upwelling regions) SST, while still dominating, appears to be of weaker relative importance than in the gyres. In addition, the other predictors' relative weights are more uniformly distributed. This suggests that a significant part of the Chl variability is not explained by processes affecting the SST but is rather related to a complex interaction of processes whose signatures are embedded in other predictors (e.g. lateral currents, light, winds).

Overall, Figure 9 shows that the multi-mode CNN approach may give mechanistic insights on the functioning of specific ocean provinces. Hypothesis drawn from such analysis may therefore be further tested using for example mechanistic numerical models.

4 Discussion and perspectives

CNNs have been widely used these last years in geosciences to leverage the spatial dimension of datasets, and their ability to capture spatial patterns have been largely demonstrated (Makantasis et al., 2015; Shen, 2018; Brodrick et al., 2019; Kattenborn et al., 2021). Consistently, our results confirm their relevance to reconstruct phytoplankton spatio-temporal distribution from physical predictors without using any explicit geographical predictors (i.e., latitude and longitude). Compared to previous studies based on CNNs, our study goes further and provides a more efficient method to manage spatio-temporal heterogeneities. This is one of the main limitations of CNNs, and deep learning models in general, when dealing with environmental variables (Bai et al. 2016; Reichstein et al., 2019; Yuan et al., 2020), especially in a highly dynamical environment such as the ocean. Introducing a multi-mode CNN, we showed that the network can identify different areas over which the learning phase can be regionally optimized. The sub-models have been shown to specialize on physically-consistent regions and thus to better capture regional processes. Such kind of neural network architectures have been previously proposed to tackle the merging of data from various sensors and/or spatio-temporal resolutions (Martínez and Yannakakis, 2014; Yang et al., 2016; Melotti et al., 2018; Ienco et al., 2019; Joze et al., 2020; Zhang et al., 2022). However, to our knowledge, no studies have been done to apply them to regionalization issues. This approach focusing on regional mechanisms converges to recent deep learning architecture built on

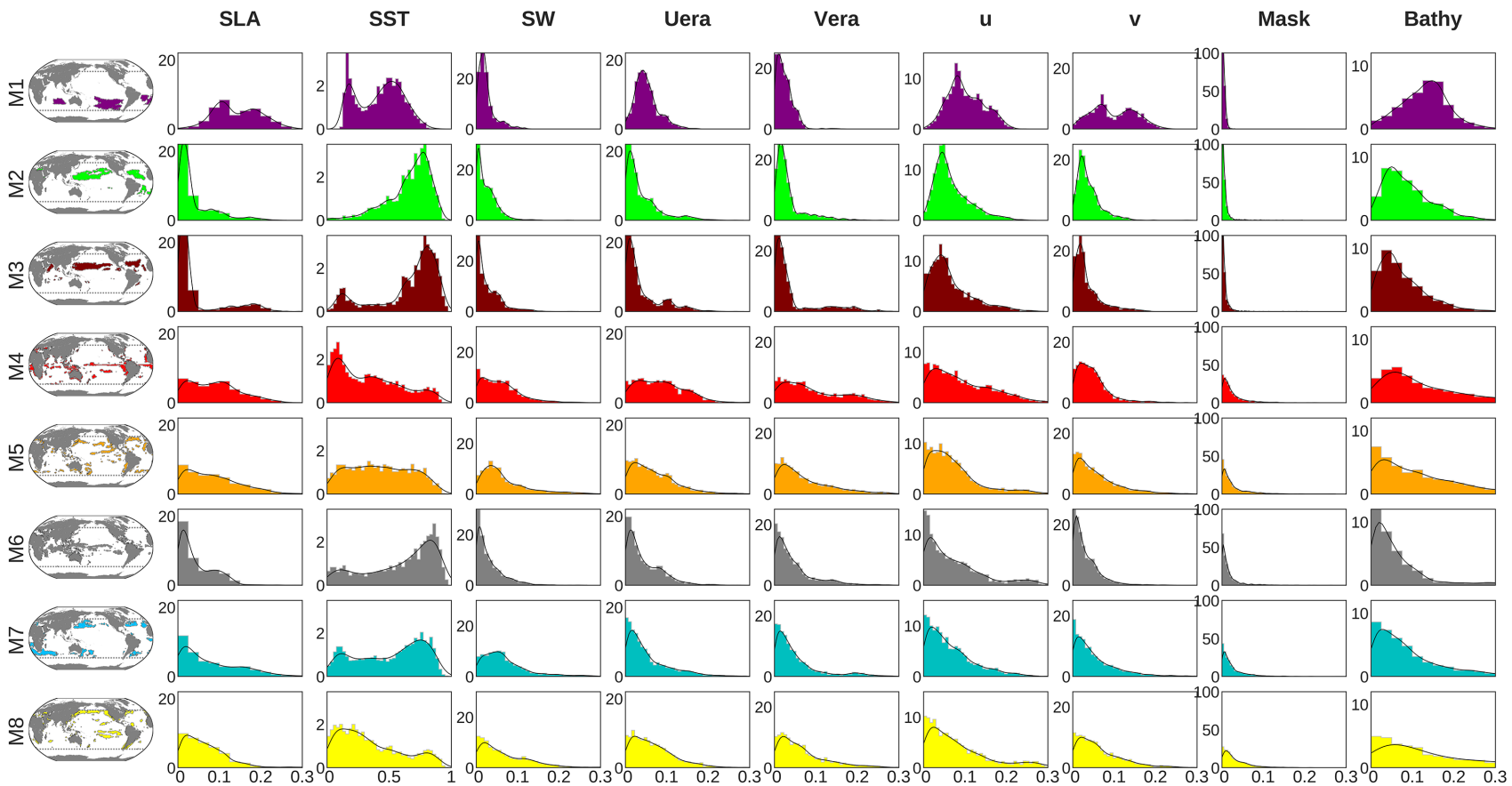


FIGURE 9
Normalized distribution (y-axis) of the relative importance of each predictor (x-axis) computed over the percentile-90 area for each mode. NB: the scale on the x-axis is homogeneous across all the variables but SST.

10.3389/fmars.2023.1077623

self-attention processes such as Transformers (Dosovitskiy et al., 2020), but are cheaper in terms of computing cost.

The multi-mode CNN not only allows a better simulation of the chlorophyll concentration spatio-temporal variations, but it also improves the network interpretability, which is of particular interest in the Earth science field as it can allow one to find new unexpected relationships among data. Some *post-hoc* explanation methods [i.e., once the model is already trained (Fan et al., 2021; Xie et al., 2020)] specifically designed for CNNs, and which are still heavily under-exploited in Earth sciences, may have been considered to go further in the network interpretability (Ras et al., 2022). For instance, Ham et al. (2019) computed heatmaps or Class Activation Maps (CAM, Zhou et al., 2016) from a CNN to analyze which parts of the global ocean contribute the most to the prediction of El Niño events. Zeiler and Fergus (2014) also proposed a way to access and visualize how much information from input data is processed according to the different network layers of CNNs. However, these latter methods do not allow one to optimize the regional learning nor to provide some interpretability from the model outputs, which are both specificities of our multi-mode CNN. Optimizing, during training, specifically designed explanations is what the so-called intrinsic methods aim to do. This is one of the advantages of the proposed multi-mode CNN compared to the mono-mode.

Here, we took advantage of both the intrinsic explainable methods and *post-hoc* diagnostics to increase the interpretability. Indeed, in the present study, the intrinsic multi-modality shows some consistency in the learning of the eight modes with the spatio-temporal variations of the ocean dynamics, which is somehow expected to be reflected in the variations of the phytoplankton biomass. Applying a basic *post-hoc* perturbation-based method to the CNN_{MM8} allowed us investigating the relative importance of the predictors (as illustrated in Figure 9). Other *post-hoc* methods shedding light on features that drive the model's decision would deserve to be investigated and compared with one another. For instance, the Shapley Additive exPlanations (SHAP) method (Lundberg and Lee, 2017), which can be applied to any kind of neural networks, measures the effects of an input perturbation on the network's output to retrieve the relative importance of each predictors (Padarian et al., 2020; Betancourt et al., 2022; Pauthenet et al., 2022). This method would allow consideration of interdependencies between variables, whereas removing them one by one, as did in our study, may not be optimum.

Multi-mode CNN results are promising even if some strategies could further improve the performance of the Chl reconstruction. From the architecture point of view, the addition of pooling layers, especially for the W attention module (see Figure 1), may allow a better consideration of large-scale spatial structures and thus of the regionalization of the different modes. Coupling the CNN sub-models with Recurrent Neural Networks (RNNs) should help accounting for temporal dynamics/time history with a more sophisticated way than if adding time-lags as predictors within our current architecture. Indeed, while instantaneous environmental fluctuations are thought to explain much of the observed phytoplankton temporal variability, time-lag responses of weeks to a few months would also be expected (Ji et al., 2010; Feng et al., 2015; Schollaert Uz et al., 2017; Lim et al., 2022b). This would arise from biological processes mainly, such as dormancy and reproduction (Ji et al., 2010), or ecological interactions as species competition or grazing pressure (Feng et al., 2015). Assessing their own impact on the Chl reconstruction using models that take into account temporal dependencies is certainly worth doing and would deserve a dedicated study but was beyond the scope of this paper. Moreover, while the current network learns different sub-models on specific areas (spatial attention), sub-models could also learn according to different temporal periods (temporal attention). In addition, the current architecture could be easily adapted to learn from predictors with different higher spatio-temporal resolutions. This may improve the Chl reconstruction performance by considering processes currently not resolved with the actual monthly dataset averaged on a 1° grid, such as the mesoscale ocean dynamics or high frequency wind events. This would also enable a better assessment of the impact of finer scale dynamics on Chl low-frequency variability at global scale.

New predictors should also be considered in upcoming studies to stand for a wider range of processes, as mentioned in **Section 3.2**. Aerosol Optical Depth (AOD) observations could help to account for the sporadic supply of nutrients into the ocean from atmospheric deposition, such as dust-derived iron that can play a significant role on interannual phytoplankton dynamics in some regions (Letelier et al., 2019; Lim et al., 2022a; Meng et al., 2022). Precipitations could help to distinguish wet from dry dust deposition, known to present different iron solubility (Fan et al., 2006), a proxy of iron bioavailability (Schulz et al., 2012). *In situ* water column data provided from Argo floats could also be considered to better represent the MLD variability rather than using surface proxy only. However, one drawback to fix is that it would reduce the length of available time series of more than 10 years (a sufficient data coverage is not expected before the 2010's). Chl is a proxy of phytoplankton biomass, and other underlying processes can be reflected on Chl changes. For instance, in response to changes in light conditions, phytoplankton cells can adjust their intracellular Chl so that Chl changes may be rather related to photoadaptation than to biomass. Thus, reconstructing the ratio between Chl and particulate backscattering coefficient [bbp, related to the size particles, Loisel et al., 2002] would deserve to be investigated in future studies. Here we used SW as a proxy of PAR, whereas SW spatiotemporal changes may not always reflect PAR variability due to strong absorption by water vapor, ozone, and clouds outside the PAR spectral range (Chou and Suarez, 1999). Yet, not considering the spectral form of incident radiation can lead to large errors in modeling oceanic primary production (Sathyendranath et al., 1989; Frouin et al., 2018). Sensitivity tests concerning the use of SW as a proxy of PAR should be carried out in the future, for example by comparing the results obtained here with those obtained using PAR products from radiometric observations or reanalysis data (e.g., MERRA-2). Finally, considering uncertainties of the different products used (either for the physical predictors or for the reference satellite-derived

Chl) would deserve to be investigated. When available, using pixel-by-pixel estimates of uncertainties as inputs of the network may, for example, allow to give less importance on the learned relationships between predictors and Chl where data quality is lower. In addition, using metrics of performance that include ocean color uncertainties would be useful to distinguish errors that arise in our reconstructions due to such uncertainties from those due to our network architecture and/or used predictive input data.

Here, we have focused on comparing the ability of different deep learning schemes to simulate phytoplankton variability at seasonal and interannual timescales, and have shown that the proposed approach outperforms previous machine learning models introduced in the literature to achieve this task [namely the MLP, and indirectly the less-performant SVR approach (Martinez et al., 2020b)]. In (Martinez et al., 2020a), the SVR approach was quantitatively compared to a coupled physical-biogeochemical ocean model simulation (NEMO-DFS5.2-PISCES) and was found to better reproduce patterns of satellite-derived Chl trends (but less well captures their amplitudes) as well as its interannual variability. This suggests that the proposed multi-mode CNN would, by extent, also better reproduce some aspects of Chl long-term variabilities than biogeochemical models, appearing as a complementary tool to retrieve past Chl variability. Further work is expected to investigate this point, especially regarding multi-decadal changes in global phytoplankton that were pointed out between the CZCS and SeaWiFS era (Martinez et al., 2009) using historical consistent ocean color dataset built by Antoine et al. (2005). Here, we also suggest that data-driven approaches can be complementary to classical models' studies to explore mechanisms driving phytoplankton variabilities at large scales. For sure, coupled physical-biogeochemical models, that are built upon explicit formulation of processes governing phytoplankton distribution, undoubtedly remain the most robust and straightforward way to test impacts on primary production of processes that are well understood and well parameterized. However, some unknown processes would be missing, or others would be roughly parameterized so that their impact on primary production would be hard to assess without bias. As an example, large variations of parameterized iron solubility in dust are reported among global ocean biogeochemistry models, and the fixed values used globally doesn't allow reproducing all the regionally and temporally variability of oceanic dust-derived dissolved iron (Tagliabue et al., 2016). On the contrary, using dust deposition flux as inputs predictors, data-driven methods could give new regards and further clues on their regional impact on phytoplankton biomass without having to explicitly parameterize bio-physical values such as solubility. As another example, deriving information about factors driving the phytoplankton ecosystem structure could be achieved using the proposed multi-mode approach. This could be done by learning and reconstructing the phytoplankton community structure [using for example PHYSAT data (Alvain et al., 2008)] with the actual set of predictors and assessing how their relative importance vary in time and space. Such information is of great importance as phytoplankton taxonomic and size composition strongly determines carbon fluxes (Boyd and Newton, 1995; Guidi et al., 2009).

5 Conclusion

In this study, a new deep learning architecture was proposed to reconstruct surface Chl from oceanic and atmospheric physical predictors in the global ocean. Spatial attention mechanisms (i.e. multi modes) were introduced into a CNN to regionally learn relationships in a preferential way according to the modes. Its performance was evaluated over a fully-independent time period hosting the strong 2015/2016 El Niño event. Both mono and multi-mode CNNs outperformed the previous state-of-the-art MLP schemes to reconstruct spatial and temporal satellite-derived Chl distribution while being computationally more efficient. One other main interest of CNNs is their ability to not need explicit geographical information as predictors (e.g. longitude and latitude) leading to the opportunity to seize BGCPs boundaries evolutions according to climate oscillations. In addition, the multi-mode CNN_{MM8} allowed us to better capture some regional processes than CNN_1 thanks to its modes that can regionally learn specific phytoplankton responses to the physical forcing. The multi-mode CNN_{MM8} also provided insights into where and when the modes preferentially activate, improving the interpretability of the network. Those activations appeared to be in general agreement with known physical-biogeochemical interactions at global scale. However, they also allowed us to highlight an unexpected difference in the mechanisms at play between the oligotrophic gyres of both hemispheres. Overall, while some biases remain between the reconstructed Chl fields and satellite observations, the proposed multi-mode model is greatly valuable as it offers an interesting perspective to reconstruct phytoplankton biomass over a long time-period and new ways to explore the physical mechanisms at play.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://www.seanoe.org/data/00807/91910/ and https://github.com/JoanaR/multi-mode-CNN-pytorch.

Author contributions

JR led the study, implemented the neural networks, processed the data and wrote the first draft of the manuscript; JR, RF, EM, TG, LD, and JL contributed to the conception of the research and results analysis; EM, RF, TG, and LD contributed to the improvement of the manuscript. JL contributed to run the regionalized mono-mode CNN and to compute the predictors' relative importance. EM: funding acquisitions. All authors contributed to the article and approved the submitted version.

Funding

The J. Roussillon PhD grant was funded by the ISblue project, Interdisciplinary graduate school for the blue planet (ANR-17-EURE-0015) and co-funded by a grant from the French government under the program "Investissements d'Avenir. We also thank the French National Program of Spatial Remote Sensing (INSU PNTS) for co-funding this work. The Master 2 internship of J. Littaye was funded by the French National Research institute for sustainable Development (IRD).

Acknowledgments

We would like to thank Anwar Brini for providing his Python program to compute the heat diffusion equation. We also warmly thank two reviewers for their very helpful comments on the manuscript. We also acknowledge the NOAA PSL for providing the NCEP-NCAR Reanalysis 1 data. Thanks to NOAA NCEP EMC CMB GLOBAL Reyn-SmithOIv2 for providing monthly SST data. The authors also acknowledge ECMWF for providing the ERA-Interim reanalysis data set. GEBCO is acknowledged for proving bathymetry data. NASA Physical Oceanography data center for providing OSCAR Data. We acknowledge the project team of Ocean Colour Climate Change Initiative for generating and sharing the merged datasets on chlorophyll-a concentrations.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1077623/full#supplementary-material

References

Ai, B., Wen, Z., Wang, Z., Wang, R., Su, D., Li, C., et al. (2020). "Convolutional neural network to retrieve water depth in marine shallow water area from remote sensing images," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 13. 2888–2898.

Alvain, S., Moulin, C., Dandonneau, Y., and Loisel, H. (2008). Seasonal distribution and succession of dominant phytoplankton groups in the global ocean: A satellite view. *Global Biogeochemical Cycles* 22 (3). doi: 10.1029/2007GB003154

Anthony, L. F. W., Kanding, B., and Selvan, R. (2020). Carbontracker: Tracking and predicting the carbon footprint of training deep learning models. *arXiv preprint arXiv:2007.03051*. doi: 10.48550/arXiv.2007.03051

Antoine, D., Morel, A., Gordon, H. R., Banzon, V. F., and Evans, R. H. (2005). Bridging ocean color observations of the 1980s and 2000s in search of long-term trends. *J. Geophysical Research: Oceans* 110 (C6). doi: 10.1029/2004JC002620

Anzai, Y., and Shimada, T. (1988). "Modular neural networks for shape and/or location recognition," in *Int Neural Network Soc First Ann Meeting*, Boston, USA.

Aubert, G., Kornprobst, P., and Aubert, G. (2006). *Mathematical problems in image processing: partial differential equations and the calculus of variations* Vol. 147 (New York: Springer), 26.

Auda, G., and Kamel, M. (1999). Modular neural networks: a survey. *Int. J. Neural Syst.* 9 (02), 129–151. doi: 10.1142/S0129065799000125

Azam, F. (2000). *Biologically inspired modular neural networks* (Virginia Polytechnic Institute and State University).

Bai, Y., Wu, L., Qin, K., Zhang, Y., Shen, Y., and Zhou, Y. (2016). A geographically and temporally weighted regression model for ground-level PM2. 5 estimation from satellite-derived 500 m resolution AOD. *Remote Sens.* 8 (3), 262. doi: 10.3390/rs8030262

Beaugrand, G., Reid, P. C., Ibanez, F., Lindley, J. A., and Edwards, M. (2002). Reorganization of north Atlantic marine copepod biodiversity and climate. *Science* 296 (5573), 1692–1694. doi: 10.1126/science.1071329

Behrenfeld, M. J., O'Malley, R. T., Siegel, D. A., McClain, C. R., Sarmiento, J. L., Feldman, G. C., et al. (2006). Climate-driven trends in contemporary ocean productivity. *Nature* 444 (7120), 752–755. doi: 10.1038/nature05317

Bergen, K. J., Johnson, P. A., Maarten, V., and Beroza, G. C. (2019). Machine learning for data-driven discovery in solid earth geoscience. *Science* 363 (6433). doi: 10.1126/science.aau0323

Betancourt, C., Stomberg, T. T., Edrich, A. K., Patnala, A., Schultz, M. G., Roscher, R., et al. (2022). Global, high-resolution mapping of tropospheric ozone–explainable machine learning and impact of uncertainties. *Geoscientific Model. Dev. Discussions* 15 (11), 4331–4354. doi: 10.5194/gmd-15-4331-2022

Bisson, K. M., Boss, E., Werdell, P. J., Ibrahim, A., Frouin, R., and Behrenfeld, M. J. (2021). Seasonal bias in global ocean color observations. *Appl. optics* 60 (23), 6978–6988. doi: 10.1364/AO.426137

Boyd, P., and Newton, P. (1995). Evidence of the potential influence of planktonic community structure on the interannual variability of particulate organic carbon flux. *Deep Sea Res. Part I: Oceanographic Res. Papers* 42 (5), 619–639. doi: 10.1016/0967-0637(95)00017-Z

Boyd, P. W., Strzepek, R., Fu, F., and Hutchins, D. A. (2010). Environmental control of open-ocean phytoplankton groups: Now and in the future. *Limnology oceanography* 55 (3), 1353–1376. doi: 10.4319/lo.2010.55.3.1353

Brodrick, P. G., Davies, A. B., and Asner, G. P. (2019). Uncovering ecological patterns with convolutional neural networks. *Trends Ecol. Evol.* 34 (8), 734–745. doi: 10.1016/j.tree.2019.03.006

Bryan, K. (2016). A review of observations of the effect of bathymetry on ocean circulation in recent decades. *Izv. Atmos. Ocean. Phys.* 52, 341–347. doi: 10.1134/S0001433816040034

Cachay, S. R., Erickson, E., Bucker, A. F. C., Pokropek, E., Potosnak, W., Osei, S., et al. (2020). Graph neural networks for improved El Nino forecasting. *arXiv preprint arXiv:2012.01598*. doi: 10.48550/arXiv.2012.01598

Chattopadhyay, A., Hassanzadeh, P., and Pasha, S. (2020). Predicting clustered weather patterns: A test case for applications of convolutional neural networks to spatio-temporal climate data. *Sci. Rep.* 10 (1), 1–13. doi: 10.1038/s41598-020-57897-9

Chavez, F. P., Strutton, P. G., Friederich, G. E., Feely, R. A., Feldman, G. C., Foley, D. G., et al. (1999). Biological and chemical response of the equatorial Pacific Ocean to the 1997-98 El Niño. *Science* 286 (5447), 2126–2131. doi: 10.1126/science.286.5447.2126

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., et al. (2017). "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5659–5667.

Chou, M. D., and Suarez, M. J. (1999). A Solar Radiation Parameterization for Atmospheric Studies. In: *Suarez (Hrsg.) Technical Report Series on Global Modeling and Data Assimilation NASA/TM-1999-104606*, Vol. 15 (Greenbelt: NASA Goddard Space Flight Center). 38 pp.

Cooke, C. L., and Scott, K. A. (2019). "Estimating sea ice concentration from SAR: Training convolutional neural networks with passive microwave data," in *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 57. 4735–4747.

d'Ortenzio, F., Antoine, D., Martinez, E., and Ribera d'Alcalà, M. (2012). Phenological changes of oceanic phytoplankton in the 1980s and 2000s as revealed by remotely sensed ocean-color observations. *Global Biogeochemical Cycles* 26 (4). doi: 10.1029/2011GB004269

Devred, E., Sathyendranath, S., and Platt, T. (2009). Decadal changes in ecological provinces of the Northwest Atlantic ocean revealed by satellite observations. *Geophysical Res. Lett.* 36 (19). doi: 10.1029/2009GL039896

Dohan, K. (2017). Ocean surface currents from satellite data. *J. Geophysical Research: Oceans* 122 (4), 2647–2651. doi: 10.1002/2017JC012961

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. doi: 10.48550/arXiv.2010.11929

ESR (2009). *OSCAR third degree resolution ocean surface currents. Ver. 1*. PO DAAC, CA, USA. doi: 10.5067/OSCAR-03D01

Falkowski, P. G., Barber, R. T., and Smetacek, V. (1998). Biogeochemical controls and feedbacks on ocean primary production. *science* 281 (5374), 200–206. doi: 10.1126/science.281.5374.200

Fan, S. M., Moxim, W. J., and Levy, H. (2006). Aeolian input of bioavailable iron to the ocean. *Geophysical Res. Lett.* 33 (7). doi: 10.1029/2005GL024852

Fan, F. L., Xiong, J., Li, M., and Wang, G. (2021). On interpretability of artificial neural networks: A survey. *IEEE Transactions on Radiation and Plasma Medical Sciences* 5 (6), 741–760. doi: 10.48550/arXiv.2001.02522

FAO (2020). *The state of world fisheries and aquaculture 2020. sustainability in action* (Rome). doi: 10.4060/ca9229en

Feng, J., Durant, J. M., Stige, L. C., Hessen, D. O., Hjermann, D.Ø., Zhu, L., et al. (2015). Contrasting correlation patterns between environmental factors and chlorophyll levels in the global ocean. *Global Biogeochemical Cycles* 29 (12), 2095–2107. doi: 10.1002/2015GB005216

Fourrier, M., Coppola, L., Claustre, H., D'Ortenzio, F., Sauzède, R., and Gattuso, J. P. (2020). A regional neural network approach to estimate water-column nutrient concentrations and carbonate system variables in the Mediterranean Sea: CANYON-MED. *Front. Mar. Sci.* 7, 620. doi: 10.3389/fmars.2020.00620

Frenger, I., Münnich, M., and Gruber, N. (2018). Imprint of southern ocean mesoscale eddies on chlorophyll. *Biogeosciences* 15, 4781–4798. doi: 10.5194/bg-15-4781-2018

Frouin, R., Ramon, D., Boss, E., Jolivet, D., Compiègne, M., Tan, J., et al. (2018). Satellite radiation products for ocean biology and biogeochemistry: needs, state-of-the-art, gaps, development priorities, and opportunities. *Front. Mar. Sci.* 5, 3. doi: 10.3389/fmars.2018.00003

Gille, S. T., Metzger, E. J., and Tokmakian, R. (2004). Seafloor topography and ocean circulation. *NAVAL Res. Lab. STENNIS SPACE CENTER MS OCEANOGRAPHY DIV*. doi: 10.5670/oceanog.2004.66

Goodfellow, I., Bengio, Y., and Courville, A. (2016). "Convolutional networks," in *Deep learning* (Cambridge, MA, USA: MIT Press), 330–372.

Gregg, W. W., and Casey, N. W. (2007). Sampling biases in MODIS and SeaWiFS ocean chlorophyll data. *Remote Sens. Environ.* 111 (1), 25–35. doi: 10.1016/j.rse.2007.03.008

Gregg, W. W., Rousseaux, C. S., and Franz, B. A. (2017). Global trends in ocean phytoplankton: a new assessment using revised ocean colour data. *Remote Sens. Lett.* 8 (12), 1102–1111. doi: 10.1080/2150704X.2017.1354263

Grimaud, G. M., Mairet, F., Sciandra, A., and Bernard, O. (2017). Modeling the temperature effect on the specific growth rate of phytoplankton: a review. *Rev. Environ. Sci. Bio/Technology* 16 (4), 625–645. doi: 10.1007/s11157-017-9443-0

Guidi, L., Stemmann, L., Jackson, G. A., Ibanez, F., Claustre, H., Legendre, L., et al. (2009). Effects of phytoplankton community on production, size, and export of large aggregates: A world-ocean analysis. *Limnology Oceanography* 54 (6), 1951–1963. doi: 10.4319/lo.2009.54.6.1951

Guieu, C., Al Azhar, M., Aumont, O., Mahowald, N., Levy, M., Éthé, C., et al. (2019). Major impact of dust deposition on the productivity of the Arabian Sea. *Geophysical Res. Lett.* 46 (12), 6736–6744. doi: 10.1029/2019GL082770

Haidar, A., and Verma, B. (2018). "Monthly rainfall forecasting using one-dimensional deep convolutional neural network," in *IEEE Access*, Vol. 6. 69053–69063.

Ham, Y. G., Kim, J. H., and Luo, J. J. (2019). Deep learning for multi-year ENSO forecasts. *Nature* 573 (7775), 568–572. doi: 10.1038/s41586-019-1559-7

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Identity mappings in deep residual networks," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. 630–645 (Springer International Publishing). doi: 10.1007/978-3-319-46493-0_38

Henson, S. A., Beaulieu, C., and Lampitt, R. (2016). Observing climate change trends in ocean biogeochemistry: when and where. *Global Change Biol.* 22 (4), 1561–1571. doi: 10.1111/gcb.13152

Henson, S. A., Dunne, J. P., and Sarmiento, J. L. (2009a). Decadal variability in north Atlantic phytoplankton blooms. *J. Geophysical Research: Oceans* 114 (C4). doi: 10.1029/2008JC005139

Henson, S. A., Raitsos, D., Dunne, J. P., and McQuatters-Gollop, A. (2009b). Decadal variability in biogeochemical models: Comparison with a 50-year ocean colour dataset. *Geophysical Res. Lett.* 36 (21). doi: 10.1029/2009GL040874

Henson, S. A., Sarmiento, J. L., Dunne, J. P., Bopp, L., Lima, I., Doney, S. C., et al. (2010). Detection of anthropogenic climate change in satellite records of ocean chlorophyll and productivity. *Biogeosciences* 7 (2), 621–640. doi: 10.5194/bg-7-621-2010

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks* 2 (5), 359–366. doi: 10.1016/0893-6080(89)90020-8

Ienco, D., Interdonato, R., Gaetano, R., and Minh, D. H. T. (2019). Combining sentinel-1 and sentinel-2 satellite image time series for land cover mapping *via* a multi-source deep learning architecture. *ISPRS J. Photogrammetry Remote Sens.* 158, 11–22. doi: 10.1016/j.isprsjprs.2019.09.016

IOCCG (2019). *Uncertainties in ocean colour remote sensing*. Ed. F. Mélin (Dartmouth, Canada: International Ocean Colour Coordinating Group). doi: 10.25607/OBP-696

Irwin, A. J., and Oliver, M. J. (2009). Are ocean deserts getting larger? *Geophysical Res. Lett.* 36 (18). doi: 10.1029/2009GL039883

Jeon, W., Kim, J. S., and Seo, K. W. (2021). Reconstruction of terrestrial water storage of GRACE/GFO using convolutional neural network and climate data. *J. Korean Earth Sci. Soc.* 42 (4), 445–458. doi: 10.5467/JKESS.2021.42.4.445

Jetley, S., Lord, N. A., Lee, N., and Torr, P. H. (2018). Learn to pay attention. *arXiv preprint arXiv:1804.02391*. doi: 10.48550/arXiv.1804.02391

Ji, R., Edwards, M., Mackas, D. L., Runge, J. A., and Thomas, A. C. (2010). Marine plankton phenology and life history in a changing climate: current research and future directions. *J. plankton Res.* 32 (10), 1355–1368. doi: 10.1093/plankt/fbq062

Jickells, T. D., An, Z. S., Andersen, K. K., Baker, A. R., Bergametti, G., Brooks, N., et al. (2005). Global iron connections between desert dust, ocean biogeochemistry, and climate. *science* 308 (5718), 67–71. doi: 10.1126/science.1105959

Jönsson, B. F., Salisbury, J., Atwood, E. C., Sathyendranath, S., and Mahadevan, A. (2023). Dominant timescales of variability in global satellite chlorophyll and SST revealed with a MOving standard deviation saturation (MOSS) approach. *Remote Sens. Environ.* 286, 113404. doi: 10.1016/j.rse.2022.113404

Joze, H. R. V., Shaban, A., Iuzzolino, M. L., and Koishida, K. (2020). "MMTM: Multimodal transfer module for CNN fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13289–13299.

Kahru, M., Gille, S. T., Murtugudde, R., Strutton, P. G., Manzano-Sarabia, M., Wang, H., et al. (2010). Global correlations between winds and ocean chlorophyll. *J. Geophysical Research: Oceans* 115 (C12). doi: 10.1029/2010JC006500

Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., et al. (1996). The NCEP/NCAR 40-year reanalysis project. *Bull. Am. meteorological Soc.* 77 (3), 437–472. doi: 10.1175/1520-0477(1996)077<0437:TNYRP>2.0.CO;2

Kattenborn, T., Leitloff, J., Schiefer, F., and Hinz, S. (2021). Review on convolutional neural networks (CNN) in vegetation remote sensing. *ISPRS J. Photogrammetry Remote Sens.* 173, 24–49. doi: 10.1016/j.isprsjprs.2020.12.010

Kim, Y. J., Kim, H. C., Han, D., Lee, S., and Im, J. (2020). Prediction of monthly Arctic sea ice concentrations using satellite and reanalysis data based on convolutional neural networks. *Cryosphere* 14 (3), 1083–1104. doi: 10.5194/tc-14-1083-2020

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv*, 1412.6980. doi: 10.48550/arXiv.1412.6980

Kirsch, L., Kunze, J., and Barber, D. (2018). Modular networks: Learning to decompose neural computation. *Adv. Neural Inf. Process. Syst.* 31. doi: 10.48550/arXiv.2001.02522

Letelier, R. M., Björkman, K. M., Church, M. J., Hamilton, D.S., Mahowald, N. M., Scanza, R.A., et al. (2019). Climate-driven oscillation of phosphorus and iron limitation in the north pacific subtropical gyre. *Proc. Natl. Acad. Sci.* 116, 12720–12728. doi: 10.1073/pnas.1900789116

Lewandowska, A. M., Boyce, D. G., Hofmann, M., Matthiessen, B., Sommer, U., and Worm, B. (2014). Effects of sea surface warming on marine plankton. *Ecol. Lett.* 17 (5), 614–623. doi: 10.1111/ele.12265

Lévy, M., Franks, P. J., and Smith, K. S. (2018). The role of submesoscale currents in structuring marine ecosystems. *Nat. Commun.* 9 (1), 1–16. doi: 10.1038/s41467-018-07059-3

Lim, H. G., Dunne, J. P., Stock, C. A., Ginoux, P., John, J. G., and Krasting, J. (2022a). Oceanic and atmospheric drivers of post-El-Niño chlorophyll rebound in the equatorial pacific. *Geophysical Res. Lett.* 49 (5), e2021GL096113. doi: 10.1029/2021GL096113

Lim, H. G., Dunne, J. P., Stock, C. A., and Kwon, M. (2022b). Attribution and predictability of climate-driven variability in global ocean color. *J. Geophysical Research: Oceans* 127 (10), e2022JC019121. doi: 10.1029/2022JC019121

Loisel, H., Nicolas, J.-M., Deschamps, P.-Y., and Frouin, R. (2002). Seasonal and inter-annual variability of particulate organic matter in the global ocean. *Geophys. Res. Lett.* 29 (24), 2196. doi: 10.1029/2002GL015948

Long, D., Shen, Y., Sun, A., Hong, Y., Longuevergne, L., Yang, Y., et al. (2014). Drought and flood monitoring for a large karst plateau in southwest China using extended GRACE data. *Remote Sens. Environ.* 155, 145–160. doi: 10.1016/j.rse.2014.08.006

Longhurst, A., Sathyendranath, S., Platt, T., and Caverhill, C. (1995). An estimate of global primary production in the ocean from satellite radiometer data. *J. plankton Res.* 17 (6), 1245–1271. doi: 10.1093/plankt/17.6.1245

Lundberg, S. M., and Lee, S. I. (2017). A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* 30. doi: 10.48550/arXiv.1705.07874

Makantasis, K., Karantzalos, K., Doulamis, A., and Doulamis, N. (2015). "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 4959–4962 (IEEE).

Martinez, E., Antoine, D., D'Ortenzio, F., and Gentili, B. (2009). Climate-driven basin-scale decadal oscillations of oceanic phytoplankton. *Science* 326 (5957), 1253–1256. doi: 10.1126/science.1177012

Martinez, E., Antoine, D., d'Ortenzio, F., and de Boyer Montegut, C. (2011). Phytoplankton spring and fall blooms in the north atlantic in the 1980s and 2000s. *J. Geophysical Research: Oceans* 116 (C11). doi: 10.1029/2010JC006836

Martinez, E., Brini, A., Gorgues, T., Drumetz, L., Roussillon, J., Tandeo, P., et al. (2020a). Neural network approaches to reconstruct phytoplankton time-series in the global ocean. *Remote Sens.* 12 (24), 4156. doi: 10.3390/rs12244156

Martinez, E., Gorgues, T., Lengaigne, M., Fontana, C., Sauzède, R., Menkes, C., et al. (2020b). Reconstructing global chlorophyll-a variations using a non-linear statistical approach. *Front. Mar. Sci.* 7, 464. doi: 10.3389/fmars.2020.00464

Martínez, H. P., and Yannakakis, G. N. (2014). "Deep multimodal fusion: Combining discrete events and continuous signals," in *Proceedings of the 16th International conference on multimodal interaction*. 34–41.

McClain, C. R., Signorini, S. R., and Christian, J. R. (2004). Subtropical gyre variability observed by ocean-color satellites. *Deep Sea Res. Part II: Topical Stud. Oceanography* 51 (1-3), 281–301. doi: 10.1016/j.dsr2.2003.08.002

Mélin, F., and Hoepffner, N. (2011). "Monitoring phytoplankton productivity from satellite–an aid to marine resources management," in *Handbook of satellite remote sensing image interpretation: Applications for marine living resources conservation and management*. Eds. J. Morales, V. Stuart, T. Platt and S. Sathyendranath (EU PRESPO and IOCCG), 79–93.

Melotti, G., Premebida, C., Gonçalves, N. M. D. S., Nunes, U. J., and Faria, D. R. (2018). "Multimodal CNN pedestrian classification: a study on combining LIDAR and camera data," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 3138–3143 (IEEE).

Meng, L., Yan, C., Zhuang, W., Zhang, W., and Yan, X. H. (2021). Reconstruction of three-dimensional temperature and salinity fields from satellite observations. *J. Geophysical Research: Oceans* 126 (11), e2021JC017605. doi: 10.1029/2021JC017605

Messié, M., and Chavez, F. P. (2012). A global analysis of ENSO synchrony: The oceans' biological response to physical forcing. *J. Geophysical Research: Oceans* 117 (C9). doi: 10.1029/2012JC007938

Yao, F., Zhang, J., Liu, Q., Liu, Q., Shi, L., Zhang, D., et al. (2022). Impact of dust deposition on phytoplankton biomass in the northwestern pacific: A long-term study from 1998 to 2020. *Sci. Total Environ.* 813, 152536. doi: 10.1016/j.scitotenv.2021.152536

Micheli-Tzanakou, E. (1987). "A neural network model of the vertebrate retina," in *9th Ann. Conf. IEEE Engineering in Medicine and Biology Society (Boston, MA, USA)*. 13–16.

Moore, J. K., Doney, S. C., Glover, D. M., and Fung, I. Y. (2001). Iron cycling and nutrient-limitation patterns in surface waters of the world ocean. *Deep Sea Res. Part II: Topical Stud. Oceanography* 49 (1-3), 463–507. doi: 10.1016/S0967-0645(01)00109-6

Moore, C. M., Mills, M. M., Arrigo, K. R., Berman-Frank, I., Bopp, L., Boyd, P. W., et al. (2013). Processes and patterns of oceanic nutrient limitation. *Nat. Geosci.* 6 (9), 701–710. doi: 10.1038/ngeo1765

Oliver, M. J., and Irwin, A. J. (2008). Objective global ocean biogeographic provinces. *Geophysical Res. Lett.* 35, L15601. doi: 10.1029/2008GL034238

Padarian, J., McBratney, A. B., and Minasny, B. (2020). Game theory interpretation of digital soil mapping convolutional neural networks. *Soil* 6 (2), 389–397. doi: 10.5194/soil-6-389-2020

Pan, B., Hsu, K., AghaKouchak, A., and Sorooshian, S. (2019). Improving precipitation estimation using convolutional neural network. *Water Resour. Res.* 55 (3), 2301–2321. doi: 10.1029/2018WR024090

Patara, L., Visbeck, M., Masina, S., Krahmann, G., and Vichi, M. (2011). Marine biogeochemical responses to the north Atlantic oscillation in a coupled climate model. *J. Geophysical Research: Oceans* 116 (C7). doi: 10.1029/2010JC006785

Pauthenet, E., Bachelot, L., Balem, K., Maze, G., Tréguier, A. M., Roquet, F., et al. (2022). Four-dimensional temperature, salinity and mixed layer depth in the gulf stream, reconstructed from remote sensing and *in situ* observations with neural networks. *Ocean Sci.* 18 (4), 1221–1244. doi: 10.5194/os-18-1221-2022

Polovina, J. J., Howell, E. A., and Abecassis, M. (2008). Ocean's least productive waters are expanding. *Geophysical Res. Lett.* 35 (3). doi: 10.1029/2007GL031745

Polovina, J. J., Howell, E., Kobayashi, D. R., and Seki, M. P. (2001). The transition zone chlorophyll front, a dynamic global feature defining migration and forage habitat for marine resources. *Prog. oceanography* 49 (1-4), 469–483. doi: 10.1016/S0079-6611(01)00036-2

Pyo, J., Cho, K. H., Kim, K., Baek, S. S., Nam, G., and Park, S. (2021). Cyanobacteria cell prediction using interpretable deep learning model with observed, numerical, and sensing data assemblage. *Water Res.* 203, 117483. doi: 10.1016/j.watres.2021.117483

Racault, M. F., Le Quéré, C., Buitenhuis, E., Sathyendranath, S., and Platt, T. (2012). Phytoplankton phenology in the global ocean. *Ecol. Indic.* 14 (1), 152–163. doi: 10.1016/j.ecolind.2011.07.010

Racault, M. F., Sathyendranath, S., Brewin, R. J., Raitsos, D. E., Jackson, T., and Platt, T. (2017). Impact of El Niño variability on oceanic phytoplankton. *Front. Mar. Sci.* 4, 133. doi: 10.3389/fmars.2017.00133

Radenac, M. H., Léger, F., Singh, A., and Delcroix, T. (2012). Sea Surface chlorophyll signature in the tropical pacific during eastern and central pacific ENSO events. *J. Geophysical Research: Oceans* 117 (C4). doi: 10.1029/2011JC007841

Ras, G., Xie, N., Van Gerven, M., and Doran, D. (2022). Explainable deep learning: A field guide for the uninitiated. *J. Artif. Intell. Res.* 73, 329–397. doi: 10.1613/jair.1.13200

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., and Carvalhais, N. (2019). Deep learning and process understanding for data-driven earth system science. *Nature* 566 (7743), 195–204. doi: 10.1038/s41586-019-0912-1

Reygondeau, G., Cheung, W. W., Wabnitz, C. C., Lam, V. W., Frölicher, T., and Maury, O. (2020). Climate change-induced emergence of novel biogeochemical provinces. *Front. Mar. Sci.* 7, 657. doi: 10.3389/fmars.2020.00657

Reygondeau, G., Longhurst, A., Martinez, E., Beaugrand, G., Antoine, D., and Maury, O. (2013). Dynamic biogeochemical provinces in the global ocean. *Global Biogeochemical Cycles* 27 (4), 1046–1058. doi: 10.1002/gbc.20089

Reynolds, R. W., Rayner, N. A., Smith, T. M., Stokes, D. C., and Wang, W. (2002). An improved *in situ* and satellite SST analysis for climate. *J. Climate* 15 (13), 1609–1625. doi: 10.1175/1520-0442(2002)015<1609:AIISAS>2.0.CO;2

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. 234–241 (Springer International Publishing). doi: 10.1007/978-3-319-24574-4_28

Roussillon, J., Fablet, R., Gorgues, T., Drumetz, L., Littaye, J., and Martinez, E. (2022). satellIte phytoplaNkton drivers in the global ocean over 1998-2015 (INDIGO benchmark dataset). *SEANOE*. doi: 10.17882/91910

Ryan, M., and Stahl, B. C. (2021). Artificial intelligence ethics guidelines for developers and users: clarifying their content and normative implications. *J. Information Communication Ethics Soc.* 19 (1), 61–86. doi: 10.1108/JICES-12-2019-0138

Sammartino, M., Buongiorno Nardelli, B., Marullo, S., and Santoleri, R. (2020). An artificial neural network to infer the Mediterranean 3D chlorophyll-a and temperature fields from remote sensing observations. *Remote Sens.* 12 (24), 4123. doi: 10.3390/rs12244123

Sathyendranath, S., Brewin, R. J. W., Brockmann, C., Brotas, V., Calton, B., Chuprin, A., et al. (2019). An ocean-colour time series for use in climate studies: the experience of the ocean-colour climate change initiative (OC-CCI). *Sensors* 19, 4285. doi: 10.3390/s19194285

Sathyendranath, S., Platt, T., Caverhill, C. M., Warnock, R. E., and Lewis, M. R. (1989). Remote sensing of oceanic primary production: computations using a spectral model. deep Sea research part a. *Oceanographic Res. Papers* 36 (3), 431–453. doi: 10.1016/0198-0149(89)90046-0

Sauzède, R., Claustre, H., Uitz, J., Jamet, C., Dall'Olmo, G., d'Ortenzio, F., et al. (2016). A neural network-based method for merging ocean color and argo data to extend surface bio-optical properties to depth: Retrieval of the particulate backscattering coefficient. *J. Geophysical Research: Oceans* 121 (4), 2552–2571. doi: 10.1002/2015JC011408

Schollaert Uz, S., Busalacchi, A. J., Smith, T. M., Evans, M. N., Brown, C. W., and Hackert, E. C. (2017). Interannual and decadal variability in tropical pacific chlorophyll from a statistical reconstruction: 1958–2008. *J. Climate* 30 (18), 7293–7315. doi: 10.1175/JCLID-16-0202.1

Schulz, M., Prospero, J. M., Baker, A. R., Dentener, F., Ickes, L., Liss, P. S., et al. (2012). Atmospheric transport and deposition of mineral dust to the ocean: Implications for research needs. *Environ. Sci. Technol.* 46 (19), 10390–10404. doi: 10.1021/es300073u

Shen, C. (2018). A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resour. Res.* 54 (11), 8558–8593. doi: 10.1029/2018WR022643

Signorini, S. R., Franz, B. A., and McClain, C. R. (2015). Chlorophyll variability in the oligotrophic gyres: mechanisms, seasonality and trends. *Front. Mar. Sci.* 2, 1. doi: 10.3389/fmars.2015.00001

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15 (1), 1929–1958. doi: 10.5555/2627435.2670313

Sun, A. Y., Scanlon, B. R., Zhang, Z., Walling, D., Bhanja, S. N., Mukherjee, A., et al. (2019). Combining physically based modeling and deep learning for fusing GRACE satellite data: can we learn from mismatch? *Water Resour. Res.* 55 (2), 1179–1195. doi: 10.1029/2018WR023333

Szeto, M., Werdell, P. J., Moore, T. S., and Campbell, J. W. (2011). Are the world's oceans optically different? *J. Geophysical Research: Oceans* 116 (C7). doi: 10.1029/2011JC007230

Taddeo, M., Tsamados, A., Cowls, J., and Floridi, L. (2021). Artificial intelligence and the climate emergency: Opportunities, challenges, and recommendations. *One Earth* 4 (6), 776–779. doi: 10.1016/j.oneear.2021.05.018

Tagliabue, A., Aumont, O., DeAth, R., Dunne, J. P., Dutkiewicz, S., Galbraith, E., et al. (2016). How well do global ocean biogeochemistry models simulate dissolved iron distributions? *Global Biogeochemical Cycles* 30 (2), 149–174. doi: 10.1002/2015GB005289

Vinuesa, R., Azizpour, H., Leite, I., Balaam, M., Dignum, V., Domisch, S., et al. (2020). The role of artificial intelligence in achieving the sustainable development goals. *Nat. Commun.* 11 (1), 233. doi: 10.1038/s41467-019-14108-y

Weyn, J. A., Durran, D. R., and Caruana, R. (2020). Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *J. Adv. Modeling Earth Syst.* 12 (9), e2020MS002109. doi: 10.1029/2020MS002109

Wilson, C., and Adamec, D. (2002). A global view of bio-physical coupling from SeaWiFS and TOPEX satellite data 1997–2001. *Geophysical Res. Lett.* 29 (8), 98–91. doi: 10.1029/2001GL014063

Wilson, C., and Coles, V. J. (2005). Global climatological relationships between satellite biological and physical observations and upper ocean properties. *J. Geophysical Research: Oceans* 110 (C10). doi: 10.1029/2004JC002724

Winder, M., and Sommer, U. (2012). Phytoplankton response to a changing climate. *Hydrobiologia* 698, 5–16. doi: 10.1007/s10750-012-1149-2

Xie, N., Ras, G., van Gerven, M., and Doran, D. (2020). Explainable deep learning: A field guide for the uninitiated. *arXiv preprint arXiv:2004.14545*. doi: 10.48550/arXiv.2004.14545

Yang, X., Molchanov, P., and Kautz, J. (2016). "Multilayer and multimodal fusion of deep neural networks for video classification," in *Proceedings of the 24th ACM international conference on Multimedia*. 978–987.

Ye, H., Tang, S., and Yang, C. (2021). Deep learning for chlorophyll-a concentration retrieval: A case study for the pearl river estuary. *Remote Sens.* 13 (18), 3717. doi: 10.3390/rs13183717

Yu, B., Xu, L., Peng, J. H., Hu, Z., and Wong, A. (2020). Global chlorophyll-a concentration estimation from moderate resolution imaging spectroradiometer using convolutional neural networks. *J. Appl. Remote Sens.* 14 (3), 034520. doi: 10.1117/1.JRS.14.034520

Yuan, Q., Shen, H., Li, T., Li, Z., Li, S., Jiang, Y., et al. (2020). Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sens. Environ.* 241, 111716. doi: 10.1016/j.rse.2020.111716

Zeiler, M. D., and Fergus, R. (2014). "Visualizing and understanding convolutional networks," in *European conference on computer vision*. 818–833.

Zhang, H., Yao, J., Ni, L., Gao, L., and Huang, M. (2022). Multimodal attention-aware convolutional neural networks for classification of hyperspectral and LiDAR data. *IEEE J. Selected Topics Appl. Earth Observations Remote Sens.* doi: 10.1109/JSTARS.2022.3187730

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2921–2929.

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 23 March 2023
DOI 10.3389/fmars.2023.1153416

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Peng Ren,
China University of Petroleum (East China), China
Qiqi Zhu,
China University of Geosciences Wuhan, China

*CORRESPONDENCE
Heng Li
12309119@kust.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 29 January 2023
ACCEPTED 13 March 2023
PUBLISHED 23 March 2023

CITATION
Zhang C, Zhang G, Li H, Liu H, Tan J and Xue X (2023) Underwater target detection algorithm based on improved YOLOv4 with SemiDSConv and FIoU loss function. *Front. Mar. Sci.* 10:1153416. doi: 10.3389/fmars.2023.1153416

COPYRIGHT
© 2023 Zhang, Zhang, Li, Liu, Tan and Xue. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Underwater target detection algorithm based on improved YOLOv4 with SemiDSConv and FIoU loss function

Chengpengfei Zhang[1], Guoyin Zhang[1], Heng Li[1*], Hui Liu[1], Jie Tan[2] and Xiaojun Xue[1]

[1]Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China, [2]College of Engineering, Tongren Polytechnic College, Tongren, China

Underwater target detection is an indispensable part of marine environmental engineering and a fast and accurate method of detecting underwater targets is essential. Although many target detection algorithms have achieved great accuracy in daily scenes, there are issues of low-quality images due to the complex underwater environment, which makes applying these deep learning algorithms directly to process underwater target detection tasks difficult. In this paper, we presented an algorithm for underwater target detection based on improved You Only Look Once (YOLO) v4 in response to the underwater environment. First, we developed a new convolution module and network structure. Second, a new intersection over union loss was defined to substitute the original loss function. Finally, we integrated some other useful strategies to achieve more improvement, such as adding one more prediction head to detect targets of varying sizes, integrating the channel attention into the network, utilizing K-means++ to cluster anchor box, and utilizing different activation functions. The experimental results indicate that, in comparison with YOLOv4, our proposed algorithm improved the average accuracy of the underwater dataset detection by 10.9%, achieving 91.1%, with a detection speed of 58.1 frames per second. Therefore, compared to other mainstream target detection algorithms, it is superior and feasible for applications in intricate underwater environments.

KEYWORDS

deep learning, underwater detection, YOLO, convolutional neural network, loss function

1 Introduction

Underwater target detection technology has been widely used in marine biodiversity monitoring, marine ecosystem health assessment, and smart mariculture (Akkaynak and Treibitz, 2019). Due to the difficulties in data acquisition and the intricate underwater environment, underwater target detection has been an important and challenging task when it comes to detecting targets. The existing research on underwater target detection

methods can be broadly classified into two types: one is the traditional approach based on using hand-crafted features and shallow classifiers, and the other is a deep learning approach based on automatic feature extraction. Traditional target detection algorithms usually use a sliding window approach to delineate the region of interest on the input picture that may contain the target. Then, features will be extracted from the region-of-interest by using feature extraction algorithms, such as histogram of oriented gradient(HOG) (Dalal and Triggs, 2005), oriented fast and rotated brief(ORB)(Rublee et al., 2011), and scale-invariant feature transform(SIFT)(Lowe, 2004). Finally, classifiers such as adaboost (Yoav and Schapire, 1997), support vector machine (SVM) (Cortes and Vapnik, 1995), and deformable part model(DPM) (Felzenszwalb et al., 2008). are used to classify the extracted features. However, traditional target detection algorithms have many disadvantages, such as their poor robustness, low efficiency, and limited accuracy, which makes it difficult to meet the current demand. For the past few years, deep convolutional neural networks(DCNN) have been widely used in many fields such as medical image semantic segmentation (Wang Z. et al., 2022), urban land-use planning (Zhu et al., 2022), and autonomous driving (Li and Jin, 2022), with satisfactory results. Many approaches based on DCNN principles have been devised, and their effectiveness has been proven in a variety of domains, including in underwater target detection.

Target detection methods based on DCNN are gradually evolving in two directions due to the divergent focus on detection accuracy and detection speed. One is a region proposal-based target detection algorithm, also called the two-stage algorithm. Among all these algorithms, the R-CNN series is the most representative. R-CNN (Girshick et al., 2014) was presented by R. Girshick et al. in 2014, and it significantly outperformed the mainstream algorithm on the Pascal VOC dataset. It applies a selective search method to engender region proposals and uses CNN to extract features. After that, features are classified using SVM. Based on R-CNN, Fast R-CNN(Girshick, 2015), Faster R-CNN (Ren et al., 2017), and Mask R-CNN (He et al. 2018), many other two-stage methods have been gradually proposed and achieved better accuracy and speed. However, these two-stage algorithms have high computation time, which makes it difficult to meet the needs for real-time target detection. In order to resolve this issue, the regression-based target detection algorithm, also called the one-stage algorithm, was proposed. You Only Look Once (YOLO) (Redmon et al., 2016) was first introduced by J. Redmon et al.in 2015. When it was proposed, it attracted a lot of attention. YOLO's core idea is to use the whole picture as the input to the CNN and output the result of bounding box prediction. (Zhang et al., 2022) Because of this, YOLO has fast detection speed. Since its development, one-stage algorithms such as single shot multibox detector (SSD) (Liu et al., 2016) and RetinaNet (Lin et al., 2017). Were gradually proposed, and one-stage target detection algorithms were developed rapidly.

Although most of the algorithms mentioned above have achieved good performance in daily scenes, applying these deep learning algorithms directly to process underwater target detection tasks still has some problems. Firstly, the targets have a relatively large variation in scale due to the shooting distance. Secondly, underwater images are generally low-quality due to the complex and changing underwater environment, which means models have a low target localization accuracy in the underwater target detection assignment. Finally, looking at the research on underwater target recognition based on deep learning, although most of the existing detection methods have high recognition precision, the real-time performance of many of them is insufficient due to their high complexity, large number of parameters, and large scale. Therefore, it is essential to develop an underwater target detection algorithm that meets the needs for real-time detection while ensuring recognition accuracy.

In this paper, we presented an algorithm for underwater target detection based on improved YOLOv4 (Bochkovskiy et al., 2020) to solve the above-mentioned issues. In terms of network structure, we followed the original version, used CSPDarknet53 (Wang et al., 2020) as the backbone, and introduced channel attention block into it to emphasize useful informative features. Then, we constructed a new convolution module by integrating the traditional convolution, the depthwise separable convolution (DSC), and channel shuffle (Zhang et al., 2018), named SemiDSConv for convenience. This module can ensure the performance similar to a traditional convolution network, reduce the computational cost, and speed up the inference while solving the channel information separation problem caused by DSC. Based on this new module, inspired by CSPNet, we further designed the SemiDSCSP module, and applied it with the SemiDSConw module to the neck part of the model to replace the original convolution network and further reduce the inference time. In the head part, we added a prediction head to help the model deal with large changes in the targets' scale. Meanwhile, we defined a new intersection over union (IoU) loss function, FIoU, which boosts the localization accuracy and the convergence speed of the model. In comparison with the original YOLOv4, our improved YOLOv4 can better deal with underwater target detection tasks. For the dataset of URPC, the mAP was increased by 10.9% with the baseline and the inference speed reaching 58.1 frames per second (FPS). Overall, the presented algorithm demonstrates good results with a quick speed. The contributions of our work can be summed up as follows:

1. Developed a new convolution module named SemiDSConv. This module's performance is close to the traditional convolution network, but with less computation and faster inference speed. Based on it, the SemiDSCSP module was then designed and replaced the traditional convolution in the neck part;
2. Defined a new IoU loss, FIoU, that obtains superior localization accuracy and faster convergence speed;
3. Integrated some other useful tricks, such as introducing the channel attention block which can help the network to extract useful informative features more easily, adding a new prediction head to deal with dramatic changes in the scale of the underwater targets, using Mish as activation function, and using the K-means++ clustering algorithm to cluster anchor boxes;
4. On the URPC dataset, the proposed method achieved 91.1% mAP, outperforming the baseline by 10.9% with 58.1 FPS.

2 Related work

2.1 YOLOv4

Since the YOLO algorithm was first presented by J. Redmon et al. in 2015, it has received great attention among researchers. YOLOv4 was introduced in 2020 and is one of the state-of-the-art object detection algorithms. It greatly improved the detection accuracy and computational speed of YOLOv3 (Redmon and Farhadi, 2018). On COCO target detection dataset, YOLOv4 improves YOLOv3's FPS by 12%. Compared to other one-stage algorithms, such as SSD, YOLOv4 has a detection accuracy that far exceeds theirs while having the speed to meet real-time detection requirements. Compared to YOLOv5 and v7, it is lighter and has a faster detection speed when handling underwater target detection tasks with not much difference in accuracy. Thus, YOLOv4 is suitable for real-time target detection tasks.

YOLOv4 mainly consists of three sections: the backbone, the neck, and the head. YOLOv4 takes CSPDarknet53 as the backbone network. CSPDarknet53 is composed of five large residual blocks which contains one, two, eight, eight, and four residual units in them, respectively. Each residual unit consists of 3*3 and 1*1 convolutional layers. This architecture can help the network to get richer gradient information while reducing the amount of calculation needed. In the neck part, YOLOv4 uses PANet(Liu et al., 2018) to fuse the feature information from different-size feature maps to enhance the ability of the model to detect objects of various sizes. Meanwhile, Yolov4 adds the SPP block into the network which can expand the receptive field, prevent overfitting, and improve scale-invariance. In the end, the extracted multi-scale feature maps are sent into the YOLOv3 detection head for detection.

2.2 Channel attention

Channel attention mechanisms have shown their utility across many tasks. For the underwater image, typically, targets only occupy a fraction of the whole image, and the rest is background information. In order to minimize the distractions of background information and highlight the target, channel attention can be used to help distinguish the target from the background as channel attention focuses on what is meaningful given an image (Woo et al., 2018). SENet(Squeeze-and-Excitation Network) (Hu et al., 2018) was proposed by Jie Hu et al., which is a prominent representative of channel attention. It is composed of two parts: a squeeze operation and an excitation operation. The squeeze operation uses global average pooling to aggregate the summarized information from each channel, and the excitation operation adjusts the relevance of each channel according to its weight. Therefore, the introduction of the SE block can enhance the feature extraction capability of the model. The structure of the SE block is indicated in Figure 1.

2.3 Activation functions

The activation Function is one of the crucial factors influencing the performance of a neural network. The rectified linear unit (ReLU) (Glorot et al., 2011) was proposed by Vinod Nair et al. in 2011. Its formula is defined in Equation (1).

$$\mathbf{f_{ReLU}(x) = max\,(0, x)}\ , x \in \mathbf{R}\,. \tag{1}$$

Due to its low computational cost and easy optimization characteristics, ReLU is widely used in neural networks. However, it is not without weaknesses. As shown in Equation (1), ReLU grows unbounded and is directly truncated at negative values. The former would lead to excessive differences in weights, resulting in reduced accuracy. The latter would result in a Dead ReLU problem, i.e. if the input is a negative value, the output of ReLU and the gradient will become zero. Finally, the network parameters will not be updated. Alex Krizhevsky proposed ReLU6 (Krizhevsky and Hinton, 2010) to address the former issue, which is formulated in Equation (2).

$$\mathbf{f_{ReLU6}(x) = \min\,(6, \max\,(0, x))}\ , x \in R\,. \tag{2}$$

But it still does not solve the Dead ReLU problem. In 2019, Diganta Misra et al. presented Mish activation function (Misra, 2019), which can be defined as:

$$\mathbf{f_{Mish}(x) = xtanh(\ln\,(1 + e^{x}))}, x \in R\,. \tag{3}$$

Compared with ReLU, Mish is non-monotonic, smoother, and allows a few negative weight inflow. Figure 2 shows visually the differences between ReLU, ReLU6, and Mish. Better expressivity and information flow are facilitated by these properties, and these properties also make the network avoid saturation.

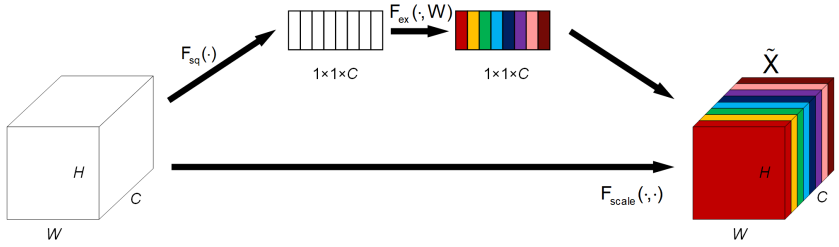


FIGURE 1
The structure of the SE block.

2.4 General target detection

Since the rise of convolutional neural networks, many researchers have continued to propose new methods and ideas due to the need for various target detection tasks. Aiming to improve the assignment of anchor labels in the current anchor-based model, Kim (Kim and Lee, 2020) et al. proposed a probabilistic model for assigning labels to anchors - Probabilistic Anchor Assignment(PAA), the assignment criteria of which depend on the combination of classification accuracy and IoU, rather than IoU alone. Redundant hyperparameters such as IoU threshold and number of positive samples, are then discarded to improve the performance and stability of the model. Yang (Yang et al., 2022) et al. proposed the Cascade Sparse Query (CSQ) mechanism, where Query represents using the query passed in the deeper-level (higher-level feature with lower resolution) layer to guide the detection of small targets in this layer, and then predicting the query in this layer to be further passed to the next layer. Sparse represents the significant reduction of the computational overhead of the detection head on the low-level feature layer by using sparse convolution. Li(Li et al., 2022) et al. improved Multiscale Vision Transformers which incorporates decomposed relative positional embeddings, proposed MViTv2, and optimized the pooling attention in the network using residual structures. After that, many experiments have been conducted to verify the superiority of the proposed algorithm in the fields of classification, detection, and video tracking. To address the problem of sample scarcity in the dataset, Hou(Hou et al., 2022) et al. creatively proposed a new idea to explore the relationship between samples and help the network to learn by focusing on the batch dimension and introducing the Transformer structure in it. The proposed BatchFormer has achieved good performance in a large number of experiments.

2.5 Underwater target detection

In the past few years, with the evolution of deep learning-based target detection algorithms, more and more researchers have been implementing this technology in the underwater environment. In 2019, Moniruzzaman (Moniruzzaman et al., 2019) et al. constructed a Halophila ovalis dataset that consists of 2,699 underwater photographs of Halophila ovalis and presented Inception V2-based Faster R-CNN network to detect seagrass. Experimentally, the proposed network achieved a high mAP of 0.3464 on laboratory images. In 2021, Zeng (Zeng et al., 2021) et al. presented a method to introduce the adversarial occlusion network (AON) to the Faster R-CNN algorithm and the resulting model achieves better robustness in terms of underwater seafood. In the same year, Wang (Wang et al., 2021) et al. introduced YOLOv5 for underwater target detection and conducted a lot of detailed experiments and comparisons based on this, and finally used the experimental results as the YOLOv5 baseline for underwater target detection. For the task of underwater sea cucumber target detection, Peng (Peng et al., 2021) et al. proposed the Shortcut Feature Pyramid Network (S- FPN) and Piecewise Focal Loss (PFL), which improved the multi-scale feature fusion approach of the network and balanced the positive and negative samples, enabling the mAP to achieve a high accuracy of 94%. Yeh (Yeh et al., 2021) et al. proposed an underwater target detector with joint image color conversion for the problem of underwater image color absorption, which converts underwater color images to grayscale images, and improved the performance of the target detector with low computational cost. In 2022, Hong (Hong et al., 2022) et al. used a parameter calibration strategy to fine-tune the parameters of the Mask RCNN model to detect and locate shrimp better. Cai (Cai et al., 2022) et al. proposed a weakly supervised learning framework for underwater object detection, using two detectors trained

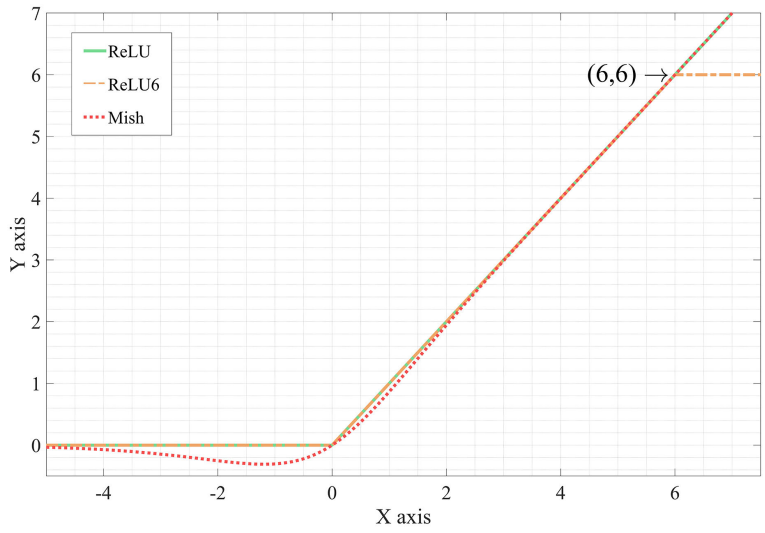


FIGURE 2
Comparison between ReLU, ReLU6, and Mish.

simultaneously and learning from each other to select cleaner samples, which eventually achieved good performance. Chen (Chen et al., 2022) et al. proposed the Sample-WeIghted hyPEr Network (SWIPENET) and a novel training paradigm called Curriculum Multi-Class Adaboost (CMA) to address both problems simultaneously for the case of ambiguous underwater targets and the presence of many small targets, which eventually achieved good performance. In 2023, Wang (Wang et al., 2023) et al. proposed a new underwater target detection algorithm based on reinforcement learning and image enhancement, which automatically learns and adjusts the combined sequence of underwater image enhancement methods by a neural network in order to help the network's detector achieve the best performance. Although these works achieved quite a high degree of detection accuracy, there are still some limitations to them, namely the low detection speed. Therefore, how to ensure a high detection accuracy with real-time rapid detection is still a research issue worthy of study.

3 Proposed model

3.1 Network structure

Considering the speed requirements of real-time detection tasks, we chose the best-known and the most used one-stage algorithm—YOLOv4-as our baseline. The framework of the improved YOLOv4 is shown in Figure 3. We introduced a new convolution module and a bottleneck structure based on it to speed up the network inference. A new IoU loss function was developed to enhance detection precision and the velocity of convergence. A new prediction head was added to deal with the large differences in underwater target scales. The prediction head we added uses mainly high-resolution and shallow features to predict, which makes it sensitive to small targets. Therefore, the newly added prediction head and the original prediction heads form a four-head structure that can better handle the drastic changes in the size of underwater targets. The channel attention module was introduced into the backbone to encourage the network to retain more useful features. In addition, we used Mish activation function to replace ReLU. It solves the Dead ReLU problem, avoids network convergence slowdown, and, at the same time, improves the accuracy of the network. Although it slightly increases the computational cost, we deem it worthwhile.

3.2 SemiDSConv module

The depthwise separable convolution (DSC) is composed of two parts: depthwise convolution and pointwise convolution. Depthwise convolution convolves each channel of the input feature map separately. If the amount of input channels is N, after convolving each of the N channels, these feature maps are collocated together to get an output feature map of channel N. Pointwise convolution is a 1×1 convolution. The pointwise convolution in DSC is mainly used to allow DSC to freely change the number of output channels and to perform channel fusion on the output feature map of depthwise convolution. The ratio of the computational cost of DSC to conventional convolution is illustrated in Equation (4)

$$\frac{k\cdot k\cdot n\cdot s\cdot s+n\cdot m\cdot s\cdot s}{k\cdot s\cdot n\cdot m\cdot s\cdot s}=\frac{1}{m}+\frac{1}{k^2} \tag{4}$$

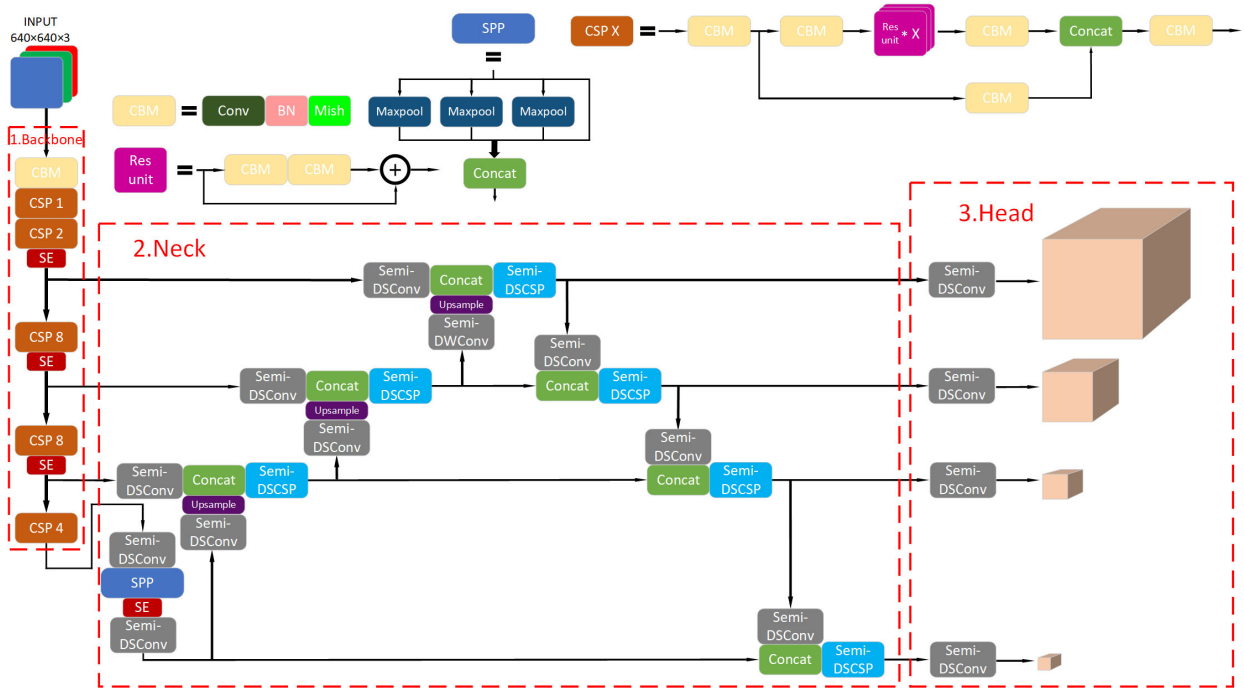


FIGURE 3
The improved YOLOv4 network structure.

Where *k·k.* is the convolution kernel size. *n* and *m* denote the input and output channels, separately. *s·s* represents the size of the feature map. From the equation, it is clear that the computational cost of DSC is much less than that of traditional convolution.

However, due to the characteristics of DSC, channel information is computed separately from each other, resulting in a significant reduction in its capability to extract and fuse features, much weaker than traditional convolution. To overcome this issue, the SemiDSConv module was designed. The structure of the SemiDSConv module is indicated in Figure 4.

The SemiDSConv module first uses a 1*1 convolution kernel to fuse the input features maps, while achieving channel dimensionality reduction to reduce the computational cost of subsequent convolution operations. After that, the feature maps are computed through the traditional convolution and the depthwise separable convolution, respectively. The channels are then concatenated together. It then performs shuffle operations so that the information between the channels is completely fused. The SemiDSConv module effectively maintains the advantages of DSC while minimizing the negative impact of its shortcomings on the network.

Based on this, inspired by the CSPNet, we also designed the SemiDSCSP module, which enables the network to better extract and fuse the feature information. The structure of the SemiDSCSP module is indicated in Figure 5.

It is worth mentioning that if all traditional convolutions in the network are replaced with SemiDSConv, the number of network layers will be too deep. This would make the resistance of data flow too high and increase the inference time significantly. In the Neck part, the feature map is extracted by the backbone, with smaller width and height, less redundant repetitive information, and shorter inference time. Therefore, we replaced traditional convolutions only in the Neck to achieve good performance.

3.3 FIoU loss function

Due to differences in the network structure and the basic idea, YOLO has its natural disadvantage in localization precision compared with a two-stage algorithm. Therefore, the authors of the YOLO series and other researchers have been exploring strategies to address this issue. Among the various improvement strategies, improving the loss function is the most effective and direct strategy. YOLOv4 includes three types of loss functions: confidence loss, category loss, and localization loss (also called the loss of bounding box coordinates). Different from YOLOv3, YOLOv4 substitutes Complete-IoU (CIoU) (Zheng et al., 2021) loss for cross entropy loss in YOLOv3 as the localization loss function and obtains better convergence speed and accuracy (Jiao et al., 2022). The CIoU loss was improved from Distance-IoU (DIoU) (Zheng et al., 2020) loss. The DIoU loss and the CIoU loss is defined in Equations (5)-(9):

$$\mathcal{L}_{DIoU}=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2} \tag{5}$$

$$\mathcal{L}_{CIoU}=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2}+\alpha v. \tag{6}$$

$$IoU=\frac{|A\cap A_{gt}|}{|A\cup A_{gt}|}. \tag{7}$$

$$\alpha=\frac{v}{(1-IoU)+v}. \tag{8}$$

$$v=\frac{4}{\pi^2}(\arctan\frac{w_{gt}}{h_{gt}}-\arctan\frac{w}{h})^2. \tag{9}$$

where p and p_{gt} are the central points of the predicted box and the ground-truth box. d is the diagonal length of the minimum bounding rectangle. ρ (p, p_{gt}) indicates the Euclidean distance between p and p_{gt}. A denotes the predicted box whereas A_{gt} denotes the ground-truth box. w, w_{gt},h, and h_{gt} respectively represent the width of the predicted box and ground-truth box and the height of the two boxes.

As shown in Equations (6), (8), and (9), the newly added penalty term av is to measure the discrepancy of aspect ratio between the predicted box and the ground-truth box. The experimental results indicate that, compared with previous IoU loss functions (GIoU and DIoU) (Rezatofighi et al., 2019), the localization accuracy and the convergence speed of the CIOU loss have substantially increased. However, CIoU still has certain limitations. Specifically, when $\{w= kw_{gt} = kh_{gt}|k\in R^+\}$ is satisfied, v becomes zero and the loss function will degrade to DIoU loss. This drawback renders the convergence speed slow in some cases. For the underwater target detection task, the slow convergence of the loss function may cause the network to fail and to converge quickly

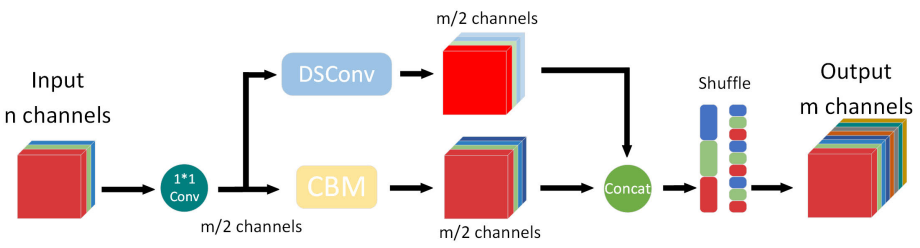


FIGURE 4
The structure of the SemiDSConv module.

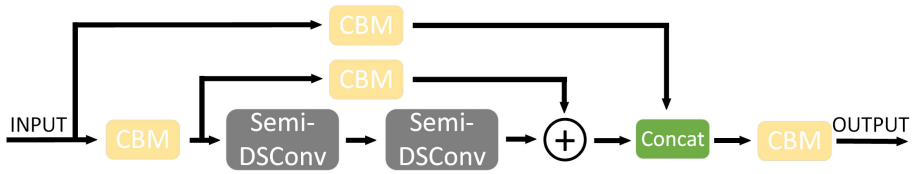


FIGURE 5
The structure of the SemiDSCSP module.

in a limited number of epochs due to the small number of samples. It may also lead to overfitting if the training epochs are extended for model convergence.

In order to address this situation, we designed a new loss function that inherited some properties from CIoU loss and added proper penalty terms to it. We call it Fast-IoU(FIoU); the specific formula is shown as follows:

$$\mathcal{L}_{FIoU}=L_{IoU}+L_D+L_R+L_L=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2}+\alpha v+\frac{\rho^2(h,h_{gt})}{l_h^2}+\frac{\rho^2(w,w_{gt})}{l_w^2}. \quad (10)$$

where, l_h and l_w are the height and width of the minimum bounding rectangle. As shown in Equation (10), we divide the whole loss function into four parts: the IoU loss L_{IoU}, the distance loss L_D, the aspect ratio loss L_R and the side length loss L_L. Generally, L_R and L_L function together to optimize the similarity between two boxes. If $\{w = kw_{gt}, h = kh_{gt} | k \in R^+\}$ is satisfied, although L_R becomes zero, L_L it is still minimizing the difference between the two boxes' width and height. The convergence process of the CIoU and the FIoU is shown in Figure 6.

In order to verify the effect of different loss functions on the network model performance, we evaluate FIoU loss function by replacing CIoU with FIoU in the original YOLOv4 algorithm. Figure 7 shows the training loss curves of two models in the URPC dataset. As can be seen, the FIoU decreased more quickly than CIoU in epochs 0 to 30. After 30 epochs, the curve of FIoU loss functions was stable while CIoU was not. Although after 45 epochs, both the FIoU and the CIoU loss functions were stabilized, FIoU was still well below CIoU. It verifies that the FIOU loss function has a quicker convergence rate and better regression accuracy than the CIOU loss function.

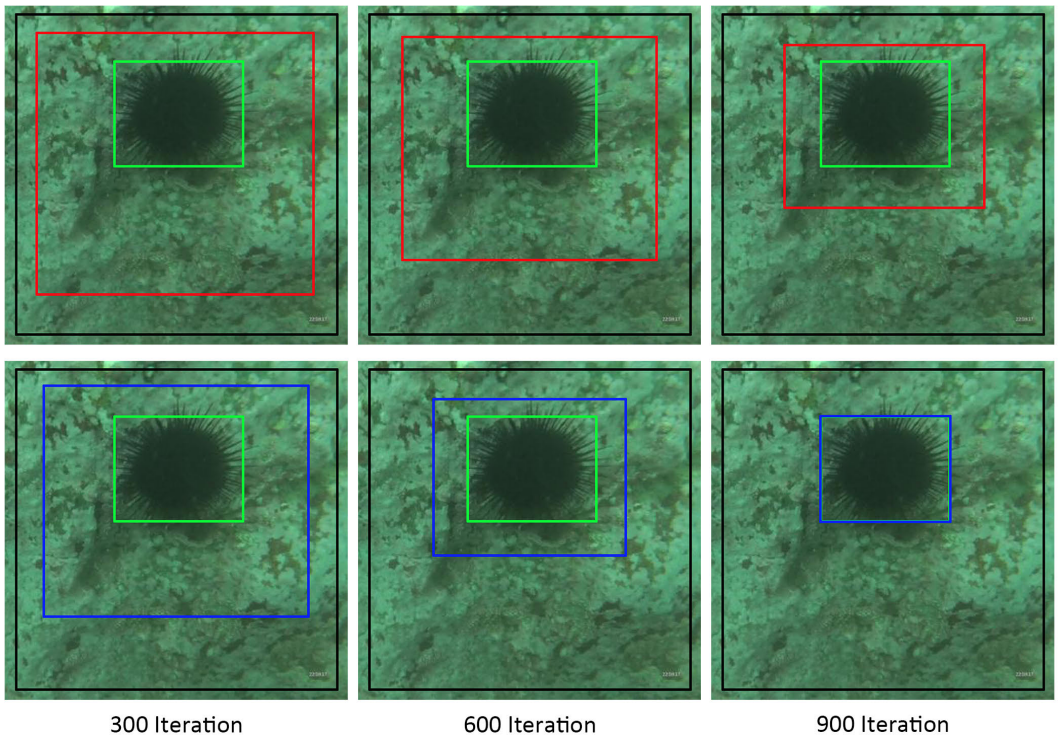


FIGURE 6
The diagrams of prediction box regression in the first and second row respectively represent the prediction box regression process of CIoU and FIoU. The green box refers to the ground truth box. The black box refers to the anchor box, and the red and blue one is the prediction boxes of CIoU and FIoU, respectively.

Overall, compared to CIoU loss, FIoU can get better localization accuracy and convergence speed. This enables the YOLOv4 network using FIoU as the loss function to have a higher performance than the network using CIoU as the loss function. We substitute FIoU loss for CIoU loss in YOLOv4, hoping to render it better for the underwater target detection task.

4 Experiments

4.1 Dataset

The dataset adopted in the paper was from the Target Recognition Group of China Underwater Robot Professional Competition (URPC), which includes four categories: echinus, holothurian, scallop, and starfish. The dataset contained 4757 images in total. The dataset is a sequence of frames from multiple video segments with a continuous distribution and a large similarity between neighboring frames. Therefore, we shuffled the dataset randomly and split the dataset into a training and test set at a ratio of 4:1, then labeled the targets. In order to better simulate the real situation in the underwater environment, we kept the images without targets detected in the training set and test set. The finally obtained training set contains 3806 images and the test set contains 951 images. One practical issue deserves mention: the resolution of images and the number of individual category samples are very unbalanced in the dataset. This would bring challenges to the training of the network.

4.2 Model evaluation metrics

In the field of target detection, Average Precision(AP) is the metric most commonly used to evaluate the performances of the model. Before introducing AP, we present a brief overview of precision (P) and recall (R), which are computed by Equations (11) and (12):

$$\mathbf{P}=\frac{\mathbf{TP}}{\mathbf{TP+FP}}\times\mathbf{100\%}. \tag{11}$$

$$\mathbf{R}=\frac{\mathbf{TP}}{\mathbf{FN+TP}}\times\mathbf{100\%} \tag{12}$$

where *TP*, *FP* and *FN* refers to the positive samples predicted to be positive by the model, the negative samples predicted by the model to be positive, and the positive samples predicted to be negative by the model, respectively.

Because P and R are interactive, to combine the two metrics, AP is introduced to evaluate the goodness of the detection accuracy of the model, as defined in Equation (13):

$$\mathbf{AP}=\int_0^1 \mathbf{P(R)dR}. \tag{13}$$

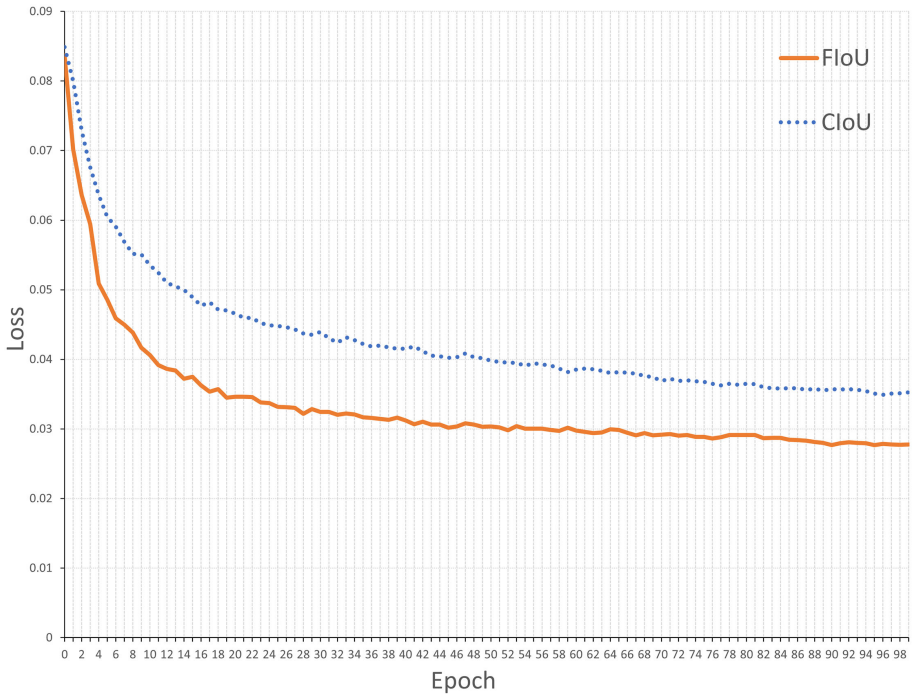


FIGURE 7
Curves of the FIoU and CIoU loss values with the epoch increasing.

In multi-class target detection tasks, mean Average Precision is commonly used to evaluate the overall model performance. Namely, AP values were averaged for each category. The equation for calculating mAP as defined below:

$$\mathbf{mAP}=\frac{1}{n}\sum_{i=0}^{n}\mathbf{AP}. \quad (14)$$

where n refers to the number of types.

4.3 Experimental environment and parameter settings

We implement the proposed method on Python 3.9.7 and Pytorch 1.8.1. All the methods were trained and tested using an NVIDIA RTX3090 GPU and an Intel Xeon E7-4809 v3 CPU.

During the training phase, we set the initial training hyperparameters for each group of experiments to be the same to ensure the fairness of our experiments. The resolution of the input images were consistently set to 640 × 640. To prevent the gradients from exploding when the learning rate was high, the learning rate was tuned based on the cosine annealing strategy (Loshchilov and Hutter, 2016).

YOLOv4 algorithm expands the anchor mechanism. Setting a predefined prior frame can well represent the original state of the target to be detected and get a more reasonable potential distribution of data sample bounding boxes. The high-quality anchor can play an optimal role in the process of small target detection and post-processing prediction. Therefore, when training underwater data, it is very important to set appropriate anchors according to the characteristics of the underwater dataset. In this paper, we used the K-means++ (Arthur and Vassilvitskii, 2007) clustering algorithm to cluster anchor boxes in the URPC dataset. Finally, we obtain the anchor parameters' fit among the underwater targets. The clustered anchor boxes are (17,14), (24,21), (31,28), (37,39), (48,32), (54,46), (69,62), (92,89), and (144,129).

The specific settings of the other hyperparameters are shown in Table 1.

The loss function curves of the proposed method are demonstrated in Figure 8, which contains three parts: localization loss, classification loss, and confidence loss. From the figure, it can be noted that all losses steadily decrease with the number of epochs. The model converged in under 100 epochs.

In the testing stage, all the resolutions of the input image were consistently set to 640 × 640. The IoU threshold was set to 0.4. All other parameters were the same. During the test, only one GPU was used uniformly for testing. The average of the 10 test results for the entire test set test time was considered as the final prediction time.

4.4 Experimental results and analysis

4.4.1 Ablation experiments

To verify the effectiveness of the proposed model or every submodule, we present ablation experiments in this paper.

Table 2 shows the results of the ablation experiments. As listed in Table 2, Model 1(baseline) was the original YOLOv4 network structure. Model 2 replaced the ReLU activation function in Model 1 with the Mish activation function. Model 3 replaced the CIoU loss function in Model 2 with our proposed FioU loss function. Model 4 was model 3 with SemiDSConv and SemiDSCSP. Model 5 was the proposed four-head structure based on model 4 and model 6 was the model in which the SE channel attention mechanism module was embedded into Model 5.

The results showed that both Model 2 and Model 5 have improved performance separately to varying degrees compared to the previous model. In comparison to Model 2, Model 3, which used FIoU loss function, increased the mAP by 4.3%. The proposed Model 4 increased the mAP by 3.7% and also improved the detection speed by about 14 FPS. After embedding SE channel attention into the network, the proposed Model 6 attained the best performance. Compared to the original YOLOv4 algorithm (Model 1), Model 6 's mAP increased from 80.2% to 91.1%, an increase of 10.9%.

It may be noted that the presented model not only reduces the computational cost and improves the detection speed, but also achieves good performance compared to the baseline.

4.4.2 Detection results comparison

To demonstrate the superiority of the proposed method in the detection of underwater targets, we compared it with the original YOLOv4 algorithm and six other methods: YOLOv5, YOLOv7 (Wang CY, et al., 2022), Tiny YOLOv4, YOLO-Fish(Al Muksit et al., 2022), Faster R-CNN, and SSD. All tests were performed on the URPC dataset. The results of these experiments are shown in Table 3.

It can clearly be seen from Table 3 that the presented method has the highest mAP, while the detection speed is faster than the baseline, meeting the demand for real-time detection.

Figure 9 indicates the visualization experimental result of YOLOv4, Tiny-YOLOv4, YOLOv5, YOLOv7, and our method for underwater detection on the URPC dataset. As can be discerned

TABLE 1 Hyperparameter settings.

Training Epochs	Batch Size	Learning Rate	Weight Decay	Momentum	Cosine Annealing
100	8	0.00522	0.00044	0.98	0.114
Translate (Image **Translation**)	Scale (Image Scale)	Fliplr **(Image Flip Left-Right)**	Flipud (Image Flip **Up-Down**)	Mosaic	Mixup
0.0726	0.9	0	0.5	0.932	0

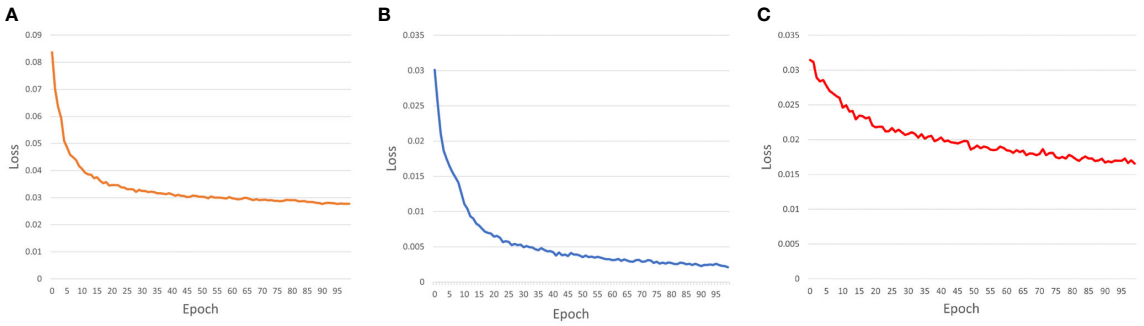


FIGURE 8
The curves of the loss values: **(A)** localization loss; **(B)** classification loss; **(C)** confidence loss.

TABLE 2 Results of ablation experiments.

Model	Method						mAP(%)*	Speed (FPS)
	Baseline	Mish	FIoU	Semi-DSConv	New Head	SE		
Model1	√						80.2	51.2
Model2		√					80.6(+0.4)	49.3
Model3		√	√				84.9(+4.3)	49.3
Model4		√	√	√			88.6(+3.7)	63.4
Model5		√	√	√	√		90.5(+1.9)	58.5
Model6		√	√	√	√	√	91.1(+0.6)	58.1

*The value within the bracket denotes the improvement compared to the previous model

TABLE 3 Experimental results of different algorithms on the URPC dataset.

Method	mAP (%)	Scallop (%)	Starfish (%)	Holothurian (%)	Echinus (%)	Model Size (MB)	Speed (FPS)
YOLOv4	80.2	73.5	87.2	77.7	82.3	204.8	51.2
YOLOv5	80.4	72.9	87.4	76.3	84.8	243.2	44.7
YOLOv7	80.5	73.6	89.7	73.7	85.1	186.0	48.9
YOLO-Fish	77.5	69.1	86.7	71.6	82.6	234.8	45.6
Tiny YOLOv4	63.7	58.5	70.8	56.5	69.0	23.0	114.9
Faster R-CNN	84.4	78.2	93.3	79.1	86.9	419.2	4.8
SSD	61.5	59.3	68.4	56.0	62.2	36.4	72.3
Ours	91.1	86.2	93.2	89.7	95.2	182.7	58.1

from Figure 9, the detection result of our method was better than YOLOv4, and considerably better than the Tiny YOLO v4.

To better demonstrate the detection results of our proposed algorithm with other algorithms, we compared our proposed algorithm with YOLOv5 and YOLOv7 in detail. Figure 10 shows the detection results of the three algorithms. As shown in the figure, the targets marked with red-dashed boxes in the figure have obscure and blurred edges, which are difficult to distinguish from the background, for which our algorithm can still identify and label well. At the same time, many targets underwater are easily misidentified due to the complex environment, and the yellow-dashed boxes in the figure mark the targets that are misidentified by the algorithm. As can be seen, our proposed algorithm has a low false detection rate and is suitable for using in complex underwater environments.

All the experimental results show that our proposed method achieves a good trade-off between detection accuracy and detection speed, which means that it is considered superior for underwater target detection.

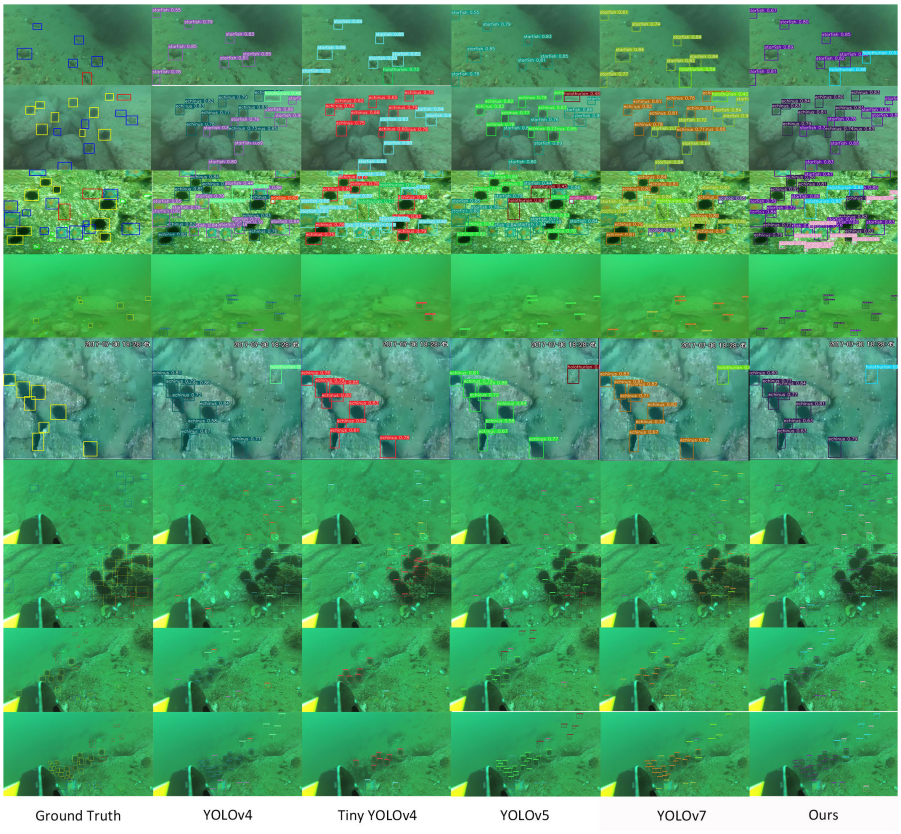


FIGURE 9
Visualization comparison of detection results with YOLO v4, Tiny YOLO v4, YOLOv5, YOLOv7, and ours.

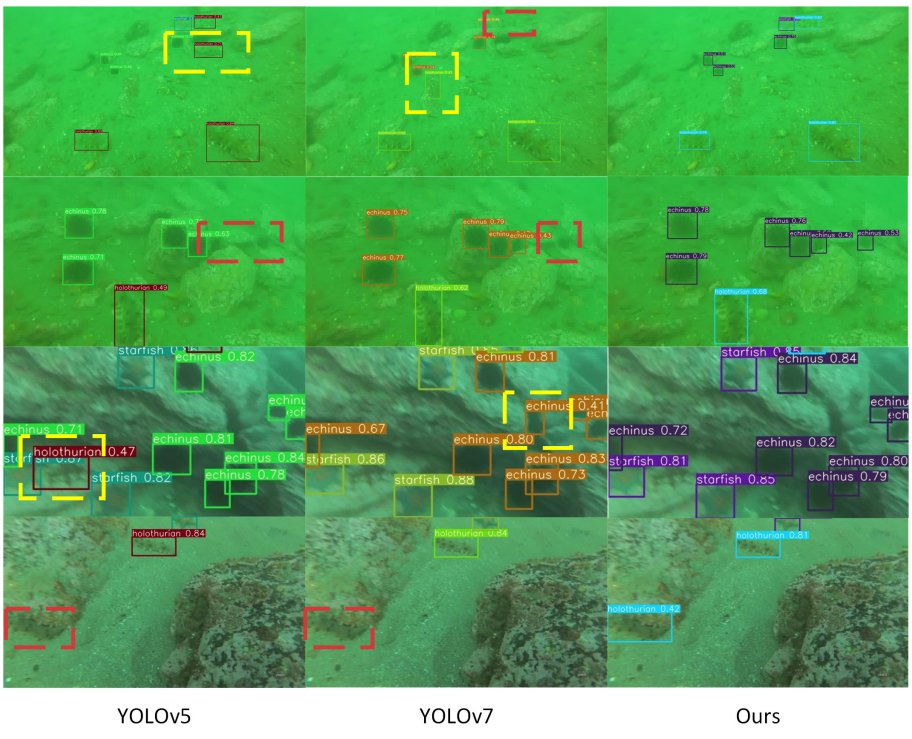


FIGURE 10
More detailed visualization comparison of detection results with YOLOv5, YOLOv7, and ours.

5 Conclusions

Detecting targets with good accuracy and fast detection speed in underwater environments is a challenging problem. In this paper, we presented a real-time underwater target detection algorithm based on improved YOLOv4. In our work, we first developed a new convolutional module and network structure to enhance the feature extraction capability for the model, reduce the computational effort, and speed up the model inferencing. Then, we defined a new IoU loss that improves the target detection performance and the convergence speed of the network. Meanwhile, we optimized the network model and made some other small improvements. We added a new prediction head to handle dramatic changes in the scale of the underwater targets and embedded the channel attention block in the network, which makes the detection and classification of the network more accurate. Experiments show that the presented model achieves 91.1% mAP and 58.1 FPS detection speed on the URPC dataset, outperforming the other listed algorithms in terms of combined performance, which indicates that the proposed model has significant advantages in handling underwater target detection tasks and is more robust in complex underwater environments.

In our future work, how to compress model size to design a more lightweight network and make it applicable to small, embedded devices while maintaining accuracy is an issue that merits further research.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

CZ conceived, planned, and performed the designs and drafted this paper. GZ, HEL, JT and HUL provided guidance and reviewed this paper. XX provided the design ideas and edited this paper. All authors contributed to the article and approved the submitted version.

Funding

This research was funded by National Natural Science Foundation of China, grant number 61863018, and the Applied Basic Research Foundation of Yunnan Province, grant number 202001AT070038.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Akkaynak, D., and Treibitz, T. (2019). "Sea-Thru: A method for removing water from underwater images," in *2019 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 1682–1691.

Arthur, D., and Vassilvitskii, S. (2007). "K-means++ the advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms* (Society for Industrial and Applied Mathematics 3600 University City Science Center Philadelphia, PA United States), 1027–1035.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv*. 10934. doi: 10.48550/arXiv.2004.10934

Cai, S., Li, G., and Shan, Y. (2022). Underwater object detection using collaborative weakly supervision. *Comput. Electrical Eng.* 102, 108159. doi: 10.1016/j.compeleceng.2022.108159

Chen, L., Zhou, F., Wang, S., Dong, J., Li, N., Ma, H., et al. (2022). SWIPENET: Object detection in noisy underwater scenes. *Pattern Recognition* 132, 108926. doi: 10.1016/j.patcog.2022.108926

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20 (3), 273–297. doi: 10.1007/BF00994018

Dalal, N., and Triggs, B. (2005). "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition* (New York: IEEE Press), 886–893.

Felzenszwalb, P., McAllester, D., and Ramanan, D. (2008). "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 1–8.

Freund, Y., and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. System Sci.* 55 (1), 119–139. doi: 10.1006/jcss.1997.1504

Girshick, R. (2015). "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision* (New York: IEEE Press), 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 580–587.

Glorot, X., Bordes, A., and Bengio, Y. (2011). "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. (Brookline: Microtome Publishing). 315–323.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2018). Mask r-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2), 386–397. doi: 10.1109/TPAMI.2018.2844175

Hong, K. T., Abdullah, S. N. H. S., Hasan, M. K., and Tarmizi, A. (2022). Underwater fish detection and counting using mask regional convolutional neural network. *Water* 14 (2), 222. doi: 10.3390/w14020222

Hou, Z., Yu, B., and Tao, D. (2022). "BatchFormer: Learning to explore sample relationships for robust representation learning," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 7246–7256.

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 7132–7141.

Huang, H., Zhou, H., Yang, X., Zhang, L., Qi, L., and Zang, A. Y. (2019). Faster r-CNN for marine organisms detection and recognition using data augmentation. *Neurocomputing* 337, 372–384. doi: 10.1016/j.neucom.2019.01.084

Jiao, W., Cheng, X., Hu, Y., Hao, Q., and Bi, H. (2022). Image recognition based on compressive imaging and optimal feature selection. *IEEE Photonics J.* 14 (2), 1–12. doi: 10.1109/JPHOT.2022.3155489

Kim, K., and Lee, H. S. (2020). "Probabilistic anchor assignment with iou prediction for object detection," in *Computer vision–ECCV 2020: 16th European conference* (Germany: Springer International Publishing), 355–371.

Krizhevsky, A., and Hinton, G. (2010) *Convolutional deep belief networks on cifar-10*. Available at: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=bea5780d621e669e8069f05d0f2fc0db9df4b50f (Accessed February 26, 2023).

Li, P., and Jin, J. (2022). "Time3D: End-to-End joint monocular 3D object detection and tracking for autonomous driving," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 3875–3884.

Li, Y., Wu, C. Y., Fan, H., Mangalam, K., Xiong, B., Malik, J., et al. (2022). "MViTv2: Improved multiscale vision transformers for classification and detection," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 4794–4804.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2018). Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2), 318–327. doi: 10.1109/iccv.2017.324

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., et al. (2016). "Ssd: Single shot multibox detector002E," in *Computer vision–ECCV 2016: 14th European conference* (Germany: Springer International Publishing), 21–37.

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). "Path aggregation network for instance segmentation," in *2018 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 8759–8768.

Loshchilov, I., and Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts. *arXiv*. 03983. doi: 10.48550/arXiv.1608.03983

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60 (2), 91–110. doi: 10.1023/b:visi.0000029664.99615.94

Misra, D. (2019). Mish: A self regularized non-monotonic neural activation function. *arXiv*, 08681. doi: 10.48550/arXiv.1908.08681

Moniruzzaman, M., Islam, S. M. S., Lavery, P., and Bennamoun, M. (2019). "Faster r-CNN based deep learning for seagrass detection from underwater digital images," in *2019 digital image computing: Techniques and applications* (New York: IEEE Press), 1–7.

Muksit, A., Hasan, F., Hasan Bhuiyan Emon, M. F., Haque, M. R., Anwary, A. R., and Shatabda, S. (2022). YOLO-fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecol. Inform* 72, 101847. doi: 10.1016/J.ECOINF.2022.101847

Peng, F., Miao, Z., Li, F., and Li, Z. (2021). S-FPN: A shortcut feature pyramid network for sea cucumber detection in underwater images. *Expert Syst. Appl.* 182, 115306. doi: 10.1016/j.eswa.2021.115306

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: Unified, real-time object detection," in *2016 IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 779–788.

Redmon, J., and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv*, 02767. doi: 10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (6), 1137–1149. doi: 10.1109/tpami.2016.2577031

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). "Generalized intersection over union: A metric and a loss for bounding box regression," in *2019 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 658–666.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). "ORB: An efficient alternative to SIFT or SURF," in *2011 international conference on computer vision* (New York: IEEE Press), 2564–2571.

Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv*, 02696. doi: 10.48550/arXiv.2207.02696

Wang, Z., Li, T., Zheng, J. Q., and Huang, B. (2022). "When cnn meet with vit: Towards semi-supervised learning for multi-class medical image semantic segmentation," in *Computer vision–ECCV 2022 workshops* (Cham: Springer Nature Switzerland), 424–441.

Wang, C.-Y., Mark Liao, H.-Y., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). "CSPNet: A new backbone that can enhance learning capability of CNN," in *2020 IEEE/CVF conference on computer vision and pattern recognition workshops* (New York: IEEE Press), 1571–1580.

Wang, H., Sun, S., Bai, X., Wang, J., and Ren, P. (2023). A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes. [Preprint]. Available at: https://ieeexplore.ieee.org/document/10058092 (Accessed March 15, 2023).

Wang, H., Sun, S., Wu, X., Li, L., Zhang, H., Li, M., et al. (2021). "A yolov5 baseline for underwater object detection," in *OCEANS 2021* (San Diego Porto: IEEE Press), 1–4.

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018b). "CBAM: Convolutional block attention module," in *Computer vision–ECCV 2018: 15th European conference* (Germany: Springer International Publishing), 3–19.

Yang, C., Huang, Z., and Wang, N. (2022). "QueryDet: Cascaded sparse query for accelerating high-resolution small object detection," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 13658–13667.

Yeh, C. H., Lin, C. H., Kang, L. W., Huang, C. H., Lin, M. H., Chang, C. Y., et al. (2021). Lightweight deep neural network for joint learning of underwater object detection and color conversion. *IEEE Trans. Neural Networks Learn. Syst.* 33 (11), 6129–6143. doi: 10.1109/TNNLS.2021.3072414

Zeng, L., Sun, B., and Zhu, D. (2021). Underwater target detection based on faster r-CNN and adversarial occlusion network. *Eng. Appl. Artif. Intell.* 100, 104190. doi: 10.1016/j.engappai.2021.104190

Zhang, Y. F., Ren, W., Zhang, Z., Jia, Z., Wang, L., and Tan, T. (2022). Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* 506, 146–157. doi: 10.1016/j.neucom.2022.07.042

Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *2018 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 6848–6856.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D. (2020). Distance-IoU loss: Faster and better learning for bounding box regression. *Proc. AAAI Conf. Artif. Intell.* 34 (07), 12993–13000. doi: 10.1609/aaai.v34i07.6999

Zheng, Z., Wang, P., Ren, D., Liu, W., Ye, R., Hu, Q., et al. (2021). Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybernetics*. 52 (8), 8574–8586. doi: 10.1109/TCYB.2021.3095305

Zhu, Q., Lei, Y., Sun, X., Guan, Q., Zhong, Y., Zhang, L., et al. (2022). Knowledge-guided land pattern depiction for urban land use mapping: A case study of Chinese cities. *Remote Sens. Environ.* 272, 112916. doi: 10.1016/j.rse.2022.112916

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 04 April 2023
DOI 10.3389/fmars.2023.1150651

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, United States

REVIEWED BY
Abdelouahid Bentamou,
Ecole Des Mines De Saint-Etienne, France
Daniel Marrable,
Curtin University, Australia

*CORRESPONDENCE
Jack H. Prior
jack.prior@noaa.gov
jhp277@msstate.ngi.edu

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 24 January 2023
ACCEPTED 20 March 2023
PUBLISHED 04 April 2023

CITATION
Prior JH, Campbell MD, Dawkins M, Mickle PF, Moorhead RJ, Alaba SY, Shah C, Salisbury JR, Rademacher KR, Felts AP and Wallace F (2023) Estimating precision and accuracy of automated video post-processing: A step towards implementation of AI/ML for optics-based fish sampling. *Front. Mar. Sci.* 10:1150651. doi: 10.3389/fmars.2023.1150651

COPYRIGHT
© 2023 Prior, Campbell, Dawkins, Mickle, Moorhead, Alaba, Shah, Salisbury, Rademacher, Felts and Wallace. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Estimating precision and accuracy of automated video post-processing: A step towards implementation of AI/ML for optics-based fish sampling

Jack H. Prior[1]*, Matthew D. Campbell[2], Matthew Dawkins[3], Paul F. Mickle[4], Robert J. Moorhead[5], Simegnew Y. Alaba[5], Chiranjibi Shah[5], Joseph R. Salisbury[6], Kevin R. Rademacher[2], A. Paul Felts[2] and Farron Wallace[7]

[1]Southeast Fisheries Science Center, Northern Gulf Institute – Mississippi State University, Pascagoula, MS, United States, [2]Southeast Fisheries Science Center, Population and Ecosystem Monitoring Division, National Marine Fisheries Service, Pascagoula, MS, United States, [3]Kitware, Inc., Clifton Park, NY, United States, [4]Stennis Space Center, MSU Science and Technology Center, Northern Gulf Institute – Mississippi State University, Stennis Space Center, MS, United States, [5]Mississippi State University (MSU) Science and Technology Center, Northern Gulf Institute – Mississippi State University, Starkville, MS, United States, [6]Technical and Engineering Support Alliance (TESA) ProTech Contract Company (JV), Rockville, MD, United States, [7]Southeast Fisheries Science Center, Fisheries, Assessment, Technology, and Engineering Support Division, National Marine Fisheries Service, Galveston, TX, United States

Increased necessity to monitor vital fish habitat has resulted in proliferation of camera-based observation methods and advancements in camera and processing technology. Automated image analysis through computer vision algorithms has emerged as a tool for fisheries to address big data needs, reduce human intervention, lower costs, and improve timeliness. Models have been developed in this study with the goal to implement such automated image analysis for commercially important Gulf of Mexico fish species and habitats. Further, this study proposes adapting comparative otolith aging methods and metrics for gauging model performance by comparing automated counts to validation set counts in addition to traditional metrics used to gauge AI/ML model performance (such as mean average precision - mAP). To evaluate model performance we calculated percent of stations matching ground-truthed counts, ratios of false-positive/negative detections, and coefficient of variation (CV) for each species over a range of filtered outputs using model generated confidence thresholds (CTs) for each detected and classified fish. Model performance generally improved with increased annotations per species, and false-positive detections were greatly reduced with a second iteration of model training. For all species and model combinations, false-positives were easily identified and removed by increasing the CT to classify more restrictively. Issues with occluded fish images and reduced performance were most prevalent for schooling species, whereas for other species lack of training data was likely limiting. For 23 of the examined species, only 7 achieved a CV less than 25%. Thus, for most species, improvements to the training library will be needed and next steps will include a queried learning approach to bring balance to the

models and focus during training. Importantly, for select species such as Red Snapper (*Lutjanus campechanus*) current models are sufficiently precise to begin utilization to filter videos for automated, versus fully manual processing. The adaption of the otolith aging QA/QC process for this process is a first step towards giving researchers the ability to track model performance through time, thereby giving researchers who engage with the models, raw data, and derived products confidence in analyses and resultant management decisions.

KEYWORDS

fisheries, machine learning, BRUVS, Maxn, Gulf of Mexico, automation

1 Introduction

Management of fish populations requires estimates of abundance, age/length composition, fecundity, mortality, and other life history variables sampled representatively from a stock (Jennings and Kaiser, 1998). Monitoring efforts are becoming increasingly critical as populations are impacted by multiple stressors such as fishing, climate change, biotic perturbations (e.g., hypoxia), habitat loss, and rising levels of pollution (e.g., microplastics). Historically, resource surveys were conducted using a wide-variety of traditional fisheries gears such as trawls, traps, and nets. Over the past 30 to 40 years, optics-based sampling methods have become a more common practice as they avoid issues with problematic habitats such as reefs, and have fewer issues with size and species selectivity (Cappo et al., 2007). Moreover, optical sampling with BRUVs (Baited Remote Underwater Videos) is less invasive, non-lethal, and can also provide valuable habitat data valuable for single-species and ecosystem-based management (EBM) and ecosystem-based fisheries management (EBFM).

One downside associated with optical sampling is the immense amount of data collected and, in turn, the human effort required to post-process collections (i.e. annotate). For example, one year of sampling of the combined Gulf Fishery Independent Survey of Habitat and Ecosystem Resources (GFISHER) and the Southeast Area Monitoring and Assessment Reef Fish Video (SEAMAP-RFV) surveys results in ~2000 camera deployments, ~1000 hrs of video, and ~30 TB of data requiring annotation (hereafter GFISHER refers to these surveys in combination). Extrapolated across NMFS Science Centers, state agencies, academic laboratories, and non-governmental organizations, the big-data issue quickly becomes overwhelming. In response, the National Marine Fisheries Service (NMFS) funded the Automated Image Analysis Strategic Initiative (AIASI) with the goal of producing software that can be trained on object detection and classification using artificial intelligence/machine learning (AI/ML) across a wide variety of natural resources. A major outcome of the AIASI was the development of the Video and Image Analytics in the Marine Environment (VIAME ©) software in partnership with Kitware Inc. (Clifton Park, NY).

New developments in graphics processing units (GPU) technology and artificial AI/ML processes can provide a means to reduce human effort for post-processing data collected in marine habitats (van Helmond et al., 2020). Frame level count data can be generated using algorithm outputs from which any number of metrics (e.g., MaxN and MeanCount) could be estimated. Among the many advantages to applying algorithms to process data over human video readers are that processing can occur 24/7, detection and identification are standardized to a single algorithm, inter and intra-reader variability is reduced, and computing costs are relatively inexpensive, particularly when considering the efficiencies in post-processing potentially gained. Additionally, features that may be missed by human eyes can be discerned and recognized by computer vision. The GPU-based classifications remain consistent and do not change based on human moods or energy levels. Despite their burgeoning development and promise, questions pertaining to algorithm accuracy and precision remain, particularly those related to sampling conditions that might limit their reliability (e.g., water visibility). This is especially important because long-term time-series require that data annotated using AI/ML is compatible with the human annotations conducted historically. This is critical in cases for which historic video is unavailable for re-processing using AI/ML methods (e.g., non-digital formats, or lost/destroyed video).

When evaluating model performance using a subset of training imagery, AI/ML algorithms have demonstrated excellent performance in detection and classification of a wide-variety of object classes (Zion et al., 2007). Yet analysis of *in situ* collections show less accuracy and precision than is suggested by analyzing precision using a subset of training imagery (Salman et al., 2020). For instance, water turbidity and/or low light intensity may reduce model accuracy and precision (Marini et al., 2018). In addition, videos with increased fish density (i.e., fish/unit area) and higher levels of species diversity may be more difficult for algorithms to process accurately. Rugose habitats of reefs may lead to larger numbers of false negatives/positives in fish detections due to cryptic behavior and/or coloring and mottling that resembles complex habitat (e.g., lionfish). Fish species of different size classes and with different swimming or schooling behaviors may be harder to detect or classify than others (Lopez-Marcano et al., 2022),

especially at variable distances from a camera at a fixed position (mobile cameras face their own challenges). Regardless of the source of error, the main challenge is that the annotation phase of post-processing is likely impacted by detection and identification differences arising from variable environmental conditions in which video is collected, and therefore great care has to be taken to ensure that time-series remain stable relative to changes made in post-processing methods. Put more simply, there are inevitable differences between manual and automated processing that have to be analyzed, evaluated and compensated for if necessary.

A common approach to solving the wide variety of problems associated with using AI/ML for classification and enumeration (e.g., schooling) is to use different model architectures and mathematical algorithms. For instance, convolutional neural networks (CNN) have been shown to produce higher accuracy than older methods such as Support-Vector Machine (SVM) models, Gaussian-Mixture Modeling (GMM), or You-Only-Look-Once (YOLO) based approaches (Cui et al., 2020; Marrable et al., 2022). Fish detection at the frame level has been achieved by many researchers and with relatively high levels of accuracy (Chuang et al., 2014; Villon et al., 2016; Allken et al., 2021); however, tracking an individual across the field-of-view (FOV) by linking detections through multiple frames has been more challenging – especially over the course of extended videos (Ditria et al., 2020). Performance of object detection models is most often evaluated by mAP (Mean Average Precision), receiving operator characteristic, or precision-recall curves, which are usually generated by testing trained models on a fraction of the annotated images (which are not used in training models). Literature review on the topic produced only a single study that compares fish classification performance alternatively to ground truth counts from unannotated video (Connolly et al., 2021). While mAP is a reliable metric for determining performance during training, methods for evaluating performance must be adapted for the practical application and Quality Assessment/Quality Control (QA/QC) of model algorithms. One purpose of this manuscript is to propose an automated workflow that can reliably produce equivalent data to current manual processes and, incorporates accuracy and precision metrics that can be tracked through time as AI/ML models improve or as camera technology changes.

Training AI/ML models to reliably track and classify fish requires manual annotation of each individual detected, per frame, for all frames included in training sets. Creation of the training library in VIAME software can include both still and video imagery and begins with manually drawing boxes around fish targets and labeling the target with an identification (i.e. labeled imagery). Tracks follow individuals over video frames and may include a fish swimming at a constant speed from one end of the FOV to another; however, tracks quite often result in one target passing behind another, passing behind habitat, moving into and out of turbidity plumes, or only partially crossing the periphery of the FOV. Manually annotating these tracks while labeling all species is a time consuming process, but is necessary to ultimately train a comprehensive model which requires lots of imagery for a complex set of fish assemblages, habitats and water conditions. Many studies have achieved high accuracy in performing similar tasks while focusing annotation on few classes of target species (Shafait et al., 2016; Villon et al., 2016; Garcia et al., 2020; Lopez-Vasquez et al., 2020; Tabak et al., 2020; Connolly et al., 2021); however, in high diversity sampling stations, this could lead to a loss of community assemblage data and increased false-positive classifications on fish species that are detected, but not included in the training dataset (Marrable et al., 2022).

In the early stages of the machine learning process, all annotations must be produced manually. This initial annotation necessitates a high cost of effort, but ultimately produces models that have increased ability to perform fish tracking and identification. Once a model can generate annotations with moderate success, it can enter a stage of supervised learning. At this point, human effort can be spent editing the computer-generated tracks rather than manually annotating each individual. Editing includes correcting false identifications and adjusting or deleting bounding boxes that are out of place. Additional editing might be required to split tracks that include multiple fish, or merge tracks where one individual's time in the FOV is incorrectly split up into multiple pieces. In the supervised stage of learning, the rate of new annotations produced for the training library is drastically increased from the manual learning stage, driving the machine learning process faster towards true automation. As automated methods accelerate in the development and uptake, concurrent QA/QC processes must be developed to evaluate outcomes with confidence, which will be necessary when data undergo review for use in stock assessment models.

As image libraries increase in size and complexity between training periods, each new iteration theoretically reduces error and increases agreement relative to validation sets. However, other factors will impact both precision and agreement, and we hypothesize this will likely be a function of site-specific species assemblage, species diversity, optical conditions, fish density, and site complexity. Based on previous studies (Marini et al., 2018; Connolly et al., 2021), it is likely that model counts become less accurate as fish counts increase. It is also possible that the algorithms ability to detect and classify fish will be reduced with increased scene complexity (e.g., complex habitat and fish density) or under less than ideal water visibility conditions (e.g., dark and turbid). The limits at which counts become less accurate are important to discern for practical model implementation because it can be used to determine which datasets models can be trusted for automation, and which datasets still require a supervised QA/QC process in the least. In this study, we seek to report our experience in coming to the supervised learning stage, and evaluate model performance as a function of a variety of precision metrics. This study also proposes developing methods and metrics for comparing model performance using video with known counts (i.e. validation sets in otolith aging), in addition to traditional AI/ML model precision metrics such as mAP.

The primary use of the combined GFISHER data set is to estimate relative abundance for focal species primarily associated with the snapper-grouper complex and as of 2023 has been used to assess 19 species in 28 separate assessments (https://sedarweb.org/). While all three surveys are now combined into a singular design (GFISHER, Thompson et al., 2022), they were historically

conducted under separate survey designs, with identical standard operating procedures and cross trained staffing. Thus implementation of automated image post-processing requires that we understand AI/ML model agreement and precision across multiple laboratories, video annotators, video archives, and data sets. In addition, common precision metrics such as mAP do not appear to be reflective of precision on full-length, high frame-rate videos beyond the domain of the training library. Thus, a method to evaluate agreement and precision will be necessary as post-processing moves to implementation of AI/ML models in vital time-series data.

Currently, manual post-processing of the GFISHER video data sets necessitates a subsampling approach (Thompson et al., 2022) in order to provide timely products for evaluation and use in stock assessments (e.g., relative abundance indices). A wide variety of metrics have been used to convert video observations into datasets used to assess fish and among the more commonly used metrics are MaxN (Ellis and DeMartini, 1995; Campbell et al., 2015), MeanCount (Bacheler and Shertzer, 2015), and time-at-first-arrival (Priede et al., 1994). Ideally, a single automated annotation would provide a dense data set that could be used to generate any metric currently desired. For example both MaxN and MeanCount could be generated from a dataset with frame level identification and counts. Developers for automated processes should not only consider current metrics in use, but also attempt to generate data sets that could be used to create a number of as yet envisioned metrics that are otherwise not possible to generate due to the aforementioned constraints (namely, time).

In lieu of creating an entirely new framework to evaluate accuracy and precision of AI/ML models, we looked to existing structures and methods built for otolith aging (Campana, 2001). Our logic is that counts in a video are akin to counts of annual otolith layers used to age fish. Each read of a video, just like an otolith, should produce similar results across reads and thus also provide a means by which we can evaluate precision. Further, evidence of bias associated with a particular model will have to be dealt with in the post-processing workflow or using analytical approaches (Connolly et al., 2021). We propose here to make use of the analytical approaches reviewed in Campana (2001) to create a QA/QC process to evaluate AI/ML against manually reviewed, ground truth data sets. This will be critical as most AI/ML models show significant improvement with increased size of training image sets (Ding et al., 2017). Therefore there will be a constant need for a thorough QA/QC process so that the resultant time-series data do not risk issues with changing detection (increasing or decreasing), classification, and enumeration capacity. More importantly, if models do show significant drift in those properties, then video archives could be re-run with updated models. Finally, this process should not be confused with validation (Campana, 2001), but rather a way to evaluate and quantify accuracy and precision through time and across laboratories. Further and more complex calibration work will be required to create a validation set (i.e., one that can be used to tune absolute abundance or density estimates). Therefore we use the term validation here to simply refer to the manually processed and QA/QC videos against which precision will be measured.

2 Methods

2.1 Model training

In 2020, VIAME developers, Kitware Inc., deployed the Cascade Faster Region Neural Network (CFRNN; Cai and Vasconcelos, 2018), along with a fish-motion based tracking approach similar to past attempts (Hsiao et al., 2014; Salman et al., 2020; Dawkins et al., 2022). VIAME software was used to manually annotate marine fish species on video data obtained during the combined GFISHER reef fish video survey (Figure 1). Coincidently, in January of 2021, a new version of VIAME (0.13.0), began to employ a two-step process that was used to train model 2.1. The first step includes consolidating tracking data from all labels in a single-class fish detector/tracker (either with motion infusion (m) into the CFRNN training, or as a single-frame classifier (s) with standard CFRNN training). The second step trains object classifiers using each label as an individual class. Models were trained using a 4x system of RTX 6000 GPUs.

For the fully manual training stage (hereafter 'manual') of the machine learning process, we compiled the initial image library with 61.5k frame extractions from 2018 and 2019 surveys with no discrimination towards species or video station locations. Frames were extracted from videos at variable rates from 1 to 10 fps. In March of 2021 software was updated to include interface options to annotate video in addition to single frame imagery - leading to a rapid increase in the amount of annotations compiled in the training set. All annotations included in the training library were produced on videos with frame rates of at least 5 fps. During May of 2021, model 2.3 was developed with a library of 170,000 annotations across 135 classification groups (Figure 1). The data in this model was mostly labeled at the species level, but some classifications are at genus or family levels if identification cannot be determined with greater resolution. Model version 2.3 was deemed capable enough to shift the annotation efforts from the manual process, to a semi-automated process. Following six months of performing corrections on model 2.3 annotations, the training library vastly expanded to its current size at 603,533 annotations across 146 classification groups in order to train model iteration 2.4 (Figure 1).

2.2 Model parameters

Each model package has a set of configurations and pipeline files available that can be modified to optimize performance. To facilitate reproduction of these methods, the following paragraphs describe the model nomenclature and text designations within the configuration files that can be selected or altered for different application purposes.

There are designations for the size of video fed into networks including 0.5x, 1.0x, 2.0x. The 0.5x size processes videos at 640x640 pixels, 1.0 at native input resolution, and 2.0x increases image resolution by a factor of 2 to 2.5. All results reviewed here were generated at the 1.0x scale configuration for all models. The fish tracking pipelines have been created with two different types of models: motion (m) and single-frame (s). The 2.4m (motion) model

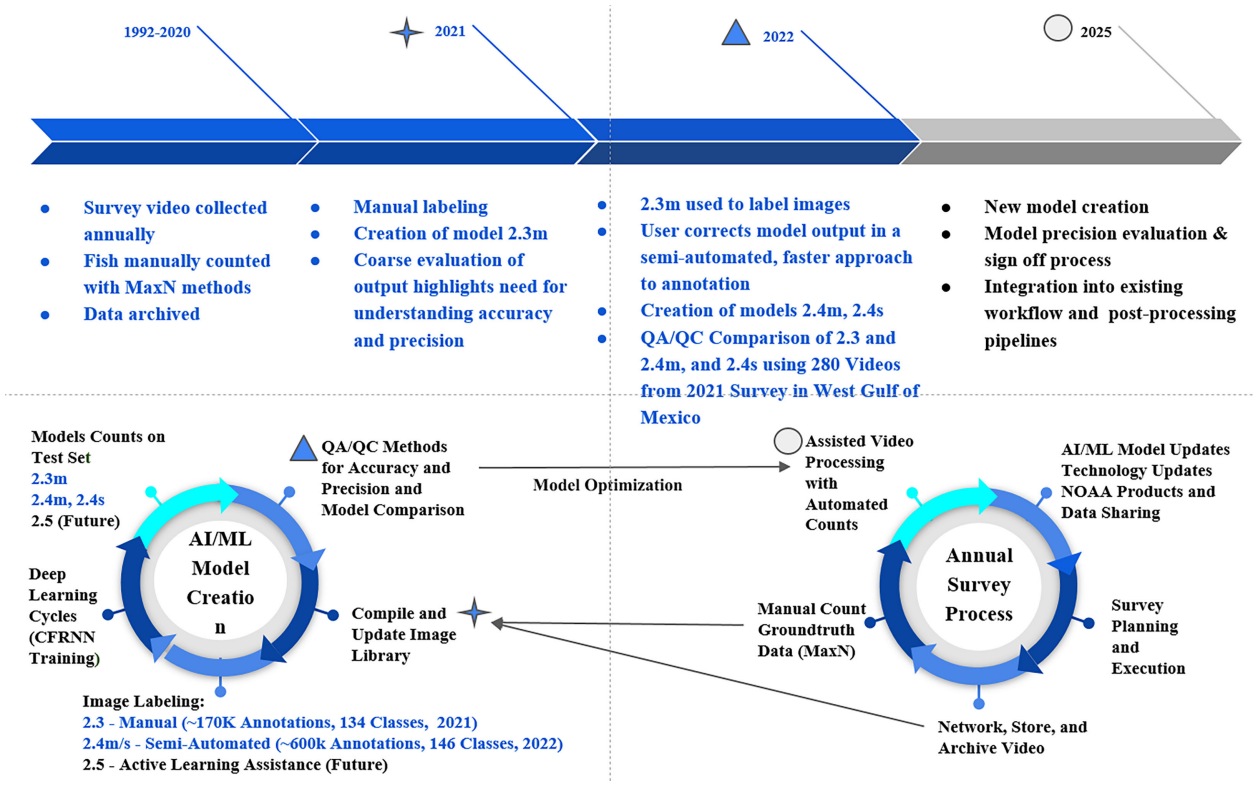


FIGURE 1
Combined workflow of model development and the annual survey process. MaxN manual counts are the maximum number of the same species, observed in the same frame, during a 25-minute station video. Model nomenclature "s" denotes single-frame fish tracking algorithms and "m" denotes motion-based tracking. Models 2.4s and 2.4m are available in a GitHub repository (https://github.com/VIAME/VIAME). These models are also available for public use as embedded pipelines on the VIAME web application.

is an updated version of 2.3m but uses a larger annotation library. Model 2.3 runs a CFRNN across two motion channels and native intensity. Model 2.4s (single-frame) is a single-frame detector (CFRNN without motion training), built on the same library as 2.4m, but across one optical intensity input channel.

All pipelines run two classifier models by default - a 'big' and 'small' classifier, which target larger and smaller fish (measured *via* raw pixel area) for better performance at each, using the 'resnet' or 'resnext' 50 and 101 architectures (He et al., 2016; Xie et al., 2017). Only one classifier is applied for a size dependent detection state. The small fish classifier and big fish classifier are based on the size of annotation boxes with limits that can be adjusted a priori. For all three model iterations compared in this study, the area pivots of positive 7000 and negative 7000 were used as a threshold to discriminate between "large" and "small" fish. This means that, in the pipeline, only one model is applied for each detection state, greater or smaller than 7000 pixels. When the localization area (width multiplied by height) of the bounding box is greater than or equal to 7000 square pixels, the big classifier is used; conversely, when less than 7000 square pixels are used, the small classifier is employed. When under the lower bound of 1000, no classifier is applied and the detection is labeled as an UNKNOWNFISH. The bound of 1000 was also arbitrarily selected, although it should be noted that these detections carry little weight if they occurred on the same track as larger detections. These classifiers were trained on only small and big area input chips, respectively, for improved classification performance in each condition. Model 2.3 employed resnext architecture for both the large and small classifier, while both 2.4 models used resnext101 for the large classifier, but resnet50 for the small classifier.

2.3 Model evaluation

Automated counts from 315, 25-minute, videos from the 2021 GFISHER combined survey were generated using models 2.3, 2.4m, and 2.4s. Videos were annotated at a rate of 5 fps, yielding 7500 frames per video. The 315 videos were selected from stations west of the Mississippi River Delta (-89.5 W). With each object classification, VIAME estimates and provides a confidence value. The confidence score is calculated in eq 1.0:

$$score_t(c) = (b + (1.0 - b) * \frac{\sum_{i=0}^{n} det_i}{n}) * \frac{\sum_{i=0}^{n} det_i * cls_i(c)}{\sum_{i=0}^{n} det_i} \quad \text{Eq 1.0(a)}$$

OR

$$score_t(c) = (b + (1.0 - b) * \frac{\sum_{t=0}^{n} fish_conf(t)}{n}) * \frac{\sum_{t=0}^{n} fish_conf(t) * class_conf(t,c)}{\sum_{t=0}^{n} fish_conf(t)} \quad \text{Eq 1.0(b)}$$

Variables are given as c = the class ID; n = total number of unique localizations along the frames of each track; deti = detection value for a particular state in a track frame i; fish_conf(t) = fish detection value for a particular state in track time t; clsi(c) = classifier value for class c at the track frame i; class_conf(t,c) = classifier confidence value for class c at time t; b = posterior probability that a track is definitely a fish [default = 0.1]. Automated counts at the model confidence thresholds (CTs) of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 0.95 were used to filter VIAME output and then subsequently compared to manually derived and reviewed data sets for 280 stations (hereafter validation set). It is assumed in this analysis that the manual post-processing and estimates of MaxN counts are accurate. However it is important to understand that these are uncalibrated values, and thus our definition of validation set is reliant on this assumption until a field calibration method is devised. We base our analysis, and proposed QA/QC method, from otolith aging models outlined in Campana (2001). Calculations were executed with the FSA Analysis R script developed by Derek Ogle of Northland College (Ogle, 2013). In these calculations our automated counts by multiple models are analogous to age estimations of otoliths generated from multiple reads against the validation set. We calculate the percent of videos with exact agreement, percent of videos within 1 and 2 counts, the ratios of false-positive and false-negative detections, and model coefficient of variation (CV, %). For each increase in CT the number of stations used for calculations is reduced number of stations with 0 automated detections increases. Stations with zero fish detected in automated processing were removed from the analysis so total percent agreement would not be inflated by agreement of zero, given that most species only appear in a fraction of the videos. Species and model specific estimates are calculated at each CT level, for all stations with positive observations of the selected species (i.e. verified by manual post-processing). CV was calculated as illustrated in Campana (2001) and eq. 2.0 below:

$$CV_j = 100\,\% * \frac{\sqrt{\sum_{i=1}^{R} \frac{(X_{ij}-X_j)^2}{R-1}}}{X_j} \qquad \text{Eq 2.0}$$

where Xij is the ith count of the jth number of fish, Xj is the mean count of the jth number of fish, and R is the number of times each fish is counted (in this case 2 – one manual, one automated).

Finally, the ratio of false-positives was determined by dividing the number of stations with automated detections when the species was not present in the ground truth, over the total number of stations where the species was not present as determined by the validation set (proportion of stations with false detections). False-negatives were also determined by dividing the number of stations without automated detections when the species was present in the ground truth, over the total number of stations where the species was present (proportion of stations with undetected species). Correlation (r^2) and slope were also calculated from the linear regression of manual versus automated model run output. Slope was used to evaluate if the linear relationship between manual and automated counts deviated from 1 (i.e. a 1:1 relationship), while correlation was used to evaluate variability about that predicted relationship.

3 Results

We used a combination of false-positive rate (proportion), percent of exact count agreement (%), percent of data within 1-2 counts (%), and model CV (%) to assess model quality per species and provide guidance on confidence filters to apply in post-processing automated output from VIAME when using the models discussed in this paper (Figures 2–5; Table 1). Evaluation of these variables is considered collectively with more weight placed on reducing false-positives, percent of data within 1-2 counts, and model CV. For example, model 2.4s achieved a slightly higher percent agreement than model 2.4m for Vermilion Snapper (*Rhomboplites aurorubens*) at a CT of 0.4, but had a higher rate of false-positives than the similarly performing model 2.4m at a CT of 0.6 – thus 2.4m @ 0.6 was chosen as the optimal model for this species (Figure 6). Given those criteria, we determined that model 2.4s was the optimal model for 13 of the 23 evaluated species

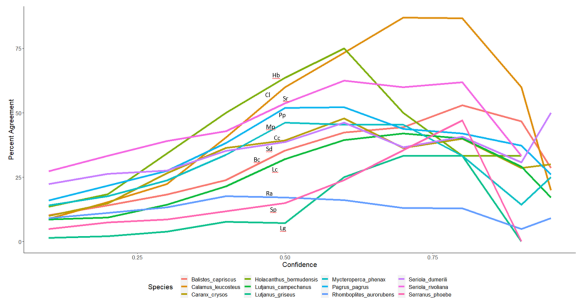


FIGURE 2
Percent Agreement of automated counts with top performing model 2.4s to expert derived counts for twelve commercially and ecologically important species of reef fish commonly observed in the Gulf of Mexico. Lines are labeled with the initials of the species name in the legend. Species with high percent agreement coupled with low false-positives and CV's can potentially filter data with higher confidence values, whereas models with worse performance would use a decreased confidence value to filter data.

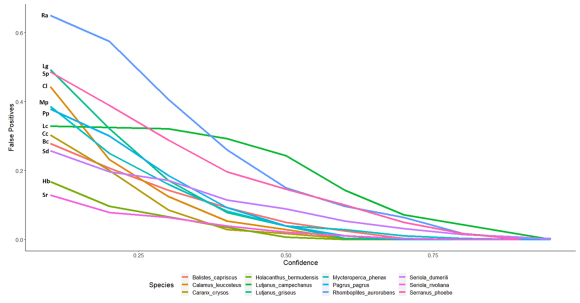


FIGURE 3
False-positive detections of automated counts with top performing model 2.4s to expert derived counts for twelve commercially and ecologically important species of reef fish commonly observed in the Gulf of Mexico. Lines are labeled with the initials of the species name in the legend. The false-positive ratio was determined by dividing the number of stations with automated detections when the species was not present in the ground truth, over the total number of stations where the species was not present as determined by the validation set.

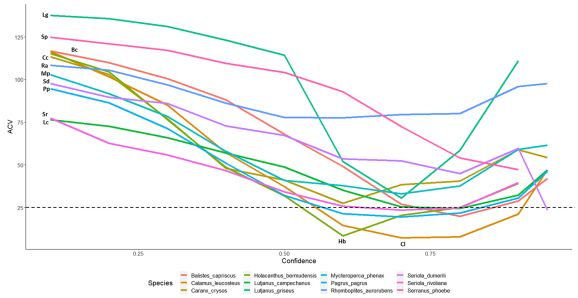


FIGURE 4
CV% for automated counts with top performing model 2.4s to expert derived counts for twelve commercially and ecologically important species of reef fish commonly observed in the Gulf of Mexico. Lines are labeled with the initials of the species name in the legend. Only 7 of 23 species make it below 25% threshold: *B. capriscus, C. leucosteus, H. bermudensis, L. campechanus, P. Pagrus, S. dumerili, S. rivoliana*.

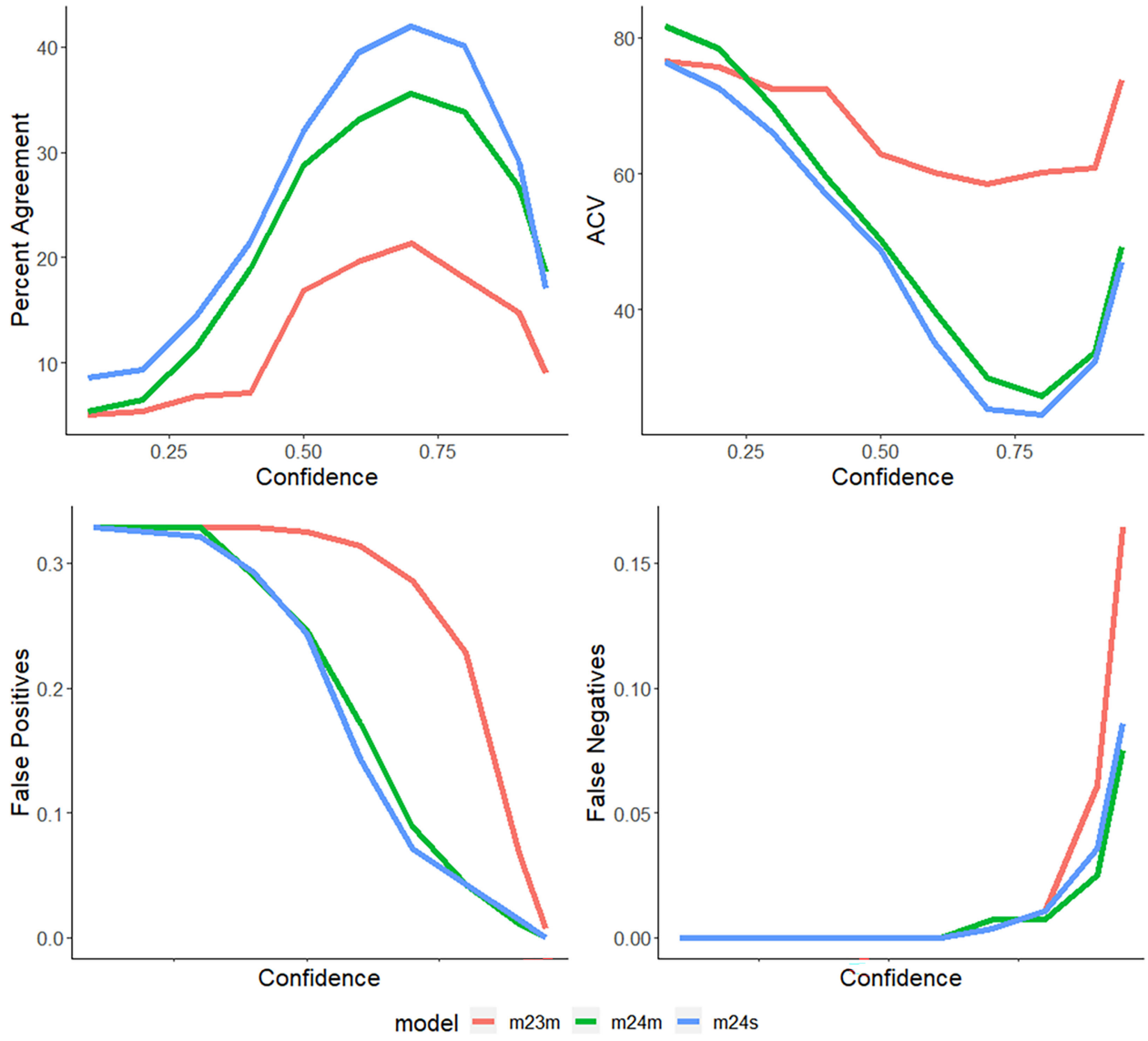


FIGURE 5
Percent agreement, ratio of false-positive detections, ratio of false-negative detections, and CV values across all confidences (0.1-0.95) for models 2.3, 2.4m, and 2.4s for (*Lutjanus campechanus*, the most observed species of the survey) with a maximum percent agreement of 42.03, and a minimum CV value of 24.5).

TABLE 1 Summary of model performance for 23 commercially and ecologically important species commonly observed in the 2021 SEAMAP Reef Fish Video Survey on reef structures along the shelf of the Gulf of Mexico West of the Mississippi River (< -89.5° W).

Species	% Exact Agreement	False- Positive Ratio	CV	% of videos +/- 1 of truth	% of videos +/- 2 of truth	Best Model and CT	r^2 at Best CT
Balistes capriscus	52.94	0	19.97	76.47	100	2.4s @ 0.3	0.612
Bodianus pulchellus	42.11	0.025	59.62	94.74	100	2.4m @ 0.3	0.665
Caranx crysos	47.83	0.004	27.5	73.91	78.26	2.4s @ 0.6	0.748
Calamus leucosteus	86.96	0	7.17	95.65	100	2.4s @ 0.7	0.454
Calamus nodosus	55.56	0.004	26.34	88.89	100	2.4m @ 0.4	0.333
Chaetodon sedentarius	70	0	12.26	100	100	2.3m @ 0.5	0.238
Haemulon aurolineatum	38.89	0.011	52.52	66.67	77.78	2.4s @ 0.6	0.664
Holacanthus bermudensis	75	0	8.42	100	100	2.4s @ 0.6	0.781
Lutjanus campechanus	42.03	0.071	25.39	79.22	85.99	2.4s @ 0.7	0.944
Lutjanus griseus	33.33	0	30.55	33.33	33.33	2.4s @ 0.7	0.969
Lutjanus synagris	100	0	0	100	100	2.4s @ 0.7	–
Mycteroperca interstitialis	100	0	0	100	100	2.4s @ 0.5	1
Mycteroperca microlepis	16.67	0.018	117.9	100	100	2.4s @ 0.5	–
Mycteroperca phenax	47.62	0.007	26.37	80.95	92.86	2.4m @ 0.7	0.385
Pristipomoides aquilonaris	5	0.064	96.11	37.5	45	2.4m @ 0.6	0.174
Paranthias furcifer	16.67	0	65.72	50	50	2.4m @ 0.3	0.268
Pagrus	52.17	0.011	21.34	95.65	98.55	2.4s @ 0.6	0.741
Pterois	4	0.082	132	100	100	2.3m @ 0.8	–
Rhomboplites aurorubens	17.02	0.082	75.54	54.26	63.83	2.4m @ 0.6	0.396
Stenotomus caprinus	14.29	0.018	68.56	71.43	100	2.4s @ 0.5	0.848
Seriola dumerili	50	0	23.57	100	100	2.4s @ 0.95	–
Seriola rivoliana	69.57	0	14.21	86.96	95.65	2.4m @ 0.7	0.749
Serranus phoebe	50	0.021	56.31	94.44	100	2.4m @ 0.7	0.46

(Table 1). For 8 species, model 2.4m was optimal. Model 2.3m, performed better for the remaining two species. In general, model performance was greatly improved from model iteration 2.3 to 2.4 with both fish tracking methods and across most species. In contrast, cryptic Lionfish (*Pterois* sp.), the 2.4 models greatly reduced the amount of high-confidence false-positive detections. As a pattern for most species, counts were overestimated at low CTs, maximum percent agreement was achieved for CTs between 0.3-0.7, and counts were underestimated at high CTs (0.8-0.95). At the CTs showing maximum percent agreement, most of the species were undercounted, suggesting that the models tend to make conservative estimates in comparison to the validation set as CTs become more restrictive. This outcome is heavily influenced by applying more restrictive filter criteria (increased CT) because the sample available to analyze the data is reduced by definition (i.e. high CTs reduce detections and thus sample sizes to conduct analyses).

Figure 2 displays the percent agreement curves for model 2.4s counts across 12 of the most frequently observed species and are representative of diverse groups of fish. As most automated counts are initially overestimated, the ratio of false-positives is also greatest at low CTs and decreases with increasing thresholds and as low

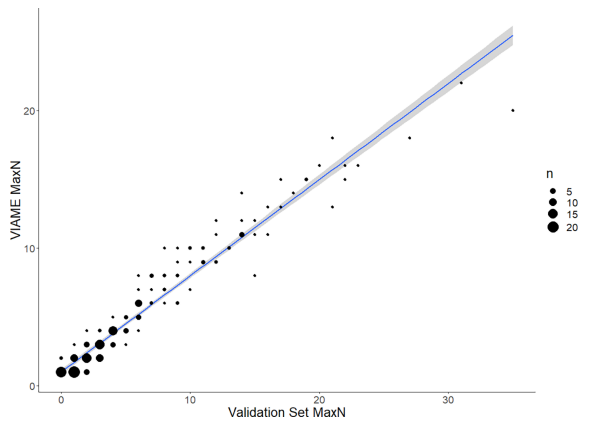


FIGURE 6
Comparison of automated counts at the top performing model (2.4s) and confidence threshold (0.7) for Red Snapper, *Lutjanus campechanus* (the species with the most observations across all stations of the survey). Total agreement sample size (n) at confidence of 0.7 was 207. Residual standard error is 1.451 on 205 df. Linear regression is VIAME MaxN = Validation Set MaxN *(0.7) + 1 with r2 of 0.9439. F-statistic is 3450 on 1 and 205 df, p = 2.2e-16. 79.22% of counts within 1, and 85.99% were within 2 of validation set counts.

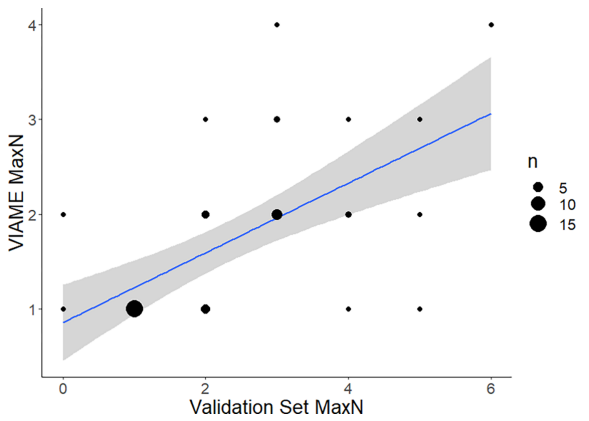


FIGURE 7
Comparison of automated counts at the top performing model (2.4m) and confidence threshold (0.7) for Scamp, *Mycteroperca phenax* (a commonly observed grouper species in the Gulf of Mexico). Total agreement n at confidence of 0.7 = 42. Residual standard error is 1.64 on 40 df. Linear regression is VIAME MaxN = Validation Set MaxN * (0.37) + 0.86 with R2 of 0.3847. F-statistic is 25.01 on 1 and 40 df, p-value is 1.81e-5. 80.95% of counts within 1 of ground truth, 92.86% within 2 of ground truth.

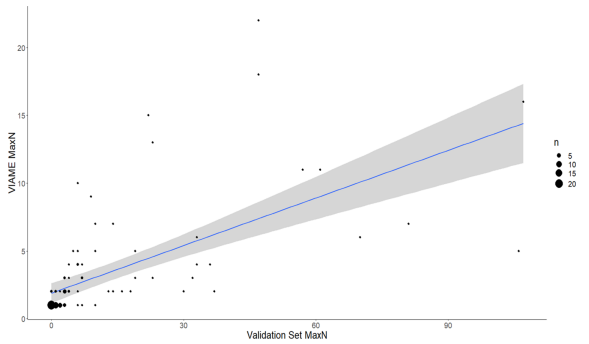


FIGURE 8
Comparison of automated counts at the top performing model (2.4m) and confidence threshold (0.6) for Vermilion Snapper, *Rhomboplites aurorubens* (a commonly observed snapper species in the Gulf of Mexico). Total agreement n at confidence of 0.6 = 94. Residual standard error is 16.8 on 92 df. Linear regression is VIAME MaxN = Manual MaxN (0.12) + 1.9 with R2 of 0.3963. F-statistic is 60.4 on 1 and 92 df, p-value is 1.073-11. 54.26% of counts within 1 of ground truth, 63.83% within 2 of ground truth.

confidence identifications become filtered out (Figures 3, 5; Table 1). False-negatives were much less common than false-positives, but occur at a higher rate at high CTs. Whitebone Porgy (*Calamus leucosteus*) was the species that achieved the highest percent agreement (86.96) and lowest CV value (7.17), while reducing false-positives to zero. Some maximum percent agreements are reported as 100% (Table 1), however caution should be made in these interpretations as sample size is greatly reduced when using CT to filter out low confidence detections. While there is limited performance in percent exact agreement, automated counts for almost all species were within 1 of true counts for at least 50% of stations where the species was detected.

Strength of the linear relationship between automated and manual counts (r^2) varied by species (0.2-0.9) and improved with increased observations in the data (Figures 6–8; Table 1). Correlation between automated and validation set counts was dependent upon the number of observations in the data set, site specific fish density, and life history patterns. For instance, Red Snapper showed high proportions of positive observations and yielded a strong enough correlation for symmetry tests to be conducted (Automated MaxN = Validation Set MaxN *(0.7) + 1, r^2 = 0.9439). Results of that analysis indicate decreased reliability at sites with species specific counts >10. Thus model accuracy deteriorates with increasing site abundance, low count values were always more accurate, and most of the variability is contained to those high count values. For species with low counts, accuracy issues have less to do with site specific abundance and more to do with the training model itself. Scamp (*Mycteroperca phenax*), show weaker correlation than Red Snapper (Automated MaxN = Validation Set MaxN *(0.37) + 0.86, r^2 = 0.3847), however have ~193k fewer annotations in the library (Figures 7, 8; Table 2). For all species, the slope of these best-confidence regression lines are less than one, which is an additional indication that the models conservatively undercount fish (a perfect model would have a slope = 1). Thus models would likely be less sensitive to increases in abundance depending on the frequency of high counts in the database.

Increased annotations used to train models resulted in increased accuracy and precision in most cases; however there are species-specific complexities that confound results (Table 2). For example, while a 180% increase in annotations led to a strong increase in percent agreement and reduction in false-positives for *C. leucosteus*, model performance does not improve similarly in cryptic and schooling species. A 266% increase in annotations only resulted in a 1.76% improvement in maximum percent

TABLE 2 Count of annotations per species that contributed to the training library for each model and the difference in maximum percent agreement between iteration 2.3 and 2.4.

Species Classification	Number of Annotations			Difference in Max % Agreement
	2.3 Count	2.4 Count	% Increase	
Lutjanus campechanus	32440	206452	536.4	20.7
Pagrus	15625	25303	61.9	19.45
Mycteroperca phenax	7932	13062	64.7	28
Pristipomoides aquilonaris	6462	9749	50.9	2.3
Rhomboplites aurorubens	6171	27439	344.6	3.64
Mycteroperca microlepis	5032	7836	55.7	14.2
Seriola dumerili	4519	6416	42	14.3
Serranus phoebe	3055	8299	171.7	16.7
Calamus nodosus	2941	11639	295.7	22.2
Balistes capriscus	2939	12968	341.2	14.7
Calamus leucosteus	2883	8099	180.9	53.6
Holacanthus bermudensis	2404	8256	243.4	57.1
Seriola rivoliana	2228	5505	147.1	9.6
Chaetodon sedentarius	2004	8306	314.5	-7.5
Lutjanus griseus	1902	11311	494.7	33.3
Pterois sp	1606	5878	266	1.76
Caranx crysos	1571	3747	138.5	35.3
Haemulon aurolineatum	1046	7821	647.7	29.6
Mycteroperca interstitialis	989	1016	2.7	83.3
Bodianus pulchellus	967	1032	6.7	24.4
Lutjanus synagris	642	3452	437.7	75
Paranthias furcifer	480	480	0	0
Stenotomus caprinus	12	27975	233025	14.3

agreement for Lionfish. While model 2.4s could achieve 4.76% agreement for *Pterois* at a CT of 0.2, this was not selected as the best option, because low CT resulted in more false-positives than the best 2.3 model (which tracked 4% agreement at a confidence of 0.8). The smaller, fast-moving, and denser schooling species such as Wenchman (*Prisitpomoides aquilonaris*) and Vermilion Snapper (*R. aurorubens*) both had substantial increases in the number of annotations, but achieved less than 4% increases in percent agreement despite the massive increase in annotations used to train the models (Table 2). Model counts for Vermilion Snapper also produced poor linear regression fits (Automated MaxN = Manual MaxN *(0.12) + 1.9, r^2 = 0.3963; Figure 6).

4 Discussion

Our efforts to create automated, fish detection and classification algorithms, has highlighted the importance of understanding accuracy and precision using methods that analyze field-collected video against ground-truthed video collections as a complement to methods such as mAP that evaluate a subset of training data. Ideally this would be accomplished using a calibrated validation set but this level of understanding remains elusive at present. Estimation of accuracy and precision of AI/ML models is a crucial step towards their implementation and integration into existing post-processing frameworks because continuity of time series is critical for use in stock assessments. For instance, stock assessment models can now incorporate time varying catchability (Wilberg et al., 2009), and thus if a technology changes catchability (e.g. AI/ML catches things humans do not), abundance estimates have to be able to measure and compensate for that effect. Critically, current manual methods have been vetted *via* thorough review in assessment or publication outlets, and thus any automation of post-processing will have to be validated and precision metrics tracked through time, including estimates from historic video archives. Critically, this study assumes that human annotation produces accurate data, but the manual counts should not be treated as a calibrated set.

We demonstrate that model performance largely depended upon the number of classification specific annotations used in model training, fish density, and the incidence of various behaviors (e.g., schooling). Regardless of model iteration and application of a confidence filter on the data, model variability increased with increasing number of fish observed. This effect of decreasing precision with increased abundance is particularly pronounced for schooling or shoaling species of fish (e.g., Vermilion Snapper). Cryptic and small fish (e.g., Lionfish, Butterflyfish) were also problematic as they look very similar to the habitat and are often not detected, presumably because the algorithm believes them to be background (e.g., soft coral). Regardless of the underlying source of error, the method we propose here provides researchers with defined metrics to track model performance as a standard component of post-processing video data sets, will help external researchers evaluate model utility for other projects, and suggests species specific output filters for current SEFSC-VIAME models. We believe the current precision of our best model (2.4s) allows for implementation of a semi-automated approach to post-processing by pre-filtering low complexity videos (e.g., low abundance) for full automation and light QA/QC, versus those that will require more intensive manual processing. Thereby we can more efficiently direct manual annotation efforts, reduce time needed to generate usable data sets, and reduce potential effects of reader bias.

Mean Average Precision (mAP) is a standard metric for gauging model precision and is calculated by withholding a portion of the training set against which precision is estimated (Padilla et al., 2020). Efforts using a portion of the dataset (the library for iteration 2.2) reported a mAP50 value of ~70% for detection precision and achieved ~70% for top-class accuracy (Boulais et al., 2021). For model 2.4 detection precision was reported with a mAP50 of 79% for 2.4s, and 74% for 2.4m (supplementary 1). Our analysis clearly shows that additional metrics such as percent agreement, ratio of false-positive detections, and CVs, are necessary for understanding accuracy and precision of models run on naive videos as opposed to evaluation of a subset of training data. Further, these metrics are likely more valuable for implementation of automated methods for post-processing critical time-series survey data as they provide direct inference to performance against existing reads that can be thought of as validated annotations. This is especially true for generating count data for long term time-series containing long-length, high-resolution, and high frame-rate videos. We believe this because mAP scores are based on a selected level of intersection of union (IoU) between frames, and are therefore considered a measure of frame-based precision, rather than precision over the course of a video relative to counts (i.e. abundance). A high mAP, may not be indicative of a models capacity to produce accurate count estimates from novel unlabeled video sets (i.e., annual survey collection). Recent review of fish detection and monitoring methods (Barbedo, 2022) highlights the need for a standardized measurement of accuracy and precision between different models working in different applications, and especially the need for doing so with large sets of unlabeled data that represent natural conditions. This step towards standardization is ultimately necessary to build trustworthy models that can emulate humans in surveys and practical situations.

One of the more obvious results was that increases in training library size, and specifically to class specific annotations, resulted in improved model performance in general and within classes. Although sample size does generally increase model performance the resultant datasets can be imbalanced in the direction of ubiquitous species, an issue known as longtail distribution (Cai et al., 2021), and which is evident in the training library used in the set of projects dealing with this data set (Table 2, Boulais et al., 2021; Alaba et al., 2022). The longtail problem arises naturally from the imagery as ubiquitous species are frequently observed, and thus labeled, even from frames in which more rare species are being targeted. While the improvements to the models can be significant, those gains may not benefit all classes included in a model. In contrast, uniquely mottled and/or shaped taxa (e.g., Sheepshead – *Archosargus probatocephalus*) generally required fewer annotations to generate reasonable models than for species with conspecifics that share similar appearance (e.g., Scamp –*Mycteroperca phenax* and Yellowmouth Grouper – *Mycteroperca interstitialis*).

An approach to dealing with the longtail distribution problem is continued development and integration of active learning algorithms into the training process. Active learning algorithms include output that directs training towards the most important classes to add to the annotation library on which models are trained. Thus creating a focused training for species with fewer annotations and introducing better balance to the training set. Human supervision combined with active-learning algorithms can begin to produce true artificial intelligence systems that recognize what is not understood by the neural network and can autonomously generate new classes for the training library (Lv and Dong, 2022). Further discussion is required to determine whether there could be a longtail bias, based on this distribution of the annotation library, or if such bias should be integrated into model training since it is part of the natural system (Alaba et al., 2022). The fact that Red Snapper has the highest rate of false-positives of any species at the optimal CT (Table 1, Figure 3) may be evidence of longtail bias. Recent efforts (Dawkins et al., 2022) combined several large annotation datasets, including the annotation library used for iteration 2.4, to train an improved and versatile tracking model in VIAME. Following another round of library growth and training with these foci, model performance can again be compared to gauge improvements, along with any alternative architectures or competing model developments. For example, mathematical changes could be made to replace the fish detection output score, with a dedicated classifier which asks how well the fish is showing (i.e., a score given to each fish detection based on quality of the image in terms of the number of pixels and the fish orientation to the camera). The detector output is currently used as a surrogate because its score likely has some correlation with how well the fish is displayed, even though it wasn't created explicitly for that purpose. Many other adjustments to parameters can be tested within the current model configurations due to the versatility of the VIAME software as a machine learning application. Capabilities currently exist to estimate lengths of fish and ongoing studies are using AI/ML for otolith age/length indices. Eventually combining these systems will lead to the future of AI/ML based governance in fisheries management. Given the increased performance of model 2.4s from

2.3, there should be a reduced cost in supervised correction effort, and therefore a more efficient path to a more proficient model 2.5 (Figure 1).

For some schooling and cryptic species, increasing the number of annotations in the training library was not entirely effective. For instance in the case of Vermilion Snapper the training library was increased 344% (Table 2), but model performance showed high variability, low percentage of exact counts, and high model CV (~75%, Table 1). Despite increased annotation, there was minimal improvement for Lionfish classification. We hesitate to speculate on the reasons for variable performance improvement with annotation increases, nor can we suggest methods to deal with this problematic bias, but challenges with high abundance obviously translates to issues for schooling species. The first suggestion is that knowing this bias, we can use this in a similar way to the VIAME generated CT data, to filter out videos for automation versus those that require more intensive supervision or a completely manual process. For instance if initial post-processing indicates a high number of tracks for Vermilion Snapper, we would pull that video for intensive QA/QC or fully manual processing. In all cases in which we see this kind of effect the frequency distribution of high-density sites indicates that these tend to be rare occurrences, and thus filtering in this fashion will result in decreased annotation time and effort. Recent efforts to mathematically deal with this issue were presented in Connolly et al. (2021). Another approach would be to train models to detect schools and create software functionality that would subset the portion of the image with the school to estimate a count (Li et al., 2022). Regardless of the approach taken there will be an obvious need to understand model performance especially at high abundance sites.

VIAME model output includes classification confidence information (i.e. CT) which can be used to filter model output and thereby optimize workflows by decreasing post-processing effort. The value of the CT itself is not used to determine model performance, but it may be important for gauging performance between models that will likely use increased training library sizes (2.3 vs 2.4), or with different training parameters (2.4m vs 2.4s). For instance, if the best confidence for a class is at 0.5 in model 2.3, but 0.7 in model 2.4, then that could be indicative of model improvement. Critically we observed that we can easily reduce false-positives by increasing confidence filters even in the worst performing models. These false-positives were common in model 2.3 and were often associated with clouds of turbidity (Figure 9), debris, parts of the camera array, and habitat structures. Whereas the incidences of false-positives were greatly reduced in model 2.4, likely as the result of improved training and better background identification. Thus, our method provides a general framework for fine-tuning VIAME output generated using the SEFSC models we presented in our analysis and that are hosted online (https://viame.kitware.com/#/root), as a tool to assist human readers in producing accurate counts and reduce post-processing effort (Table 1). Critically, the CT filter enables video annotators to focus on conditions and species that require more intensive review. For instance videos with few individuals and/or with high confidence species could be processed using automated methods and follow up quality control processing. Species with high percentages of automated counts within 1 of true counts will require minimal QA/QC compared to those with lower percentages. In contrast, models with high abundance and/or low confidence species would require a semi-automated approach with

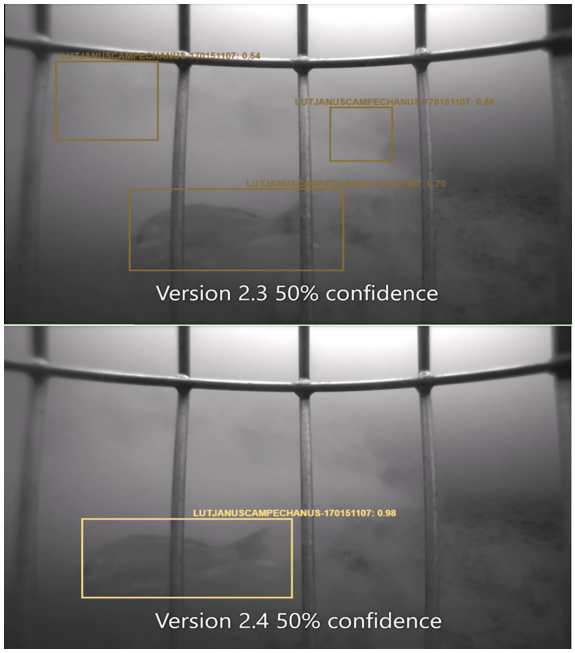


FIGURE 9
Example of reduction of false-positives and increase in confidence of detection and identification (Station 762101220 as turbidity plume clears the field of view). "50% confidence" in figure refers to the confidence threshold of the model.

an intensive manual QA/QC process. Importantly for Red Snapper, the 2.4s model may reliably provide automated counts for stations in the West Gulf up to a MaxN of 10 fish. Of the 280 stations evaluated, 244 had counts <11 Red Snapper. Thus if a request were made for red snapper data we could reliably automate ~90% of the reads, leaving manual annotation to the remaining 10% plus a full QA/QC process to complete for all annotations. Further training and testing of larger sample sizes is required to establish reliability limits for other species. We anticipate as model performance improves through time, annotation speed will increase due to a reduction in effort during the quality control process.

There are also benefits for a reader viewing the low confidence VIAME detections, as sometimes the AI/ML algorithm is better at detecting minute differences, or was trained over a range of augmented orientations and shades simultaneously that enables classification on characters that a human may not have seen or be tuned to recognize. In cases where specific classification is not necessary VIAME has a general fish detector that can be helpful for generating counts and for visualizing individual fish. We believe methods outlined here will provide researchers a consistent and robust method by which model performance can be evaluated as technology, both on the camera and algorithm sides, continues to improve. Importantly, this approach provides a method by which future model performance can be gauged. In the case that ecosystem based management processes require improved assemblage data, the automated methods provided here would offer precision metrics that are invaluable in calibrating and tuning ecosystem models. Moreover, the proposed methods here for a QA/QC process could be adapted to any type of machine learning model development in the future, and could be beneficial both inside and outside of fisheries research to ensure globally cooperative systems of trustworthy AI.

Future efforts for model improvement must include increased annotation for species that demonstrated high levels of misclassification rates, decreased matching to exact counts, and increased CV values. Methods that bring balance into the training model are therefore needed such as the queried learning or longtail alleviation approaches mentioned earlier (Alaba et al., 2022). Conversely, effort should not be expended on increasing annotations for species with associated high precision models. Many observed species have low levels of percent agreement and high levels of false-positives, whereas many others have not yet been annotated in the training library. Thus a deliberate analysis that highlights those species is needed to help direct efforts to improve the image library itself. At times 'handoffs' occur when one or more fish cross paths and causes track identity to switch among individuals (i.e. more than one individual included in a single track). This can result in misclassification to the wrong species which we hope to address with the global tracking model. Dense schools have not been annotated and represent a gap in the annotation library. Schools of baitfish (e.g., Scad – Figure 10), even smaller than Vermilion Snapper, will likely require alternative annotation methods that allow for density estimation rather than individual tracking. Other issues such as gaps in tracks, double-boxing of single individuals, and single-boxes on multiple individuals can also occur but are mostly nuisances and should reduce as software and algorithms improve. Automated workflows show promise in these early phases of development, but for many of the reasons highlighted here it is our opinion they will always require some variety of human oversight, thus frameworks that include model metadata and performance against validation sets need to be developed in concert with the algorithms themselves.

Accuracy and precision present significant hurdles for the implementation of automated processes, but nearly as important will be realizing the benefits of automation in reduced annotation time. The track-based annotation and modeling can provide more accurate identifications because they are derived from multiple frames strung together to create a majority-vote classification over many frames. A single correction of a track, corrects all annotations associated with the track in a single pass and the end result is decreased post-processing time. Using this method increases the number of images, fish angles, and light conditions used to classify fish, and therefore is theoretically increasing classification agreement. It is also beneficial from a memory-cost standpoint. One 25-minute video, which is 7500 frames at 5 fps, is compressed to 1.17 GB (camera specifications from this survey) but when extracted as 7500 individual PNG files, it amounts to 9.84 GB. This reduction in memory is due to the ability to exploit correlation between frames in storing video. (Jain, 1989). Critically, because VIAME produces frame level counts and identifications, any current metric in the literature can likely be produced (e.g.,

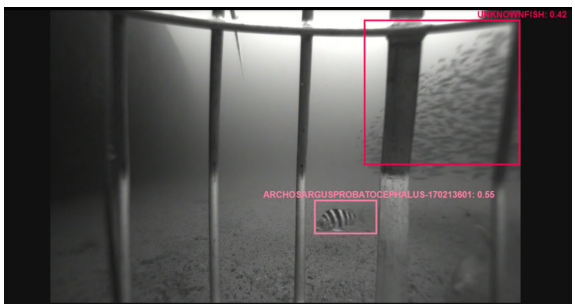


FIGURE 10
Example of a successful detection of a Sheepshead (*Archosargus probatocephalus*) with juxtaposition to the breakdown of performance with large schools of small, less distinguishable fish (Scad).

MaxN, MeanCount, Time-At-First-Arrival, etc). This will have the additional benefit of facilitating analysis on the use of the various derived metrics and perhaps others not yet conceived.

Our translation of the otolith aging methodology for use in estimating the accuracy and precision of automated image analysis models shows promise as a means to ensure data quality for time-series creation and for both existing and anticipated data analysis needs. Model precision and agreement varied by species, number of annotations used in the training set, and only slightly by choice in tracking model (motion or single frame). CV comparisons have historically been acceptable up to around 10% in the otolith aging literature (Campana, 2001). Few of the CV values of the presented species with acceptable sample sizes fall in to this acceptable range in this analysis – only 7 of 23 species have CVs less than 25%, and only 2 are less than 10% (Figure 4). However, for this new application of these quality control methods, it must be decided if those are applicable in this example or determine what level is acceptable. There is a significant amount of investigation still needed on this topic, but we believe the framework presented here is a good first step towards establishment of best practices for integration of automated image post-processing into existing standard-operating-procedures.

5 Conclusion

Advanced technology, in particular miniaturization of computing and sensors, is providing researchers with data and insights into marine systems that were previously inaccessible. These technological advancements are both a boon, in that enormous amounts of data can be collected, but simultaneously present significant bottlenecks often due to being limited to manual post-processing methods (i.e. most data is in storage). Therefore, it is clear that AI/ML will be a significant component of marine laboratory toolkits to help facilitate post-processing necessary for further analysis and optimal use of datasets. This is particularly important in situations for which data timeliness is an important consideration for management decision making processes. Our experience over the course this investigation is that AI/ML has shown significant progress in utility, enough that we believe their integration into post-processing pipelines is a logical next step in the near future (e.g. 5-10 years). Our advice for researchers interested in deployment of AI/ML in optical post-processing is to develop accuracy and precision metrics in concert with the models themselves. This step is critical as many iterations of models can be simultaneously developed, but for their proper deployment their effectiveness has to be measured objectively. Our method presented here offers a way to judge model performance by evaluating model accuracy and precision against ground-truthed video sets. The method assumes a linear relationship between ground-truthed and automated counts and thus we have a simple model by which we can evaluate bias and drift as annual collections are analyzed and new versions of AI/ML models are developed. While the future is bright there remains significant hurdles associated with cryptic, schooling species, and with those having similar looking conspecifics. Some problems are likely going to be resolved by increasing the number of class specific annotations for rare species (e.g. gag) and bringing balance to training libraries, whereas solutions for schooling species are not as obvious and are potentially a limit of the technology. In addition to implementing model QA/QC protocols, programs that are looking to integrate automation into post-processing pipelines should also look to build equivalent manual data sets over an overlapping period of time to evaluate conservation of important time-series data.

Author's note

MD does not imply a NOAA endorsement of Kitware or products and services related to the VIAME software.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://github.com/VIAME/VIAME - The Git Hub page for VIAME software. Reference: SEFSC 100-200 Class Fish Models, All OS. Full video datasets and count data are available upon request.

Author contributions

JP, MC, and MD contributed to conception and design of the study. JP, MC, and JS organized the database. JP, JS, AF, and KR performed expert video annotation and ground truth reviews. JP and MC performed the statistical analysis. JP wrote the first draft of the manuscript. JP and MC wrote sections of the manuscript. All authors contributed to the article and approved the submitted version.

Funding

Funding was provided through Federal Marine Fisheries Initiative (MARFIN) Grant 20MFIH002 through NOAA's National Marine Fisheries Service and the associated award NA21OAR4320190 to the Northern Gulf Institute from NOAA's Office of Oceanic and Atmospheric Research, U.S. Department of Commerce. At sea data collections were funded by Southeast Fisheries Science Center base funds, and the RESTORE act funded Gulf Fishery Independent Survey of Habitat and Ecosystem Resources (GFISHER) grant number NA19NOS4510192.

Acknowledgments

The first author would like to acknowledge the efforts of all other authors in guidance to this point. The data could not be analyzed

without having first been collected by NOAA research vessels *Southern Journey* and *Pisces*, and all those who contributed to the at-sea surveys. In addition, this study would not be possible without the continued technical support from VIAME software developers.

Conflict of interest

Author MD was employed by Kitware, Inc. Author JS was employed by Technical and Engineering Support Alliance (TESA) ProTechContract Company (JV).

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1150651/full#supplementary-material

References

Alaba, S. Y., Nabi, M. M., Shah, C., Prior, J., Campbell, M. D., Wallace, F., et al. (2022). Class-aware fish species recognition using deep learning for an imbalanced dataset. *Sensors* 22 (21), 8268. doi: 10.3390/s22218268

Allken, V., Rosen, S., Handegard, N. O., and Malde, K. (2021). A real-world dataset and data simulation algorithm for automated fish species identification. *Geosci. Data J.* 8 (2), 199–209. doi: 10.1002/gdj3.114

Bacheler, N. M., and Shertzer, K. W. (2015). Estimating relative abundance and species richness from video surveys of reef fishes. *Fish. Bull.* 113, 15–26. doi: 10.7755/FB.113.1.2

Barbedo, J. C. A. (2022). A review of the use of computer vision and artificial intelligence for fish recognition, monitoring, and management. *Fishes* 7, 335. doi: 10.3390/fishes7060335

Boulais, O., Alaba, S., Yu, J., Iftekhar, A., Zheng, A., Prior, J., et al. (2021). "SEAMAPD21: a large-scale reef fish dataset for fine-grained categorization," in *The Eighth Workshop on Fine-Grained Visual Categorization – CVPR21*.

Cai, Z., and Vasconcelos, N. (2018). "Cascade r-CNN: Delving into high quality object detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA. 6154–6162. doi: 10.1109/CVPR.2018.00644

Cai, J., Wang, Y., and Hwang, J.-N. (2021). "ACE: Ally complementary experts for solving long-tailed recognition in one-shot," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 1–10.

Campana, S. E. (2001). Accuracy, precision and quality control in age determination, including a review of the use and abuse of age validation methods. *J. Fish Biol.* 59, 197–242. doi: 10.1111/j.1095-8649.2001.tb00127.x

Campbell, M. D., Pollack, A. G., Gledhill, C. T., Switzer, T. S., and DeVries, D. A. (2015). Comparison of relative abundance indices calculated from two methods of generating video count data. *Fish. Res.* 170, 125–133. doi: 10.1016/j.fishres.2015.05.011

Cappo, M., Harvey, E., and Shortis, M. (2007). Counting and measuring fish with baited video techniques – an overview. *Aust. Soc. Fish Biol. 2006 Workshop Proc.* 1, 101–114.

Chuang, M., Hwang, J., Williams, K., and Towler, R. (2014). Tracking live fish from low-contrast and low-frame-rate stereo videos. *IEEE Trans. Circuits Syst. Video Technol.* 25 (1), 167–179. doi: 10.1109/TCSVT.2014.2357093

Connolly, R., Fairclough, D., Jinks, E., Ditria, E., Jackson, G., Lopez-Marcano, S., et al. (2021). Improved accuracy for automated counting of a fish in baited underwater videos for stock assessment. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.658135

Cui, S., Zhou, Y., Wang, Y., and Zhai, L. (2020). Fish detection using deep learning. *Appl. Comput. Intell. Soft Computing* 2020, 1–13. doi: 10.1155/2020/3738108

Dawkins, M., Campbell, M. D., Prior, J., Faillettaz, R., Simon, J., Lucero, M., et al. (2022). FishTrack22: An ensemble dataset for multi-object tracking evaluation. *Second Workshop Comput. Vision Anim.*

Ding, J., Li, X. , and Gudivada, V. N. (2017). Augmentation and evaluation of training data for deep learning. *IEEE Int. Conf. Big Data (Big Data),* 2017, 2603–2611. doi: 10.1109/BigData.2017.8258220

Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., and Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: Optimizing animal ecology with deep learning. *Front. Mar. Sci.* 7, 429. doi: 10.3389/fmars.2020.00429

Ellis, D. M., and DeMartini, E. E. (1995). Evaluation of a video camera technique for indexing abundances of juvenile pink snapper, *Pristipomoides filamentosus*, and other Hawaiian insular shelf fishes. *Fish. Bull.* 93 (1), 67–77.

Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., et al. (2020). Automatic segmentation of fish using deep learning with application to fish size measurement. *ICES J. Mar. Sci.* 77 (4), 1354–1366. doi: 10.1093/icesjms/fsz186

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. doi: 10.1109/CVPR.2016.90

Hsiao, Y., Chen, C., Lin, S., and Lin, F. (2014). Real-world underwater fish recognition and identification using sparse representation. *Ecol. Inf.* 23, 14–21. doi: 10.1016/j.ecoinf.2013.10.002

Jain, A. K. (1989). *Fundamentals of digital image processing* (Englewood Cliffs, NJ: Prentice-Hall).

Jennings, S., and Kaiser, M. J. (1998). The effects of fishing on marine ecosystems. *Adv. Mar. Biol.* 34, 201–352. doi: 10.1016/S0065-2881(08)60212-6

Li, J., Liu, C., Lu, X., and Wu, B. (2022). CME-YOLOv5: An efficient object detection network for densely spaced fish and small targets. *Water* 14 (15), 2412. doi: 10.3390/w14152412

Lopez-Marcano, S., Turschwell, M. P., Brown, C. J., Links, E. L., Wang, D., and Connolly, R. M. (2022). Computer vision reveals fish behaviour through structural equation modelling of movement patterns. *Res. Square Prelim. Rep.*, 1–24. doi: 10.21203/rs.3.rs-1371027/v1

Lopez-Vasquez, V., Lopez-Guede, J., Marini, S., Fanelli, E., Johnsen, E., and Aguzzi, J. (2020). Video image enhancement and machine learning pipeline for underwater animal detection and classification at cabled observatories. *Sensors* 20, 726. doi: 10.3390/s20030726

Lv, Q., and Dong, M. (2022). Active learning of three-way decision based on neighborhood entropy. *Int. J. Innovative Computing Inf. Control* 18 (2), 37–393. doi: 10.1016/j.ins.2022.07.133

Marini, S., Fanelli, E., Sbragaglia, V., Azzurro, E., Rio Fernandez, J., and Aguzzi, J. (2018). Tracking fish abundance by underwater image recognition. *Nat. Sci. Rep.* 8, 13748. doi: 10.1038/s41598-018-32089-8

Marrable, D., Barker, K., Tippaya, S., Wyatt, M., Bainbridge, S., Stowar, M., et al. (2022). Accelerating species recognition and labelling of fish from underwater video with machine-assisted deep learning. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.944582

Ogle, D. (2013). *fishR vignette – precision and accuracy in ages* (Ashland, Wisconsin, United States: Northland College).

Padilla, R., Netto, S. L., and da Silva, E. A. B. (2020). "A survey on performance metrics for object-detection algorithms," in *IEEE International Conference on Systems, Signals, and Processing (IWSSIP)*, Vol. 2020. 237–242. doi: 10.1109/IWSSIP48289.2020.9145130

Priede, I. G., Bagley, P. M., Smith, A., Creasey, S., and Merrett, N. R. (1994). Scavenging deep demersal fishes of the porcupine seabight, north-east Atlantic: observations by baited camera, trap and trawl. *J. Mar. Biol. Assoc. United Kingdom* 74 (3), 481–498. doi: 10.1017/S0025315400047615

Salman, A., Siddiqui, S., Shafait, F., Mian, A., Shortis, M., Khurshid, K., et al. (2020). Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES J. Mar. Sci.* 77 (4), 1295–1307. doi: 10.1093/icesjms/fsz025

Shafait, F., Mian, A., Shortis, M., Ghanem, B., Culverhouse, P., Edgington, D., et al. (2016). Fish identification from videos captured in uncontrolled underwater environments. *ICES J. Mar. Sci.* 73 (10), 2737–2746. doi: 10.1093/icesjms/fsw106

Tabak, M., Norouzzadeh, M., Wolfson, D., Newton, E., Boughton, R., Ivan, J., et al. (2020). Improving the accessibility and transferability of machine learning algorithms for identification of animals in camera trap images: MLWIC2. *Ecol. Evol.* 10, 10374–10383. doi: 10.1002/ece3.6692

Thompson, K. A., Switzer, T. S., Chirstman, M. C., Keenan, S. F., Gardner, C. L., Overly, K. E., et al. (2022). A novel habitat-based approach for combining indices of abundance from multiple fishery-independent video surveys. *Fish. Res.* 247, 106178. doi: 10.1016/j.fishres.2021.106178

van Helmond, A. T. M., Mortensen, L. O., Plet-Hansen, K. S., Ulrich, C., Needle, C. L., and Oesterwind, D. (2020). Electronic monitoring in fisheries: lessons from global experiences and future opportunities. *Fish* 21, 162–189. doi: 10.1111/faf.12425

Villon, S., Chaumont, M., Subsol, G., Villéger, S., Claverie, T., and Mouillot, D. (2016). "Coral reef fish detection and recognition in underwater videos by supervised machine learning: comparison between deep learning and HOG+SVM methods," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, Vol. 2016. 160–171.

Wilberg, M. J., Thorson, J. T., Linton, B. C., and Berkson, J. (2009). Incorporating time-varying catchability into population dynamic stock assessment models. *Rev. Fish. Sci.* 18 (1), 7–24. doi: 10.1080/10641260903294647

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). "Aggregated residual transformations for deep neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5987–5995. doi: 10.1109/CVPR.2017.634

Zion, B., Alchanatis, V., Ostrovsky, V., Barki, A., and Karplus, I. (2007). Real-time underwater sorting of edible fish species. *Comput. Electron. Agric.* 56, 34–35. doi: 10.1016/j.compag.2006.12.007

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 04 April 2023
DOI 10.3389/fmars.2023.1147690

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Luciano Ortenzi,
University of Tuscia, Italy
Sabine Stöhr,
Swedish Museum of Natural History,
Sweden

*CORRESPONDENCE
Xiaoyong Pan
2008xypan@sjtu.edu.cn
Peng Zhou
zhoupeng@sio.org.cn

SPECIALTY SECTION
This article was submitted to
Ocean Observation,
a section of the journal
Frontiers in Marine Science

RECEIVED 19 January 2023
ACCEPTED 09 March 2023
PUBLISHED 04 April 2023

CITATION
Zhou Z, Fu G-Y, Fang Y, Yuan Y,
Shen H-B, Wang C-S, Xu X-W, Zhou P
and Pan X (2023) EchoAI: A deep-learning
based model for classification of
echinoderms in global oceans.
Front. Mar. Sci. 10:1147690.
doi: 10.3389/fmars.2023.1147690

COPYRIGHT
© 2023 Zhou, Fu, Fang, Yuan, Shen, Wang,
Xu, Zhou and Pan. This is an open-access
article distributed under the terms of the
Creative Commons Attribution License
(CC BY). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that
the original publication in this journal is
cited, in accordance with accepted
academic practice. No use, distribution or
reproduction is permitted which does not
comply with these terms.

EchoAI: A deep-learning based model for classification of echinoderms in global oceans

Zhinuo Zhou[1], Ge-Yi Fu[2], Yi Fang[1], Ye Yuan[1], Hong-Bin Shen[1], Chun-Sheng Wang[2], Xue-Wei Xu[2], Peng Zhou[2*] and Xiaoyong Pan[1*]

[1]Key Laboratory of System Control and Information Processing, Ministry of Education of China, Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, [2]Key Laboratory of Marine Ecosystem Dynamics, Ministry of Natural Resources and Second Institute of Oceanography, Ministry of Natural Resources, Hangzhou, China

Introduction: In response to the need for automated classification in global marine biological studies, deep learning is applied to image-based classification of marine echinoderms.

Methods: Images of marine echinoderms are collected and classified according to their systematic taxonomy. The images belong to 5 classes, 38 orders, 145 families, 459 genera, and 1021 species, respectively. The deep learning model, EfficientNetV2, outperforms the competing model and is chosen for developing the automated classification tool, EchoAI. Then, the EfficientNetV2-based tool, EchoAI is applied to each taxonomic level.

Results: The accuracy for the test dataset was 0.980 (class), 0.876 (order), 0.738 (family), 0.612 (genus), and 0.469 (species), respectively. Online prediction service is provided.

Discussion: The EchoAI model and results are facilitated for investigating the diversity, abundance and distribution of species at the global scale, and the methodological strategy can also be applied to image classification of other categories of marine organisms, which is of great significance for global marine studies. EchoAI is freely available at http://www.csbio.sjtu.edu.cn/bioinf/EchoAI/ for academic use.

KEYWORDS

echinoderms, marine organism, deep learning, EfficientNetV2, model interpretability, image classification

Introduction

Extensive survey on marine biodiversity is critical to the sustainable development of oceans, which results in significant workloads of taxonomic determination and classification. For instance, manually determining and classifying images of marine organisms is labor-consuming and time-costing, which requires experienced taxonomic researchers with strong domain knowledge. Moreover, different taxonomic researchers may make different decisions on the same image. Therefore, technologies of automated image classification are greatly demanded, such as machine-learning-based strategies, which consist of feature extraction, classification model training, and prediction. To date, there exist some machine learning-based approaches for automatic marine image classification. For example, these machine learning-based approaches were first applied in fish classifications (White et al., 2006; Larsen et al., 2009; Alsmadi, 2010). Compared with nektons, benthic fauna is relatively motionless, making them suitable for underwater imaging. Currently, deep learning approaches based on convolutional neural networks (CNNs) are increasingly being applied in studies on benthic fauna, such as automated identification of benthic epifauna with computer vision (Piechaud et al., 2019), automated classification of fauna in seabed photographs (Durden et al., 2021).

Among the benthic fauna, species of Echinodermata distribute widely in the oceans, from shallow to abyssal zone, and their biodiversity could be an indicator for health of their habitat. Echinodermata comprises five classes, Asteroidea (starfish), Crinoidea (sea lilies and feather stars), Echinoidea (sea urchins), Holothuroidea (sea cucumbers), and Ophiuroidea (brittle stars) (Mah and Blake, 2012; Stöhr et al., 2012), which differ from each other greatly in appearance. The differences in appearance gradually decrease with taxonomic levels going lower, while the difficulty in classification increases. However, existing machine learning based approaches generally train a unified model on collected images at different taxonomic levels. Currently there is still no specific model for classifying echinoderms at different taxonomic levels, which is in an urgent need for further extensive marine surveys. Therefore, an image-based artificial intelligence classification tool EchoAI for echinoderms at different taxonomic levels is developed in this study, including benchmark dataset construction, model training at different taxonomic levels, model evaluation and online application.

Materials and methods

In this study, we first collected the echinoderms images from World Register of Marine Species (WoRMS, https://marinespecies.org). Then, we trained a deep learning model using these collected images according to the biological systematic classification order. In the end, model interpretation was applied to the images for detecting the key regions. The workflow is shown in Figure 1.

Dataset preparation

The images used in this study were retrieved from the World Register of Marine Species (WoRMS, https://marinespecies.org), which focuses on a worldwide collection of information on marine species. Moreover, the WoRMS platform contains comprehensive taxonomic information on marine species, such as scientific names, corresponding synonyms, and habitat information. Regarding to the dataset at each taxonomic level, the images with missing taxonomic information were not included in the training and test datasets.

Since the format of the raw data downloaded from WoRMS is not exactly the same, it is first necessary to unify the format of the files and convert all the images to the RGB format, so that the image data is consistent with the model input. After the format unification, the images that are corrupted for various reasons were then removed, including images that were lost during format conversion, images that were formatted corruptly when they were downloaded, and images with some special formats. Since the image data downloaded from WoRMS contained images, such as sketch, maps, manual screening of all the images was conducted. Finally, we obtained the dataset for benchmarking in this study (Supplementary Table S1). The details of the datasets for the five classification levels are shown in Table 1.

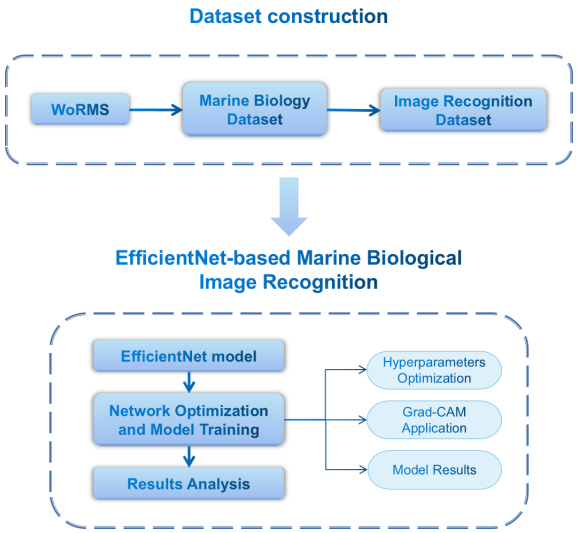


FIGURE 1
The workflow of EchoAI in this study. The entire pipeline starts with dataset preparation, followed by model training using deep-learning model, and interpretation of the prediction results.

Model architecture

EfficientNetV2 model in EchoAI

The module scaling architecture EfficientNet (Tan and Le, 2019) consists of the baseline and a range of non-independent parameters. The most common way is to scale up ConvNets by their depth (He et al., 2016) or width (Zagoruyko and Komodakis, 2016). Another less common, but increasingly popular, way is to scale up the models by image resolution (Huang et al., 2019). In previous work, it is common to scale only one of the three dimensions: depth, width or image size. EfficientNet proposes a simple yet effective module scaling method. The method uniformly scales the network

TABLE 1 The number of images for the five taxonomic levels.

Category Level	Class	Order	Family	Genus	Species
Number of images	4026	3996	3999	4002	3925
Category Number	5	38	145	459	1021

width, depth, and resolution with a set of fixed scaling coefficients (Tan and Le, 2019). This strategy can reduce the number of parameters and the amount of computational resource, while achieving improved performance. However, the series of EfficientNet models still have some defects.

EfficientNetV2 is an improved model based on EfficientNet, it is a smaller and faster group of CNNs compared to the previous models for image recognition. Many previous works, such as FixRes (Touvron et al., 2019), and Mix&Match (Hoffer et al., 2019), usually keep the same regularization for all image sizes, causing a drop in the prediction accuracy. However, EfficientNetV2 proposes a progressive learning, in the early training epochs, they train the network with a small image size and weak regularization, then they gradually increase the image size and add stronger regularization (Tan and Le, 2021). In spite of training parameter efficiency, recent works aim to improve training or inference speed instead of the parameter efficiency. For example, RegNet (Radosavovic et al., 2020), ResNet (Zhang et al., 2020), TResNet (Ridnik et al., 2021), and EfficientNet-X (Li et al., 2021) focus on GPU inference speed. NFNets (Brock et al., 2021) and BoTNets(Srinivas et al., 2021) focus on improving training speed. Their training or inference speed often comes with the cost of more parameters while EfficientNetV2 aims to significantly improve both training speed and parameter efficiency than prior methods (Tan and Le, 2021) Another improvement of EfficientNetV2 is the use of Fused-MBConv (Gupta and Tan, 2019). The structure of the Fused-Convolution block is shown in Supplementary Figure 1. The use of depthwise convolutions (Sifre and Mallat, 2014) in the shallow layers of the network slows down the training in the early stages. EfficientNetV2 leverages the network architecture search to automatically search for the best combination of MBConv and Fused-MBConv.

Learning rate and batch size optimization for EfficientNetV2

The learning rate is a hyperparameter that guides how to adjust the network weights using the gradient of the loss function. The lower the learning rate is, the slower the loss function of the network model changes. The low learning rate allows the model to not miss any of the minimal values, but the model tends to get trapped in the local minima or saddle points. Moreover, the model may fail to converge, while higher learning rates result in faster parameter updates. A high learning rate can lead to gradient explosion, oscillations, etc.

Batch size is the number of samples selected for each training session. During model training, due to the large number of data samples, a certain amount of images from the dataset is selected in batches for training, and then the weights are updated based on the average value of this batch of images. If the batch size is too small, the training time of the model will be too long and the gradient will oscillate severely, making the model too slow to converge. If the batch size is too large, the gradient direction between different batches will vary too small, making the model easy to converge at the local optimum point.

To select the best hyperparameters for model training, empirical hyperparameters and multiple experiments are needed to find the hyperparameters that achieve the best performance on the validation set using grid search, where the optimized model was called as EchoAI (Classification of Echinoderms in the Oceans by EfficientNetV2).

Grad-CAM for model interpretation

The interpretability of network models is of great research importance in evaluating the model robustness. Using the Grad-CAM approach (Selvaraju et al., 2020), the interpretability of EchoAI can be explored, providing a visual interpretation of the decisions for the subsequent classification levels and the accuracy analysis of each category.

Previous work (Zhou et al., 2015) has shown that the convolutional units of various layers of CNNs actually behave as object detectors, even no supervision on the location of the object was provided. CAM (Zhou et al., 2016) is class activation mapping, it can display what the model considers to be the most important in the image during the decision making, which is similar to a heat map. Grad-CAM (Selvaraju et al., 2020) overcomes the disadvantage of CAM that requires replacing the classifier to retrain the model. The basic principle of Grad-CAM is to calculate the weights of each feature map in the convolution layer relative to the image class, and then maps the weighted and summed feature maps to the original input image. The general structure of Grad-CAM is shown in Supplementary Figure 2.

For a category c, Grad-CAM's class activation mapping is calculated as follows:

$$L^{c}_{Grad-CAM} = \mathrm{ReLU}\left(\sum_{i} \alpha_i^c A^i\right) \tag{1}$$

$$\alpha_k^c = \frac{1}{Z}\sum_i^{c_1}\sum_j^{c_2}\frac{\partial S_c}{\partial A_{ij}^k} \tag{2}$$

Where S_c denotes the predicted value of the model for this image; $Z=c_1 \times c_2$ denotes the size of the feature map; k denotes the k-th channel in the feature layer A; A_{ij}^k denotes the data of the feature layer A at the i-th row and j-th column position in the channel k; A^k denotes the data of the k-th channel in the feature layer A; alpha;$_k^c$ *hannel in the feature laye* denotes the targeted weight parameter of A^k.

The mechanism of Grad-CAM (Supplementary Figure 2): The model first makes decisions on the input image, then the output of the last convolutional layer and the final model prediction score are

obtained in the forward propagation. After back-propagating gradient information, the Grad-CAM heat map is obtained by summing the mean value of each point of the feature map with the ReLU activation function.

Experiments

Model evaluation criteria

In this study, we use the accuracy as an evaluation metric to assess the classification performance of the model which rely on a confusion matrix (Manel et al., 2001).

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

where TP, TN, FP, FN are true positives, true negatives, false positives and false negatives.

In order to explore the performance of the training results at each taxonomy level in the echinoderm dataset, the accuracy metric is also extended to multi-class classification tasks. For the overall performance, the accuracy of each taxonomy was also evaluated separately, which takes the impact of inter-class imbalance of the dataset on the model performance into account. The accuracy of each taxonomy is calculated the same as the overall accuracy of the model. For the accuracy of each taxonomy, TP, TN, FP, FN are counted in one specific taxonomy. While for the overall accuracy of the model, TP, TN, FP, FN are counted in the whole dataset.

Results

Learning rate and batch size optimization

In order to optimize the model performance and investigate the relationship between the hyperparameters and the performance of the model, we train the classification model with different learning rates (0.01, 0.001, 0.0001) and different batch sizes (4, 8, 16) at the class level.

As shown in Figures 2A, B, the training loss and accuracy change with the number of iterations for the model training and evaluation. Overall, the higher the learning rate, the faster the model converges. When the learning rate is too low, e.g., learning rate=0.0001, the model falls into a local optimum and cannot find the global optimal solution, and the final training loss is higher than the other two cases. In addition, the accuracy, both in the train and validation sets, is also lower than the other two cases. For the learning rates of 0.01 and 0.001, the performance of the model with a learning rate of 0.01 is better than that of the model with a learning rate of 0.001, both in terms of training loss and accuracy on the train set and the validation set. Thus, 0.01 is chosen as the learning rate of the EchoAI model in our work.

In order to select the appropriate batch size, the batch sizes are set to 4, 8, and 16, respectively. The results of different batch sizes are shown in Figures 2C, D. In terms of the accuracy of the training set, the accuracy of the model with a batch size=4 is lower than that of the model with a batch size 8 or 16 on the training set, but the difference between the models with a batch size 8 or 16 is small.

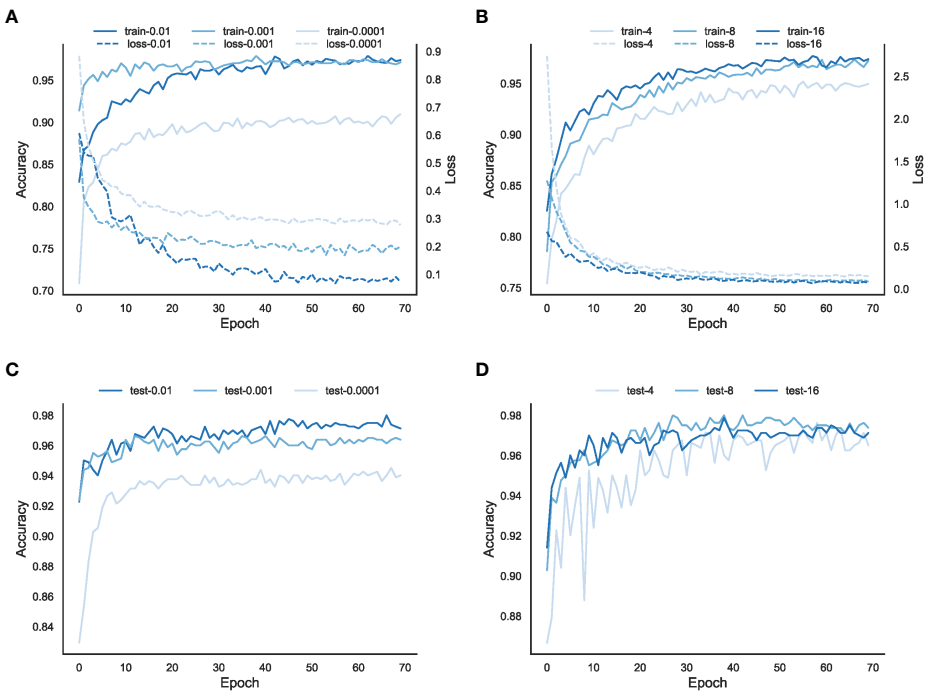


FIGURE 2
Parameter optimization of the model EchoAI with different learning rates and batch sizes. **(A)** is the loss and accuracy of the model in the training set for different learning rates; **(B)** is the accuracy of the model in the validation set for different learning rates; **(C)** is the loss and accuracy of the model in the training set for different batch sizes; and **(D)** is the accuracy of the model in the validation set for different batch sizes.

From the accuracy of the validation set, the accuracy of all three batch sizes is not very different, of which, the batch size=4 is slightly lower. Considering that the training speed is faster with the batch size=16, and the accuracy rates on both the train and validation sets are good, we choose 16 as the batch size for the EchoAI model.

In-depth exploration at different taxonomic levels

When the taxonomic levels going lower, from class to species, the number of categories increases, from 5 (class), 38 (order), 145 (family), 459 (genus), to 1021 (species), and the number of training samples for each category decreases a lot. The performance of the models trained at the five taxonomic levels are shown in Figure 3.

As shown in Figure 3A, EchoAI yields the highest accuracy in the test set at the class level, because the dataset has the least number of categories and each category has the largest number of training samples. The optimal model yields an accuracy of 98.0% in the test set. EchoAI in the order level yields an accuracy of 87.6% in the test set. The accuracy of the EchoAI model in the family level reaches 73.8% in the test set. Based on the higher number of categories in the family level, it can be assumed that the model under the family level also has good predictive power. The accuracy of the EchoAI model in genus level in the test set reaches 61.2%, with the number of categories in the dataset from 145 to 459. The accuracy of the model in the species level reaches 46.9% in the test set, which has expanded the number of categories in the dataset to 1021, and the model can be considered to still have potential predictive power. Although the models in the family, genus and species levels do not perform as well as the models in the class and order levels, the EchoAI model in these levels can still be used as a reference for manual classification.

Since there exist small sample categories in the dataset, it is necessary to focus on the accuracy of each category in addition to the overall accuracy (Figure 3B). The accuracies of the EchoAI model show that there is no small sample classification problem in the classification level of Class. In Order level, its performance is slightly worse than that of Class level classification, but better than the other three classification levels. It is because the number of categories in the Order level is more than that at the Class level, but less than the others, and the number of small sample categories in Order level is smaller. In the classification of the Family level, the distribution of accuracy becomes scattered, the accuracy of some categories reach100%, but the accuracy of a few categories is lower than 75% or even 50%. Moreover, the accuracy of some categories is 0, which shows that the imbalance problem has a big impact on the model performance for those minority categories.

EchoAI model uses EfficientNetV2 as the backbone network, to demonstrate its advantage, we further compare it with ResNet (He et al., 2016) backbone on the same echinoderm dataset. The results are shown in Figure 4. From the loss of the training set at different taxonomic levels (Figure 4A), the convergence speed of the EchoAI model is faster than that of ResNet at each taxonomic level, and the final converged loss is smaller than that of ResNet. From the accuracy of the optimal model in the test set (Figure 4B), the accuracy of the EchoAI model is higher than that of ResNet at each taxonomic level. The results demonstrate that EchoAI with EfficientNetV2 yields better performance on the echinoderm dataset than ResNet.

Model results by top-n prediction

As the classification level of the dataset gradually refines, the number of categories of the data increases and the number of training samples for each category decreases. When the model encounters a more complex multi-classification task, there will be a high probability of predicting the image as other categories, especially for those similar categories. In the previous model training, only the classification of the maximum probability was considered as the predicted category. For the sake of more complete and comprehensive evaluation of the predictive power of the model, we use another judgment criterion for evaluating the model. The model prediction is judged to be correct if the model has the correct category in its top n predictions (the n highest prediction probabilities by EchoAI model). In order to investigate the effect of different values of n on the model evaluation, we perform the evaluation on $n = 1, 2, 3, 4, 5$, respectively, the results for different values of n are shown in Figure 5. We can see that the accuracy decreases with the number of categories and a bigger n yields a higher performance. It is worth noting that, after adjusting the model evaluation criterion, the accuracy of the model EchoAI trained at the species level exceeds

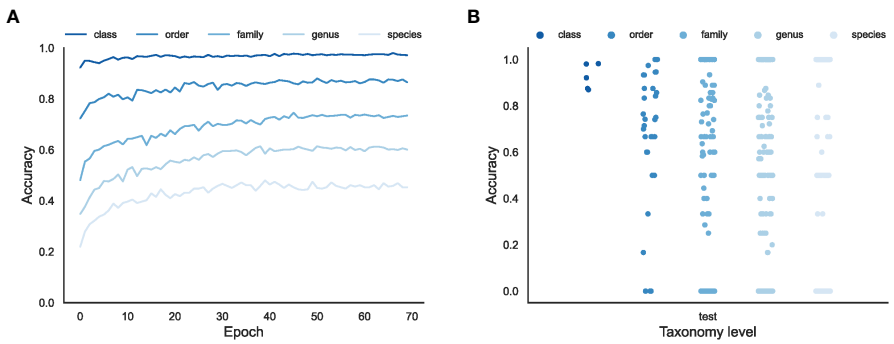


FIGURE 3
The overall effectiveness of the EchoAI models for the test set at different taxonomic levels. **(A)** the change of accuracy over Epoch; **(B)** the distribution of accuracy for each taxon (represented by the point) predicted by the optimized EchoAI.

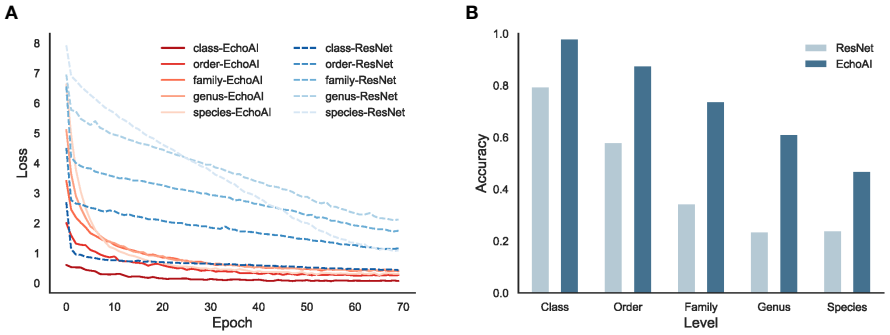


FIGURE 4
The performance comparison of EchoAI and ResNet on the echinoderm dataset. **(A)** is the loss of the training set at different taxonomic levels for EchoAI and ResNet; **(B)** is the optimal accuracy of EchoAI and the ResNet on the test set at each taxonomic level.

0.600 in the test set, reaching 0.678, which is considered to be more reliable with 1021 categories at the species level.

Interpretable analysis of Grad-CAM

In this study, we analyze the impact of the model when the classification level is deepened in three perspectives: the number of categories, the data features in each category, and the amount of images in each category. The information of the dataset has been given in Table 1, and it can be assumed that the size of the data volume at the five classification levels does not affect the model comparison within the error range. We apply Grad-CAM on the trained models for each category. Heat maps (Figure 6) are first drawn by applying Grad-CAM's model at five classification level.

The Grad-CAM heat map shows that the "attention" of the EchoAI model trained at the Class level is well focused on the biological object to be recognized, and the model is not disturbed by the background environment and color. While the attention to the background and the object itself varies at the other classification levels. In contrast, the heat maps of the EchoAI models at other taxonomic levels show that the models do not focus exclusively on the object themselves, and there are even cases where most of the attention is focused on the background. A potential explanation is that Figures 6A–F, the amount of images in this category is small, resulting in the model not learning the discriminate features of the objects for this category.

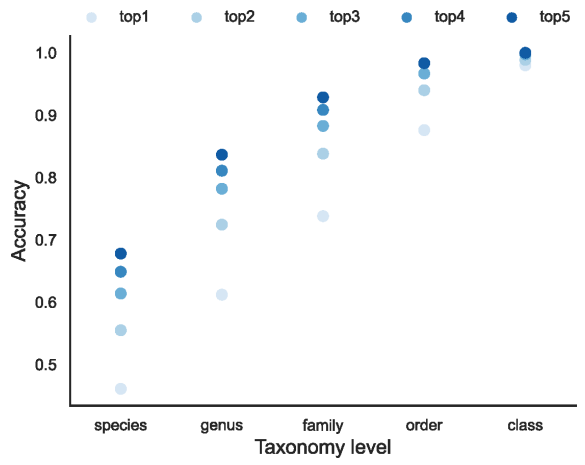


FIGURE 5
The top-n accuracy of EchoAI at different taxonomy levels.

Demonstration and web service of EchoAI

Using the optimized EchoAI model, we demonstrate some prediction examples (shown in Figure 7). Predictions of the above images are all accurate and the probability of prediction is close to 100%, which reflects the strong prediction ability of the model EchoAI. To make EchoAI be accessible for taxonomic classifications of echinoderm images, an online prediction service of EchoAI is provided (http://www.csbio.sjtu.edu.cn/bioinf/EchoAI/). The users could upload their own images and conduct the prediction, by following the instructions on the webpage.

Discussion

Although EchoAI is superior to competing methods, but its accuracy levels may still be not high enough from the perspective of experienced taxonomists. Identification at the family, genus and species levels are much more difficult than that at class and order levels due to the following reasons: 1) the images at the family, genus and species level are very morphologically heterogeneous, which are so similar that microscopic examination is needed; 2) The number of images for each category at the family, genus and species level is very small, which is not sufficient for training a high-accuracy deep model. To improve the performance of EchoAI, the training dataset could be enlarged, even covering the microscopic images.

Verification by the expertise is important for the images fed into the deep model. Some images retrieved from WoRMS may not be verified by a taxonomic expert and may be misidentified. Therefore, EchoAI would be constantly updated along with WoRMS in case certain image is verified by a taxonomic expert. Since there may be misidentifications in the prediction results provided by EchoAI, where non-experts will not be able to recognize them, EchoAI could

10.3389/fmars.2023.1147690

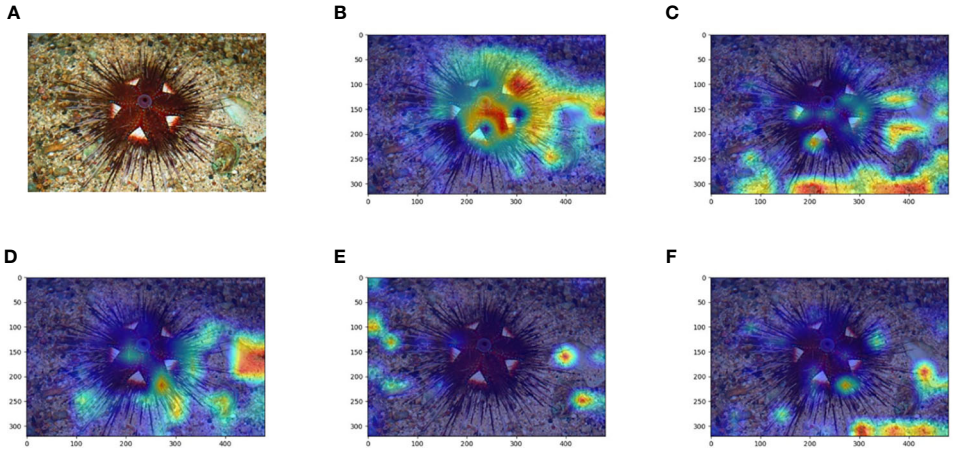


FIGURE 6
Comparison of the original image and Grad-CAM heat map of the Echinoidea image at the five classification levels. **(A)** is the original image, **(B–F)** are Grad-CAM heat maps at the Class, Order, Family, Genus and Species classification levels, respectively.

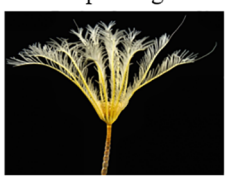
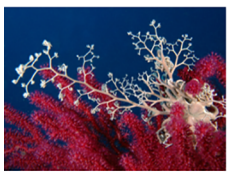
	Inupt Image	Taxonomy	Prediciton(score)
A		class	Crinoidea (1.0)
		order	Comatulida (0.997)
		family	Phrynocrinidae (0.99)
		genus	Porphhyrocrinus (0.995)
		species	Porphyrocrinus daniellalevyae (0.996)
B		class	Asteroidea (1.0)
		order	Paxillosida (1.0)
		family	Astropectinidae (1.0)
		genus	Astropecten (1.0)
		species	Astropecten aranciacus (0.994)
C		class	Ophiuroidea (1.0)
		order	Euryalida (1.0)
		family	Gorgonocephalidae (1.0)
		genus	Astrospartus (1.0)
		species	Astrospartus mediterraneus (1.0)
D		class	Echinoidea (1.0)
		order	Camarodonta (0.999)
		family	Echinometridae (1.0)
		genus	Heterocentrotus (1.0)
		species	Heterocentrotus mamillatus (1.0)
E		class	Holothuroidea (1.0)
		order	Holothuriida (1.0)
		family	Holothuriidae (1.0)
		genus	Holothuria (1.0)
		species	Holothuria (Thymiosycia) impatiens (0.999)

FIGURE 7
Illustration of image prediction results by EchoAI. **(A)** belongs to the category Crinoidea (class classification level); **(B)** belongs to the category Paxillosida (order classification level); **(C)** belongs to the category Gorgonocephalidae (family classification level); **(D)** belongs to the category Heterocentrotus (genus classification level); **(E)** belongs to the category Holothuria (Thymiosycia) impatiens (species classification level).

be functioned as an assistant tool for experienced taxonomists and the misidentifications could be corrected. We expect that EchoAI would benefit the studies on the taxonomic determination.

In future research, the dataset size can be increased and the image quality can be further improved. Considering the difficulty of data acquisition, the development of generative models to augment the categories with fewer samples, especially deep diffusion models (Yang et al., 2022), will be mainly considered. The forward diffusion process is used to model the multi-level hidden variables for this category of image samples, and then the inverse process is used to extract the multi-leveled feature information of the intermediate hidden variables using neural networks, and then the new image is generated as synthesized training samples by inverse sampling of the hidden variables for this category.

Conclusion

In this study, based on images collected from WoRMS, we applied and optimized EchoAI with EfficientNetV2 as the backbone model for classifying marine echinoderms at the levels of class, order, family, genus, species. At the genus level, the size of the dataset is 4002 and the total number of categories is 459. The trained model achieves an accuracy of 0.612 in the test set. The classification by EchoAI is interpretably analyzed using Grad-CAM, and online classification prediction service is provided based on EchoAI. In addition, the classification module can also be extended to other platforms, such as laboratory image analysis equipment, underwater vehicle, etc., to help improve the efficiency of the marine survey and real-time monitoring. The study would help investigate the diversity, abundance and distribution of species at a global scale, and the strategy can also be applied to the image classification of other marine organisms.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding authors.

Author contributions

XP and PZ conceived the study. ZZ, XP, and PZ designed algorithms. ZZ, GF, YY, HS, CW, XX, XP, PZ, and HS wrote the paper. YF implemented the web server. All authors contributed to the article and approved the submitted version.

Funding

This work was sponsored by the Oceanic Interdisciplinary Program of Shanghai Jiao Tong University (No. SL2021MS005, SL2022ZD108), the National Natural Science Foundation of China (No. 61903248), and the Scientific Research Fund of the Second Institute of Oceanography, MNR (No. SZ2101).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1147690/full#supplementary-material

References

Alsmadi, (2010). Fish recognition based on robust features extraction from size and shape measurements using neural network. *J. Comput. Sci.* 6 (10), 1088–1094. doi: 10.3844/jcssp.2010.1088.1094

Brock, A., De, S., Smith, S. L., and Simonyan, K. (2021). "High-performance large-scale image recognition without normalization," in *International conference on machine learning: PMLR* (Vienna, Austria: Proceedings of the 38th International Conference on Machine Learning, PMLR), 1059–1071.

Durden, J. M., Hosking, B., Bett, B. J., Cline, D., and Ruhl, H. A. (2021). Automated classification of fauna in seabed photographs: The impact of training and validation dataset size, with considerations for the class imbalance. *Prog. Oceanography* 196, 102612. doi: 10.1016/j.pocean.2021.102612

Gupta, S., and Tan, M. (2019). EfficientNet-EdgeTPU: Creating accelerator-optimized neural networks with AutoML. *Google AI Blog* 2, 1. Available: https://ai.googleblog.com/2019/08/efficientnet-edgetpu-creating.html.

He, K. M., Zhang, X. Y., Ren, S. Q., and Sun, J. (2016). "Deep residual learning for image recognition," in *2016 ieee conference on computer vision and pattern recognition (Cvpr)* (Las Vegas, Nevada: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)), 770–778. doi: 10.1109/Cvpr.2016.90

Hoffer, E., Weinstein, B., Hubara, I., Ben-Nun, T., Hoefler, T., and Soudry, D. (2019). Mix & match: training convnets with mixed image sizes for improved accuracy, speed and scale resiliency. *arXiv* preprint arXiv:1908.08986. doi: 10.48550/arXiv.1908.08986

Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M., et al. (2019). Gpipe: Efficient training of giant neural networks using pipeline parallelism. *Adv. Neural Inf. Process. Syst.* 32, 103–112. doi: 10.48550/arXiv.1811.06965

Larsen, R., Olafsdottir, H., and Ersboll, B. K. (2009). Shape and texture based classification of fish species. *Image Analysis Proc.* 5575, 745–749. doi: 10.1007/978-3-642-02230-2_76

Li, S., Tan, M., Pang, R., Li, A., Cheng, L., Le, Q. V., et al (2021). "Searching for fast model families on datacenter accelerators," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)), 8085–8095.

Mah, C. L., and Blake, D. B. (2012). Global diversity and phylogeny of the asteroidea (Echinodermata). *PloS One* 7 (4), e35644. doi: 10.1371/journal.pone.0035644

Manel, S., Williams, H. C., and Ormerod, S. J. (2001). Evaluating presence–absence models in ecology: the need to account for prevalence. *J. Appl. Ecol.* 38 (5), 921–931. doi: 10.1046/j.1365-2664.2001.00647.x

Piechaud, N., Hunt, C., Culverhouse, P., Foster, N., and Howell, K. (2019). Automated identification of benthic epifauna with computer vision. *Mar. Ecol. Prog. Ser.* 615, 15–30. doi: 10.3354/meps12925

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., and Dollár, P. (2020). "Designing network design spaces," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)), 10428–10436.

Ridnik, T., Lawen, H., Noy, A., Ben Baruch, E., Sharir, G., and Friedman, I. (2021). "Tresnet: High performance gpu-dedicated architecture," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (Waikoloa, HI, USA: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)), 1400–1409.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2020). Grad-CAM: Visual explanations from deep networks *via* gradient-based localization. *Int. J. Comput. Vision* 128 (2), 336–359. doi: 10.1007/s11263-019-01228-7

Sifre, L., and Mallat, S. (2014). Rigid-motion scattering for texture classification. *arXiv* preprint arXiv:1403.1687.

Srinivas, A., Lin, T.-Y., Parmar, N., Shlens, J., Abbeel, P., and Vaswani, A. (2021). "Bottleneck transformers for visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (Nashville, Tennessee), 16519–16529. doi: 10.48550/arXiv.1403.1687

Stöhr, S., O'Hara, T. D., and Thuy, B. (2012). Global diversity of brittle stars (Echinodermata: Ophiuroidea). *PloS One* 7 (3), e31940. doi: 10.1371/journal.pone.0031940

Tan, M. X., and Le, Q. V. (2019). "EfficientNet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning* (Los Angeles, United States: Proceedings of the 36th International Conference on Machine Learning, PMLR), vol. 97. .

Tan, M. X., and Le, Q. V. (2021). "EfficientNetV2: Smaller models and faster training," in *International conference on machine learning* (Vienna, Austria: Proceedings of the 38th International Conference on Machine Learning, PMLR), vol. 139. , 7102–7110.

Touvron, H., Vedaldi, A., Douze, M., and Jégou, H. (2019). Fixing the train-test resolution discrepancy. *Adv. Neural Inf. Process. Syst.* 32, 8252–8262. doi: 10.48550/arXiv.1906.06423

White, D. J., Svellingen, C., and Strachan, N. J. C. (2006). Automated measurement of species and length of fish by computer vision. *Fisheries Res.* 80 (2-3), 203–210. doi: 10.1016/j.fishres.2006.04.009

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., et al. (2022). Diffusion models: A comprehensive survey of methods and applications. *arXiv* preprint arXiv:2209.00796. doi: 10.48550/arXiv.2209.00796

Zagoruyko, S., and Komodakis, N. (2016). Wide residual networks. *Proceedings of the British Machine Vision Conference (BMVC)* 2016, 87.1-87.12. doi: 10.5244/C.30.87

Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., et al. (2022). ResNeSt: Split-attention networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2736-2746. doi: 10.1109/CVPRW56347.2022.00309

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba., A. (2015). Object detectors emerge in deep scene cnns. *arXiv* preprint arXiv:1412.6856. doi: 10.48550/arXiv.1412.6856

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). "Learning deep features for discriminative localization," in *2016 ieee conference on computer vision and pattern recognition (Cvpr)* (Las Vegas, Nevada: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)), 2921–2929. doi: 10.1109/Cvpr.2016.319

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 05 April 2023
DOI 10.3389/fmars.2023.1151758

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Duane Edgington,
Monterey Bay Aquarium Research Institute (MBARI), United States
Peng Ren,
China University of Petroleum, China

*CORRESPONDENCE
Ignacio A. Catalán
Ignacio@imedea.uib-csic.es

†These authors share first authorship

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 26 January 2023
ACCEPTED 20 March 2023
PUBLISHED 05 April 2023

CITATION
Catalán IA, Álvarez-Ellacuría A, Lisani J-L, Sánchez J, Vizoso G, Heinrichs-Maquilón AE, Hinz H, Alós J, Signarioli M, Aguzzi J, Francescangeli M and Palmer M (2023) Automatic detection and classification of coastal Mediterranean fish from underwater images: Good practices for robust training. *Front. Mar. Sci.* 10:1151758. doi: 10.3389/fmars.2023.1151758

COPYRIGHT
© 2023 Catalán, Álvarez-Ellacuría, Lisani, Sánchez, Vizoso, Heinrichs-Maquilón, Hinz, Alós, Signarioli, Aguzzi, Francescangeli and Palmer. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Automatic detection and classification of coastal Mediterranean fish from underwater images: Good practices for robust training

Ignacio A. Catalán[1*†], Amaya Álvarez-Ellacuría[1†], José-Luis Lisani[2†], Josep Sánchez[2], Guillermo Vizoso[1], Antoni Enric Heinrichs-Maquilón[2], Hilmar Hinz[1], Josep Alós[1], Marco Signarioli[1], Jacopo Aguzzi[3,4], Marco Francescangeli[5] and Miquel Palmer[1]

[1]Marine Ecology Department, Mediterranean Institute for Advanced Studies (IMEDEA) Spanish National Research Council-University of the Balearic Islands (CSIC-UIB), Esporles, Spain, [2]Mathematics and Computer Science Department, University of the Balearic Islands (UIB), Palma, Spain, [3]Department of Renewable Marine Resources, Institut de Ciències del Mar (ICM-Spanish National Research Council), Passeig Maritim de la Barceloneta, Barcelona, Spain, [4]Department of Research Infrastructures for Marine Biological Resources, Anton Dohrn Zoological Sation, Naples, Italy, [5]SARTI Research Group, Electronics Department, Universitat Politècnica de Catalunya (UPC), Vilanova i la Geltrú, Spain

Further investigation is needed to improve the identification and classification of fish in underwater images using artificial intelligence, specifically deep learning. Questions that need to be explored include the importance of using diverse backgrounds, the effect of (not) labeling small fish on precision, the number of images needed for successful classification, and whether they should be randomly selected. To address these questions, a new labeled dataset was created with over 18,400 recorded Mediterranean fish from 20 species from over 1,600 underwater images with different backgrounds. Two state-of-the-art object detectors/classifiers, YOLOv5m and Faster RCNN, were compared for the detection of the 'fish' category in different datasets. YOLOv5m performed better and was thus selected for classifying an increasing number of species in six combinations of labeled datasets varying in background types, balanced or unbalanced number of fishes per background, number of labeled fish, and quality of labeling. Results showed that i) it is cost-efficient to work with a reduced labeled set (a few hundred labeled objects per category) if images are carefully selected, ii) the usefulness of the trained model for classifying unseen datasets improves with the use of different backgrounds in the training dataset, and iii) avoiding training with low-quality labels (e.g., small relative size or incomplete silhouettes) yields better classification metrics. These results and dataset will help select and label images in the most effective way to improve the use of deep learning in studying underwater organisms.

KEYWORDS

deep learning, mediterranean, fish, pre-treatment, YOLOv5, EfficientNet, faster RCNN

Introduction

Underwater marine images are widely used to study fish abundance, behavior, size structure, and biodiversity at multiple spatial and temporal scales (Aguzzi et al., 2015; Díaz-Gil et al., 2017; Follana-Berná et al., 2022; Francescangeli et al., 2022). In recent years, advances in artificial intelligence and computer vision, specifically deep learning (DL), have enabled the reduction of the number of hours required for manually detecting and classifying species in images. Studies have demonstrated the capabilities of these techniques, particularly deep convolutional networks (CNN; LeCun et al., 1998; Lecun et al., 2015) in detecting and classifying fish in underwater images or video streams (Salman et al., 2016; Villon et al., 2018, see reviews in Goodwin et al., 2022; Li and Du, 2022; Mittal et al., 2022; Saleh, Sheaves and Rahimi Azghadi, 2022). These studies have utilized different types of image databases and have faced similar unresolved questions, such as the number of fish needed for training (Marrable et al., 2022), the need for color image pre-processing (e.g., Lisani et al., 2022), the need for transfer learning from large databases (e.g., Imagenet or coco), improving results when working with small image areas or limited computing power (Paraschiv et al., 2022), whether to use segmentation of bounding boxes and how well a trained set will perform for different habitats (backgrounds). In particular, the detection and classification of multiple species using different combinations of backgrounds (the "domain-shift" phenomenon: Kalogeiton et al., 2016; Ditria et al., 2020), number of species, and labeling quality, is an area that requires further investigation. In general, it is believed that a greater volume of training data and a greater variety of backgrounds can improve the performance of DL datasets (Moniruzzaman et al., 2017; Sarwar et al., 2020; Ditria et al., 2020). Highly varied backgrounds are typical in coastal areas, where non-invasive video-based automatic fish censusing methods are increasingly needed for conservation and fisheries sustainability issues (Aguzzi et al., 2020; Connolly et al., 2021; Follana-Berná et al., 2022). However, these types of exercises are limited, and the need for a high number of labeled individuals from many species can be challenging in areas or laboratories with limited resources.

The Mediterranean Sea is an example of a scarcity of approaches in the field of DL for fish detection. A recent search in the Web of Science for papers on "Deep Learning", "Fish" and "Mediterranean" (conducted in December 2022) yielded only seven results, with only one of them taking into account the variation of background (seasonal variation over time, in a fixed station) in a multispecific dataset of Mediterranean fish (Ottaviani et al., 2022). The Mediterranean is a highly diverse sea (Coll et al., 2010) where underwater video monitoring exercises are primarily semi-supervised (Aguzzi et al., 2015; Díaz-Gil et al., 2017; Marini et al., 2018b; Follana-Berná et al., 2019, Follana-Berná et al., 2022) and monitoring is essential due to the high impact of invasive species and climate change (Azzurro et al., 2022). In this context, the main objective of this work is to evaluate, for newly generated Mediterranean fish datasets of over 20 species, the relative importance of combining backgrounds in the detection (of "fish") and classification (of species), how these combinations interact with the balance/unbalance in fish labeling, and how the labeling quality affects the quality of fish detection. Additionally, we compare, as a function of matrix size, the classification performance of a single-step classifier (i.e., objects are classified into specific categories) versus a classifier requiring a two-step procedure (objects are first classified into a generic fish category, and then classified into more specific categories).

Material and methods

Four different underwater image datasets were constructed for analysis (Table 1). Datasets A through C are newly generated images and are available in a free repository (Zenodo, https://doi.org/10.5281/zenodo.7534425). Dataset A (Figure 1) was created using images from an underwater cabled camera located in a wreck inside Andratx Bay on the western coast of Mallorca Island (Subeye, https://imedea.uib-csic.es/sites/sub-eye/home_es/). The camera (SAIS-IP-bullet cam, 2096 x 1561 pixels) was situated within the wreck (6 m depth) and has been sending still images every 5 mins since 2019 to our research center. Dataset B was obtained from various underwater video surveys in Palma Bay on the southern coast of Mallorca Island. The cameras were used either in drop-down surveys (Go-pro Hero 3, 1920 x 1440) or were operated by scuba divers (Go-pro Hero 7, 1920 x 1440). The obtained images included depths ranging from 5 to 20 meters, and balanced backgrounds, including sand, seagrass meadows and rocks were selected (Figure 1B). In both A and B datasets, more than 20 object classes (species/genus) were observed (Table 2) and labeled by an expert using bounding boxes. The number of observations of each species ranged from 2 to more than 3000; this imbalance forced us to reduce the bulk of the main analyses to 9 fish classes with a higher number of observations, although some species with a low number of labels were included for comparison (Table 2). Subsets of the main datasets A and B were used as validation sets, as detailed in Table 1. Training and validation (approx 20% of the images) were conducted using an NVIDIA QUADRO GV100 32 GB GPU. Four small test sets (images not belonging to the validation or training sets) were also used, both from datasets A and B and from two external datasets. The first external dataset (dataset C, Table 1, Figure 1) consisted of images from a second fixed camera located at 4 m from the wreck (8 m depth, Sony Ipela SNC-CH210 2048 x 1536 pix). Additionally, a small set (dataset D, Table 1; see Figure 1) from the OBSEA cabled observatory located in Catalonia, NE Spain (Aguzzi et al., 2011) was also used as a test set (Francescangeli et al., 2023).

Datasets pre-processing and scenarios

Underwater images often exhibit low contrast, color cast, noise and haze due to depth-dependent attenuation of light wavelengths and the scattering effect (Hsiao et al., 2014; Wang et al., 2019; Zhou et al., 2020; Wang et al., 2023). To improve the dataset images, we employed the Multiscale Retinex Model (MSR, Land and McCann, 1971), which has been identified as one of the best methods for

TABLE 1 Combination of images and number of fish for each of the scenarios (E0-5) used to detect fish.

		Scenarios					
Train and Validation datasets	Fish or image (train/validation/test)	E0	E1	E2	E3	E4	E5
		Imb; all A & B	Imb; reduced A, no B	Imb; All B	Bal; A & B	Imb; All A	Bal; reduced and selected A & B
A	FISH (train)	12096	3074	0	3074	12096	1462
	FISH (validation)	2422	892	0	892	2431	140
B	FISH (train)	3032	0	3032	3032	0	1716
	FISH (validation)	892	0	892	892	0	388
	TOTAL FISH (train)	15128	3074	3032	6106	12096	3178
	TOTAL FISH (validation)	3314	892	892	1784	2431	528
A	IMAGES (train)	762	196	0	196	762	168
	IMAGES (validation)	184	69	0	70	184	24
B	IMAGES (train)	576	0	576	576	0	305
	IMAGES (validation)	143	0	143	142	0	58
	TOTAL IMAGES (train)	1338	196	576	772	762	473
	TOTAL IMAGES (validation)	327	69	143	212	184	82
Test datasets							
A	FISH (test)	235	235	235	235	235	235
	IMAGES (test)	15	15	15	15	15	15
B	FISH (test)	290	290	290	290	290	290
	IMAGES (test)	13	13	13	13	13	13
C	FISH (test)	369	369	369	369	369	369
	IMAGES (test)	43	43	43	43	43	43
D	FISH (test)	103	103	103	103	103	103
	IMAGES (test)	21	21	21	21	21	21

A and B datasets were split into training, validation and test sets. Further, test sets C and D were obtained from different areas and backgrounds. For classification, see further in the text. Imb, imbalanced scenario. Bal, balanced scenario.

detecting fish for labeling purposes using different backgrounds (Lisani et al., 2022).

After image enhancement, labeling was conducted using the free online software Supervisely (https://supervise.ly/). Six training datasets were created to evaluate the relevance of the type of background and number of fish within the images for neural network training (Table 1).

Scenario E0 included all the training images available from both datasets A and B (15128 objects and 1338 images for training, 3314 objects and 327 images for validation). Scenario E1 was a reduced subsample of dataset A, comprising 196 images and 3074 fishes. Scenario E2 included all of dataset B, comprising 546 images and 3032 fishes. Scenario E3 was a balanced scenario, containing around 3000 fish for each dataset A and B. Scenario E4 contained all the training images of dataset A (12096 objects and 762 images for training, 2442 objects and 184 images for validation). Finally, scenario E5 consisted of a selected group of images from both datasets A and B (approx 1500 fish each), avoiding images that appeared to disturb the training, particularly those that did not include small fish (<100 pixels^2, Figure 2) or overlapping fish.

Fish detection and classification were compared in two steps. First, two state-of-the-art CNNs (Faster R-CNN and YOLOv5M) were compared across scenarios for single-class detection (fish/no fish). Second, classification metrics were compared between the best-performing network in classifying fish/no fish, which was then used as both a detector and classifier, and a pure classifier network (the latter using only the bounding boxes previously classified as "fish"). For classification training, fish were pre-classified to the lowest taxonomical category possible (species, genus, or family).

Models metrics

Model comparison and evaluation (see below) on validation or test sets was conducted through the analysis of the interaction of two standard metrics: precision (P) and recall or sensitivity (R) (Everingham et al., 2010). For a given fish class, precision is defined

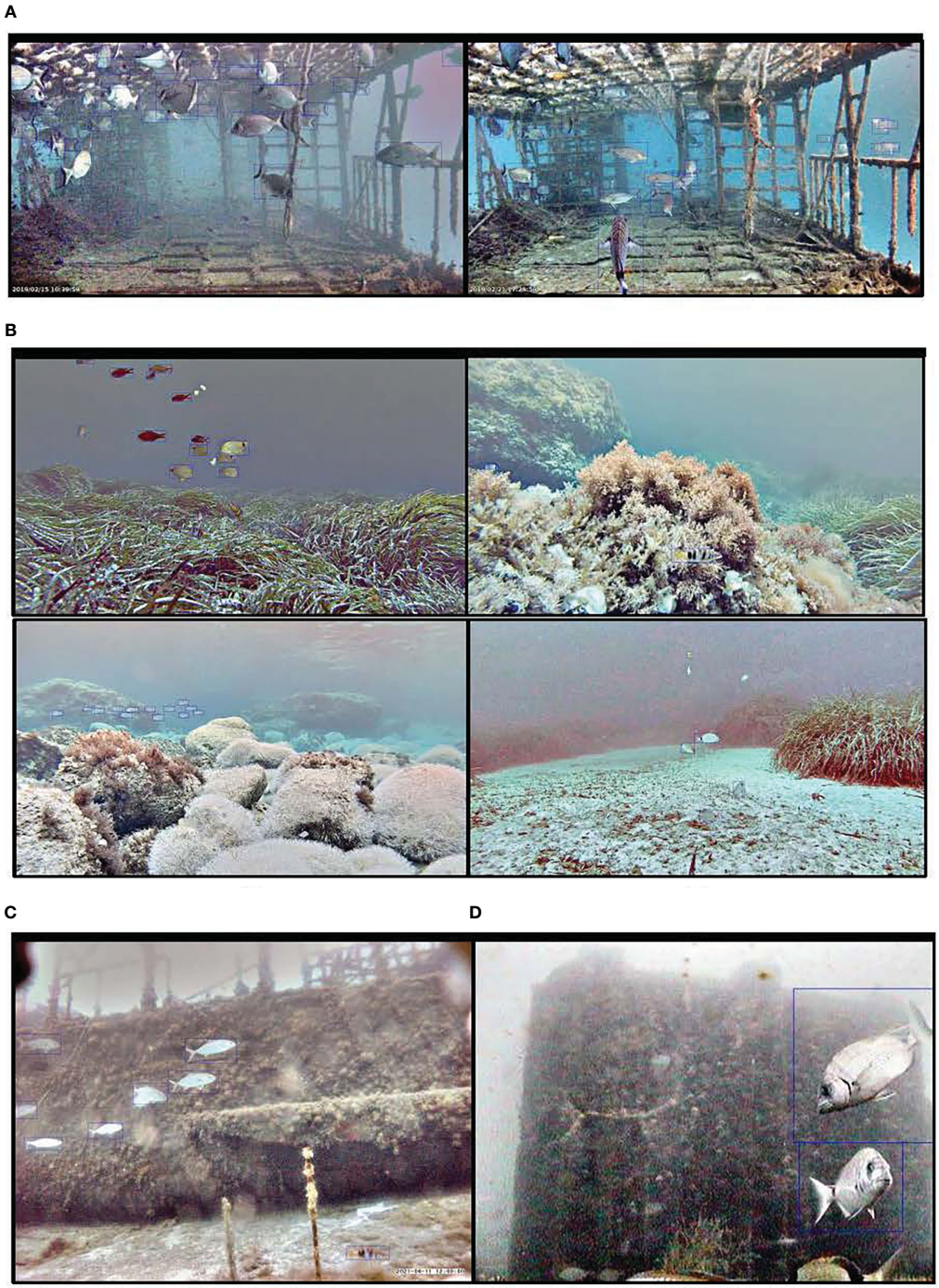


FIGURE 1
Example images from the main coastal Mediterranean datasets **(A)** fixed observatory, **(B)** varied coastal bottoms, and two other test datasets **(C)** fixed observatory in Mallorca, **(D)** fixed observatory in Catalonia.

as the fraction of relevant fish among all retrieved fish, whereas recall is the fraction of retrieved and relevant fish among all relevant fish. They are defined as:

$$P = \frac{TP}{TP + FP}; R = \frac{TP}{TP + FN}$$

where TP=true positive, FP=false positive and FN=false negative.

Neither P nor R provide a full picture of the model performance. To attain a more global metric for comparisons, we calculated the F1 score and the mean average precision (mAP). The F1 score will only be high if both P and R are high and is calculated as:

10.3389/fmars.2023.1151758

TABLE 2 Count and image example (after MSR model pre-processing) of the main fish classes appearing in datasets A and B.

Class name	Example Image	Occurrence in A	Occurrence in B	Total
Unidentified fish	–	3309	771	4080
Chromis chromis		2788	1357	4145
Coris julis		7	572	579
Dentex dentex		5	0	5
Diplodus annularis		121	637	758
Diplodus puntazzo		2	5	7
Diplodus sargus		3301	12	3313
Diplodus sp.		1090	8	1098
Diplodus vulgaris		1155	379	1534
Epinephelus costae		2	0	2
Epinephelus marginatus		2	0	2
Lithognathus mormyrus		395	0	395
Mugilidae (prob Chelon)		483	0	483
Mullus surmuletus		3	12	15
Oblada melanura		972	68	1040
Pomatous saltatrix		234	0	234
Sarpa salpa		20	75	95
Seriola dumerilii		1256	0	1256

(Continued)

TABLE 2 Continued

Class name	Example Image	Occurrence in A	Occurrence in B	Total
Serranus scriba		17	203	220
Sparus aurata		80	0	80
Sphyraena viridis		27	0	27
Symphodus sp.		22	257	279

Some species were aggregated to a genus level if species could not be recognized, or it was a genus with many species appearing in low abundances (e.g., *Symphodus*).

$$F1\ \ score = \frac{2 \cdot P \cdot R}{P + R}$$

The P and R values from the nets were obtained so that they maximized the F1 score, thus achieving their best balance. The mAP is often used for global model comparison and is calculated as the area under the precision vs recall curve, at all levels of intersection over union (http://cocodataset.org/). Here, we calculated mAP@0.5, meaning that true positives are defined as detections whose bounding boxes have at least a 50% overlap with the ground truth bounding boxes. This overlap is measured in terms of the Intersection over Union (IoU), which ranges from 0 to 1, as the ratio between the area of their intersection and the area of their union.

Object detection

For object (class "fish") detection, we first compared the performance of Faster RCNN (Ren et al., 2015) and several configurations of the fifth version of the You Only Look Once (YOLO) algorithm (first described by Redmon et al., 2016), using the implementation from Ultralytics (https://github.com/ultralytics/yolov5). YOLOv5 has been shown to work particularly well in underwater environments (Wang et al., 2021). The medium pre-trained model from YOLOv5, YOLOv5m (pre-trained on COCO image database, http://cocodataset.org/) was selected after training on the E0 scenario with the l, m and x pre-trained models (Supplementary Table S1).

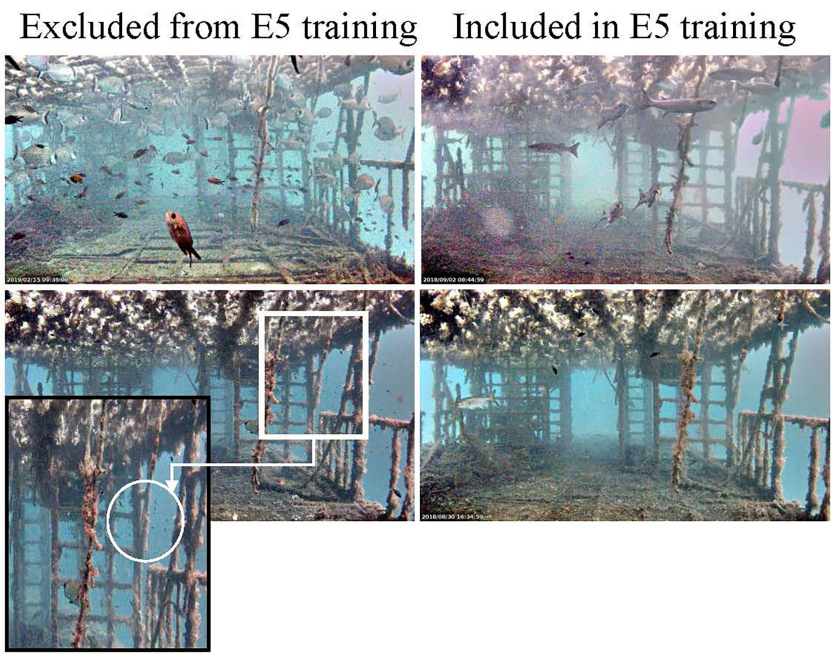


FIGURE 2
Left panel, images excluded from scenario E5 due to small fish and abundant overlap (note inset in the lower-left picture for small fish). Right panel: types of images included in scenario E5, selected for clearly identifiable species.

YOLOv5m (hereafter referred to as YOLO) produced the best compromise between metrics (mAP@0.5 = 0.84, precision=0.83, recall=0.78) and computation time and was selected for subsequent analyses. For Faster RCNN we used the implementation for object detection from de TensorFlow API (https://www.tensorflow.org/api_docs), with the ResNet50 configuration, pre-trained on ImageNet (https://www.image-net.org/). Object detection performance was evaluated on each training scenario using the aforementioned metrics.

Classification

Fish can be classified in a single step using the YOLO algorithm, which scans the entire image, identifies fish, and classifies them. Alternatively, a classifier that only operates on pre-defined bounding boxes of fish can also be used among other possibilities. We compared the results from a state-of-the-art classifier, EfficientNet V2 (here forth EfficientNet) (Tan and Le, 2021) implemented with the TensorFlow API, with those from the best-performing YOLO model. The EfficientNet was trained on the Google Colab platform (https://colab.research.google.com/), while the YOLO network was trained locally on an NVIDIA GPU. An initial comparison was conducted using two sets of increasing fish object classes (4 and 8 classes) to observe the effect of the number of classes and instances on classification success. Given the superior performance of YOLO on classification (see corresponding section), it was used to further compare the effect on increasing the number of fish categories with more than 50 individuals (4, 8, 14 species) in expanded class sets. Each trial was trained using only the selected classes in each set. Confusion matrices are provided for selected results, and specific variations in the species composition were made, re-training the network to illustrate the confounding effect of including new fishes at the genus level that could not be classified to species level but were morphologically similar. Direct comparisons between YOLO and EfficientNet performance using mAP cannot be made due to structural differences in the networks, so F1 score (mean ± SD) was used to compare equal sets of species datasets.

Results

Fish detection

The comparison of the two networks, YOLO and Faster RCNN, across six scenarios revealed that YOLO performed notably better than Faster RCNN both in validation and test sets in most cases (Tables 3, 4) with mAP@0.5 values over 0.8 in most scenarios in the validation datasets (Table 3). The inferior performance of Faster RCNN was primarily attributed to lower R values. In general, using a larger number of fish resulted in slightly better results. However, it was noteworthy that E5 achieved nearly as good results using one-tenth the number of objects for training, but only considering images without small fish and using a balanced set of backgrounds. Comparing YOLO results in the test sets across scenarios, the following patterns were apparent (Table 4, see Supplementary Figure S1 for examples): the evaluation of scenarios that were not trained with either A or B datasets performed poorly on the test sets from datasets not used in training, but not necessarily with other never-before-seen datasets (C and D, scenarios E2, E4). The best results across test sets were obtained using a YOLO network trained in scenario E0 (high number of fish but unbalanced background), followed by E3 (trained on approximately half the objects but with balanced datasets) and E5 (half the images than E3, balanced datasets and selected images). These three training scenarios yielded mAP@0.5 values ranging from 0.70-0.84 across all test scenarios.

Species classification

As expected, classification metrics tended to improve with an increasing number of objects. On average, YOLO performed better

TABLE 3 Performance metrics for each scenario computed over the training datasets (see Table 1).

Training Scenario	Model	P	R	mAP@0.5
E0	Faster RCNN	0.80	0.48	0.60
	YOLO	0.83	0.78	0.84
E1	Faster RCNN	0.66	0.45	0.37
	YOLO	0.75	0.75	0.77
E2	Faster RCNN	0.76	0.52	0.83
	YOLO	0.84	0.73	0.80
E3	Faster RCNN	0.81	0.45	0.70
	YOLO	0.82	0.73	0.80
E4	Faster RCNN	0.71	0.53	0.42
	YOLO	0.81	0.79	0.84
E5	Faster RCNN	0.78	0.53	0.83
	YOLO	0.88	0.71	0.83

See Table 4 for test sets. Noticeably, E5 yielded relatively good results with a low number of training objects (by eliminating fish that are only dots or very difficult to recognize at the species level).

TABLE 4 Results of the application of YOLO to the four test datasets (never seen by the trained DL nets, see Table 1).

	mAP@0.5 value for each test dataset Number of training objects						
Training Scenario	Model	A	B	C	D	Training objects	Backgrounds
	Faster RCNN	0.34	0.34	0.48	0.63		
E0	YOLO	0.84	0.83	0.80	0.78	15128	Unbalanced
	Faster RCNN	0.35	0.16	0.47	0.57		
E1	YOLO	0.82	0.34	0.78	0.83	3074	Unbalanced
	Faster RCNN	0.15	0.35	0.39	0.42		
E2	YOLO	0.34	0.64	0.49	0.43	3032	Unbalanced
	Faster RCNN	0.32	0.30	0.47	0.55		
E3	YOLO	0.82	0.81	0.80	0.76	6106	Balanced
	Faster RCNN	0.35	0.18	0.48	0.74		
E4	YOLO	0.86	0.45	0.79	0.82	12096	Unbalanced
	Faster RCNN	0.32	0.29	0.48	0.69		
E5	YOLO	0.79	0.80	0.76	0.76	3178	Balanced

Balanced and unbalanced scenarios and the number of training objects are specified.

than EfficientNet when using eight species, although both networks performed similarly on four species (Table 5). The average F1 score for both networks was around 0.75 for four species. For eight species, YOLO showed around 14% higher values than EfficientNet, with a standard deviation one order of magnitude lower. In some cases, EfficientNet had high precision and F1 score for classes with a low number of objects (e.g., *S. scriba*) when the number of classes was low. Overall, YOLO was considered a more convenient tool, providing reasonable results in an integrated detection and classification process. A test for confounding species showed that if a class that could contain two similar species was included (*Diplodus* sp.), YOLO confused it with *D. sargus* at the same proportion as the generic *Diplodus* sp. (Figure 3). The category "background" (Figure 3, see also Figure S2) comprises different objects depending on the matrix size. In small matrices (e.g., four sp.), wrongly classified information is included in the background category, which in fact contains general categories like "fish", plus others (See Supplementary Figure S2). When the category "fish" is included, most of the information previously attributed to background is, in many instances, attributed to this "fish" category (see Figure S2). This general category is comprised by fish that were unidentifiable at a higher taxonomic resolution. Additionally, a large proportion of true *Diplodus* sp was inferred to be background, likely due to initial labeling issues: the contour of these *Diplodus* sp. could not be fully determined due to partial overlap with other fish. Using YOLO in a larger dataset (Table S2, Supplementary Figure S2) showed that, although the average classification power decreased, i) increasing the number of species did not necessarily decrease the classification success for the species with large numbers (e.g., *C.chromis*, *D.sargus*, *D.vulgaris*) or conspicuous shape differences with respect to the others (e.g., Mugilidae) (See Figure S1 for an example), ii) several other species with a low number of labels were reasonably classified (e.g., *L.mormyrus*, *P.saltatrix*). These well-detected species were conspicuous and largely different in shape or color from the rest (see Table 2).

Discussion

In this paper, we present a new labeled dataset of underwater images of coastal Mediterranean fishes and investigate the best dataset combinations for obtaining optimal deep learning (DL)-based classification results that can be applied to various habitats. Firstly, we compared two popular architectures, Faster RCNN and YOLO, in terms of their object detection capabilities. Results indicate that YOLO significantly outperforms Faster RCNN in detecting the category "fish" and performs better than EfficientNet in many cases, without the need for pre-defining bounding boxes. However, in some instances, such as classifying conspicuous species in scenarios of limited training data, directly utilizing bounding boxes may yield better results, as observed in other studies (Knausgård et al., 2022).

Using YOLO, we addressed specific areas that required further investigation, particularly the "domain shift" phenomenon (Kalogeiton et al., 2016; Ditria et al., 2020) characterized by a decrease in classification performance with varying habitat backgrounds and fish species assemblages. Automatic fish classification often involves the use of relative or absolute (e.g., Campos-Candela et al., 2018) abundance estimators that utilize underwater baited cameras (Connolly et al., 2021) or cabled observatories (Bonofiglio et al., 2022) to count, classify or track fish (Saleh et al., 2022). These underwater images differ significantly from typical free datasets that contain single individuals; these images contain a high diversity of species and large variability in abundance, resulting in reduced classification success. However, as

TABLE 5 Results of comparable classification metrics between YOLO and EfficientNet using either 4 or 8 classes.

			YOLO				EfficientNet		
4 classes	Training objects	Validation objects	P	R	F1 score		P	R	F1 score
C. chromis	2730	854	0.80	0.65	0.72	*C. chromis*	0.86	0.97	0.91
D. sargus	2281	492	0.79	0.73	0.76	*D. sargus*	0.81	0.85	0.83
D. vulgaris	1011	251	0.83	0.61	0.70	*D. vulgaris*	0.74	0.37	0.50
S. scriba	152	48	0.87	0.75	0.80	*S.scriba*	0.94	0.71	0.81
Av F1 score					0.75	Av F1 score			0.76
Sd F1 score					0.05	Sd F1 score			0.18
8 classes									
C.chromis	2730	854	0.79	0.67	0.73	*C.chromis*	0.62	0.96	0.75
D.sargus	2281	492	0.73	0.75	0.74	*D.sargus*	0.73	0.78	0.76
D.vulgaris	1011	251	0.75	0.64	0.69	*D. vulgaris*	0.71	0.16	0.27
S. scriba	152	48	0.81	0.75	0.78	*S.scriba*	0.97	0.60	0.74
S.dumerilii	870	172	0.89	0.89	0.83	*S. dumerilii*	0.88	0.66	0.75
D.annularis	434	195	0.85	0.72	0.78	*D. annularis*	0.87	0.51	0.65
O.melanura	691	184	0.80	0.80	0.62	*O.melanura*	0.93	0.14	0.24
C. julis	368	132	0.78	0.78	0.82	*C. julis*	0.87	0.82	0.84
Av F1 score					0.75	Av F1 score			0.63
Sd F1 score					0.07	Sd F1 score			0.23

Bounding boxes are extracted from 343 images.

previously identified (e.g., Saleh et al., 2022), it is necessary to develop models that can generalize their learning and perform well on new, unseen data samples, bridging the gap between DL and the requirements of image-based ecological monitoring (e.g. MacLeod et al., 2010; Christin et al., 2019; Aguzzi et al., 2020; Goodwin et al., 2022).

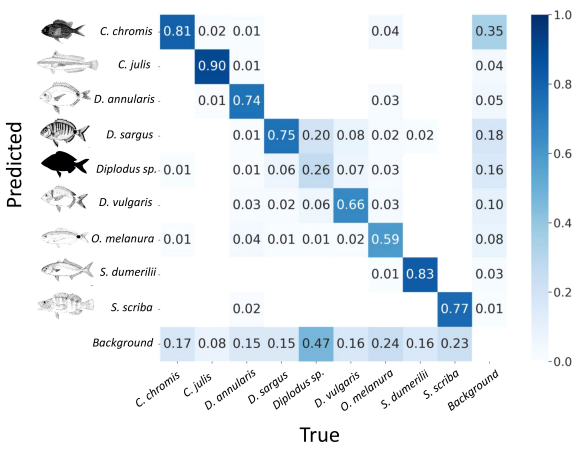


FIGURE 3
Confusion matrix based on YOLO results for eight species, background, and one genus (*Diplodus* sp) that could not be attributed to other congeneric species. Refer to the text for further explanations.

Related to the above, another common problem in classification is the imbalance of objects per class, as the DL model tends to weigh more heavily on the more abundant classes. Class-aware approaches have been proposed for fish classifications (Alaba et al., 2022). Beyond confirming that balancing improved classification in our datasets, we found that comparable results to an imbalanced dataset with an order of magnitude more training images could be obtained by carefully selecting images. Additionally, our results showed that avoiding training with images containing many small bounding boxes yields better precision and recall values on validation and test sets. The relation between object size and classification properties has been described previously, and it is recommended to separate the classification analyses as a function of object size (e.g., Connolly et al., 2022). However, to our best knowledge, this practice is not commonly used in fish ecology studies applying DL algorithms to underwater images. Overall, the fact that a model trained with a limited number of images performs relatively well across multiple test scenarios is a promising result for applications in ecological studies.

Recent reviews (e.g., Goodwin et al., 2022; Saleh et al., 2022) have concluded that for the application of DL methods to fish ecology research, transparent and reproducible research data and tools are necessary. This paper aims to contribute to this goal. There have been few studies on Mediterranean fish that have been experimental in nature (e.g., testing new network developments on a reduced number of species, such as Paraschiv et al., 2022 for a

few pelagic species). To increase the use of DL in this field, we concur with other authors that not only should common databases and reproducible methods be made available (e.g., Francescangeli et al., 2023), but also that more integrated engineers-ecologists interactions are institutionally needed (Logares et al., 2021). Additionally, statistical corrections to DL estimates must be developed (Connolly et al., 2021) and the use of lighter networks (e.g., Paraschiv et al., 2022) should become more common, as computer power may be a significant limitation for unplugged underwater devices (e.g., Lisani et al., 2012).

In summary, our research has discovered or reinforced several key findings that have important implications for fish ecology. Firstly,we found that using fast, single-step classifiers like YOLOv5, we can classify fishes in entire images cost-effectively, without the need for a two-step approach. Secondly, while having a large number of labeled fish images is important, a better approach may be to use a variety of backgrounds with a smaller, more carefully selected set of images. When selecting images, it is important to ensure that the bounding box fully captures the fish, and that the bounding box is not too small relative to the image. Lastly, we found that increasing the number of classes in the training dataset may lower overall classification metrics, but it may not significantly affect species with a high number of labels and can improve the identification of less abundant species.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/Supplementary Material.

Author contributions

IC and AA-E conceptualized the paper. AA-E, J-LL and JS ran the models. MP, HH, GV, MF and IC provided images. All authors contributed to the writing and interpretation of the results, lead by IC. IC and J-LL contributed to funding. All authors contributed to the article and approved the submitted version.

Funding

Project DEEP-ECOMAR. 10.13039/100018685-Comunitat Autonoma de les Illes Balears through the Direcció General de Política Universitària i Recerca with funds from the Tourist Stay Tax law ITS 2017-006 (Grant Number: PRD2018/26).

Acknowledgments

We thank Juan José Enseñat for his help in the acquisition and storage of images. The present research was carried out within the framework of the activities of the Spanish Government through the "María de Maeztu Centre of Excellence" accreditation to IMEDEA (CSIC-UIB) (CEX2021-001198-M) and the "Severo Ochoa Centre Excellence" accreditation to ICM-CSIC (CEX2019-000928-S) and the Research Unit Tecnoterra (ICM-CSIC/UPC).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1151758/full#supplementary-material

References

Aguzzi, J., Chatzievangelou, D., Company, J. B., Thomsen, L., Marini, S., Bonofiglio, F., et al. (2020). The potential of video imagery from worldwide cabled observatory networks to provide information supporting fish-stock and biodiversity assessment. *ICES J. Mar. Sci.* 77, 2396–2410. doi: 10.1093/icesjms/fsaa169

Aguzzi, J., Doya, C., Tecchio, S., De Leo, F. C., Azzurro, E., Costa, C., et al. (2015). Coastal observatories for monitoring of fish behaviour and their responses to environmental changes. *Rev. Fish Biol. Fish.* 25:463–83. doi: 10.1007/s11160-015-9387-9

Aguzzi, J., Mànuel, A., Condal, F., Guillén, J., Nogueras, M., del Rio, J., et al. (2011). The new seafloor observatory (OBSEA) for remote and long-term coastal ecosystem monitoring. *Sensors* 11, 5850–5872. doi: 10.3390/s110605850

Alaba, S. Y., Nabi, M. M., Shah, C., Prior, J., Campbell, M. D., Wallace, F., et al. (2022). Class-aware fish species recognition using deep learning for an imbalanced dataset. *Sensors* 22, 8268. doi: 10.3390/s22218268

Azzurro, E., Smeraldo, S., and D'Amen, M. (2022). Spatio-temporal dynamics of exotic fish species in the Mediterranean Sea: Over a century of invasion reconstructed. *Glob. Change Biol.* 28, 6268–6279. doi: 10.1111/gcb.16362

Bonofiglio, F., De Leo, F. C., Yee, C., Chatzievangelou, D., Aguzzi, J., and Marini, S. (2022). Machine learning applied to big data from marine cabled observatories: A case study of sablefish monitoring in the NE pacific. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.842946

Campos-Candela, A., Palmer, M., Balle, S., and Alós, J. (2018). A camera-based method for estimating absolute density in animals displaying home range behaviour. *J. Anim. Ecol.* 87, 825–837. doi: 10.1111/1365-2656.12787

Christin, S., Hervet, É., and Lecomte, N. (2019). Applications for deep learning in ecology. *Methods Ecol. Evol.* 10, 1632–1644. doi: 10.1111/2041-210X.13256

Coll, M., Piroddi, C., Steenbeek, J., Kaschner, K., Lasram, F. B. R., Aguzzi, J., et al. (2010). The biodiversity of the Mediterranean Sea: Estimates, patterns, and threats. *PLoS One* 5(8):e118. doi: 10.1371/journal.pone.0011842

Connolly, R. M., Fairclough, D. V., Jinks, E. L., Ditria, E. M., Jackson, G., Lopez-Marcano, S., et al. (2021). Improved accuracy for automated counting of a fish in baited underwater videos for stock assessment. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.658135

Connolly, R. M., Jinks, K. I., Herrera, C., and Lopez-Marcano, S. (2022). Fish surveys on the move: Adapting automated fish detection and classification frameworks for videos on a remotely operated vehicle in shallow marine waters. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.918504

Diaz-Gil, C., Smee, S. L., Cotgrove, L., Follana-Berná, G., Hinz, H., Marti-Puig, P., et al. (2017). Using stereoscopic video cameras to evaluate seagrass meadows nursery function in the Mediterranean. *Mar. Biol.* 164:137. doi: 10.1007/s00227-017-3169-y

Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., and Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: Optimizing animal ecology with deep learning. *Front. Mar. Sci.* 7. doi: 10.3389/fmars.2020.00429

Everingham, M., Van Gool, L., Williams, C., Winn, K., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* 88, 303–338. doi: 10.1007/s11263-009-0275-4

Follana-Berná, G., Arechavala-Lopez, P., Ramirez-Romero, E., Koleva, E., Grau, A., and Palmer, M. (2022). Mesoscale assessment of sedentary coastal fish density using vertical underwater cameras. *Fish. Res.* 253:106362. doi: 10.1016/j.fishres.2022.106362

Follana-Berná, G., Palmer, M., Campos-Candela, A., Arechavala-Lopez, P., Diaz-Gil, C., Alós, J., et al. (2019). Estimating the density of resident coastal firsh using underwater cameras: Accounting for individual detectability. *arXiv* 615:177–88. doi: 10.3354/meps12926

Francescangeli, M., Marini, S., Martínez, E., Del Río, J., Toma, D. M., Nogueras, M., et al. (2023). Image dataset for benchmarking automated fish detection and classification algorithms. *Sci. Data* 10, 1–13. doi: 10.1038/s41597-022-01906-1

Francescangeli, M., Sbragaglia, V., Del Río, J., Trullols, E., Antonijuan, J., Massana, I., et al. (2022). Long-term monitoring of diel and seasonal rhythm of dentex dentex at an artificial reef. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.837216

Goodwin, M., Halvorsen, K. T., Jiao, L., Knausgård, K. M., Martin, A. H., Moyano, M., et al. (2022). Unlocking the potential of deep learning for marine ecology: Overview, applications, and outlook. *ICES J. Mar. Sci.* 79, 319–336. doi: 10.1093/icesjms/fsab255

Hsiao, Y. H., Chen, C. C., Lin, S. I., and Lin, F. P. (2014). Real-world underwater fish recognition and identification, using sparse representation. *Ecol. Inform.* 23, 13–21. doi: 10.1016/j.ecoinf.2013.10.002

Kalogeiton, V., Ferrari, V., and Schmid, C. (2016). Analysing domain shift factors between videos and images for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 2327–2334. doi: 10.1109/tpami.2016.2551239

Knausgård, K. M., Wiklund, A., Sørdalen, T. K., Halvorsen, K. T., Kleiven, A. R., Jiao, L., et al. (2022). Temperate fish detection and classification: a deep learning based approach. *Appl. Intell.* 52, 6988–7001. doi: 10.1007/s10489-020-02154-9

Land, E. H., and McCann, J. J. (1971). Lightness and retinex theory. *Josa* 61, 1–11. doi: 10.1364/JOSA.61.000001

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 2278–2323. doi: 10.1109/5.726791

Li, D., and Du, L. (2022). *Recent advances of deep learning algorithms for aquacultural machine vision systems with emphasis on fish* (Netherlands: Springer). doi: 10.1007/s10462-021-10102-3

Lisani, J. L., Petro, A. B., Sbert, C., Álvarez-Ellacuría, A., Catalán, I. A., and Palmer, M. (2022). Analysis of underwater image processing methods for annotation in deep learning based fish detection. *IEEE Access* 10:130359–72. doi: 10.1109/ACCESS.2022.3227026

Logares, R., Alós, J., Catalán, I. A., Solana, A. C., del Campo, J., Ercilla, G., et al. (2021). "Oceans of big data and artificial intelligence," in *Ocean science challenges for 2030* (Madrid: CSIC), 163–179.

MacLeod, N., Benfield, M., and Culverhouse, P. (2010). Time to automate identification. *Nature* 467, 154–155. doi: 10.1038/467154a

Marini, S., Corgnati, L., Manotovani, C., Bastianini, M., Ottaviani, E., Fanelli, E., et al. (2018a). Automated estimate of fish abundance through the autonomous imaging device GUARD1 126, 72–75. doi: 10.1016/j.measurement.2018.05.035

Marini, S., Fanelli, E., Sbragaglia, V., Azzurro, E., Del Rio Fernandez, J., and Aguzzi, J. (2018b). Tracking fish abundance by underwater image recognition. *Sci. Rep.* 8:13748. doi: 10.1038/s41598-018-32089-8

Marrable, D., Barker, K., Tippaya, S., Wyatt, M., Bainbridge, S., Stowar, M., et al (2022). Accelerating species recognition and labelling of fish from underwater video with machine-assisted deep learning. *Front. Mar. Sci.* 9, 944584. doi: 10.3389/fmars.2022.944582

Mittal, S., Srivastava, S., and Jayanth, J. P. (2022). A survey of deep learning techniques for underwater image classification. *IEEE Trans. Neural Networks Learn. Syst*, 1–15. doi: 10.1109/TNNLS.2022.3143887

Moniruzzaman, M., Islam, S. M. S., Bennamoun, M., and Lavery, P. (2017). Deep Learning on Underwater Marine Object Detection: A Survey. *Adv. Concepts Intell. Vis. Syst. ACIVS 2017 Lect. Notes Comput. Sci.*, 10617. doi: 10.1007/978-3-319-70353-4_13

Ottaviani, E., Aguzzi, J., Francescangeli, M., and Marini, S. (2022). Assessing the image semantic drift at coastal underwater cabled observatories. *Front. Mar. Sci.* 9, 840088. doi: 10.3389/fmars.2022.840088

Paraschiv, M., Padrino, R., Casari, P., Bigal, E., Scheinin, A., Tchernov, D., et al. (2022). Classification of underwater fish images and videos *via* very small convolutional neural networks†. *J. Mar. Sci. Eng.* 10, 1–21. doi: 10.3390/jmse10060736

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). *You only look once: Unified, real- time object detection* in CVPR.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). "Faster r-CNN: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems* (United States: Microsoft Research), 91–99. Available at: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84960980241&partnerID=40&md5=18aaa500235b11fb99e953f8b227f46d.

Saleh, A., Sheaves, M., and Rahimi Azghadi, M. (2022). Computer vision and deep learning for fish classification in underwater habitats: A survey. *Fish Fish.* 23:977–99. doi: 10.1111/faf.12666

Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., et al. (2016). Fish species classification in unconstrained underwater environments based on deep learning. *Limnol. Oceanogr. Methods* 14, 570–585. doi: 10.1002/lom3.10113

Sarwar, S. S., Ankit, A., and Roy, K. (2020). Incremental learning in deep convolutional neural networks using partial network sharing. *IEEE Access* 8, 4615–4628.

Tan, M., and Le, Q. V. (2021). Smaller models and faster training. *arXiv* 2104: arXiv:2104.00298v3. doi: 10.48550/arXiv.2104.00298

Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., et al. (2018). A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecol. Inform.* 48, 238–244. doi: 10.1016/j.ecoinf.2018.09.007

Wang, Y., Song, W., Fortino, G., Qi, L. Z., Zhang, W., and Liotta, A. (2019). An experimental-based review of image enhancement and image restoration methods for underwater imaging. *IEEE Access* 7, 233–251. doi: 10.1109/ACCESS.2019.2932130

Wang, H., Sun, S., Bai, X., and Wang, J. (2023). A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes. *IEEE J. Ocean. Eng.*, 1–19. doi: 10.1109/JOE.2022.3226202

Wang, H., Sun, S., Wu, X., Li, L., Zhang, H., Li, M., et al. (2021). A YOLOv5 baseline for underwater object detection. *Ocean. Conf. Rec.*, 2021–2024. doi: 10.23919/OCEANS44145.2021.9705896

Zhou, J. C., Zhang, D. H., and Zhang, W. S. (2020). Classical and state-of-the-art approaches for underwater image defogging: a comprehensive survey. *Front. Inf. Technol. Electron. Eng.* 21, 1745–1769. doi: 10.1109/JOE.2018.2863961

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 25 April 2023
DOI 10.3389/fmars.2023.1174347

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Ning Wang,
Dalian Maritime University, China
Peng Ren,
China University of Petroleum (East China), China

*CORRESPONDENCE
Xiaodong Wang
wangxiaodong@ouc.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 26 February 2023
ACCEPTED 04 April 2023
PUBLISHED 25 April 2023

CITATION
Si G, Xiao Y, Wei B, Bullock LB, Wang Y and Wang X (2023) Token-Selective Vision Transformer for fine-grained image recognition of marine organisms. *Front. Mar. Sci.* 10:1174347. doi: 10.3389/fmars.2023.1174347

COPYRIGHT
© 2023 Si, Xiao, Wei, Bullock, Wang and Wang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Token-Selective Vision Transformer for fine-grained image recognition of marine organisms

Guangzhe Si[1], Ying Xiao[2], Bin Wei[3], Leon Bevan Bullock[4], Yueyue Wang[5] and Xiaodong Wang[4]*

[1]College of Electronic Engineering, Ocean University of China, Qingdao, Shandong, China, [2]School of Science, The Hong Kong University of Science and Technology, Hong Kong, Hong Kong SAR, China, [3]The Affiliated Hospital of Qingdao University/Shandong Key Laboratory of Digital Medicine and Computer Assisted Surgery, Qingdao University, Qingdao, Shandong, China, [4]College of Computer Science and Technology, Ocean University of China, Qingdao, Shandong, China, [5]Computing Center, Ocean University of China, Qingdao, Shandong, China

Introduction: The objective of fine-grained image classification on marine organisms is to distinguish the subtle variations in the organisms so as to accurately classify them into subcategories. The key to accurate classification is to locate the distinguishing feature regions, such as the fish's eye, fins, or tail, etc. Images of marine organisms are hard to work with as they are often taken from multiple angles and contain different scenes, additionally they usually have complex backgrounds and often contain human or other distractions, all of which makes it difficult to focus on the marine organism itself and identify its most distinctive features.

Related work: Most existing fine-grained image classification methods based on Convolutional Neural Networks (CNN) cannot accurately enough locate the distinguishing feature regions, and the identified regions also contain a large amount of background data. Vision Transformer (ViT) has strong global information capturing abilities and gives strong performances in traditional classification tasks. The core of ViT, is a Multi-Head Self-Attention mechanism (MSA) which first establishes a connection between different patch tokens in a pair of images, then combines all the information of the tokens for classification.

Methods: However, not all tokens are conducive to fine-grained classification, many of them contain extraneous data (noise). We hope to eliminate the influence of interfering tokens such as background data on the identification of marine organisms, and then gradually narrow down the local feature area to accurately determine the distinctive features. To this end, this paper put forwards a novel Transformer-based framework, namely Token-Selective Vision Transformer (TSVT), in which the Token-Selective Self-Attention (TSSA) is proposed to select the discriminating important tokens for attention computation which helps limits the attention to more precise local regions.

TSSA is applied to different layers, and the number of selected tokens in each layer decreases on the basis of the previous layer, this method gradually locates the distinguishing regions in a hierarchical manner.

Results: The effectiveness of TSVT is verified on three marine organism datasets and it is demonstrated that TSVT can achieve the state-of-the-art performance.

KEYWORDS

token-selective, self-attention, vision transformer, fine-grained image classification, marine organisms

1 Introduction

Fine-grained Image Classification (FIC) is a challenging task which utilizes subtle variations of the same species to differentiate the different subcategories, examples include birds (Van Horn et al., 2015), dogs (Khosla et al., 2011), and cars (Krause et al., 2013). Unlike general image classification, FIC requires sufficient attention being paid to the distinguishing features between the subcategories. There are a large number of highly similar fish and plankton in the ocean, and the classification of these subcategories (Li et al., 2019; Li et al., 2022) is conducive to the protection of marine ecology and biodiversity. However, the images of marine organisms are often taken in multi-angle and multi-scene situations, additionally, the background of marine life images is complex, which also increases the difficulty of recognition.

Recently, fine-grained image classification methods have made great progress due to the development of Deep Neural Networks (DNNs) (Simonyan and Zisserman, 2015; He et al., 2016; Liu et al., 2022; Shi et al., 2022; Wang et al., 2022). Strongly supervised fine-grained classification methods (Branson et al., 2014; Zhang et al., 2014; Wei et al., 2018) require labor-intensive labeling of images, so weakly supervised classification methods which rely only on category labels are now commonly preferred. CNN-based weakly supervised methods on fine-grained image classification can be mainly divided into localization methods and feature-encoding methods. Localization methods first locate the distinguishing regions and then extract features from these regions for classification. For example, some works (Ge et al., 2019; Liu et al., 2020) obtain the discriminating bounding boxes through Region Proposal Networks (RPNs) and then feed these regions into the backbone network for classification. However, the bounding boxes contain a lot of background areas with interfering information. Therefore, the discriminating regions localized by these methods are not precise enough. In addition, whilst the feature-encoding methods (Lin T.-Y. et al., 2015; Yu et al., 2018) make the output of the network change from semantic features to high-order features which can represent fine-grained information by means of feature fusion, the high-order features obtained by these methods have large dimensions, and the fine-grained information is not distinguishable.

Recently, Vision Transformer (ViT) (Dosovitskiy et al., 2021) has demonstrated potent performance on various visual tasks (Carion et al., 2020; Zheng et al., 2021; Guo et al., 2022). Specifically, in the task of image classification, a whole image is split into several patches, and each patch is converted into a token through linear projection. Then, the importance of each token is obtained through the Multi-Head Self Attention (MSA), and finally all of the tokens are combined according to the importance for classification. MSA in Transformer provides long-range dependency to enhance the interaction among image patches, so Transformer is able to locate subtle features and explore their relations from a large global scale perspective, whereas a traditional CNN has limited receptive fields and weak long range relationship abilities in very high layers with fixed-size convolutional kernels. ViT is therefore better suited to fine-grained classification tasks. In addition to the above advantages, ViT also has certain shortcomings, such as insufficient local sensing ability, tedious computation of MSA, and the need to consider the correlation among all tokens, our research is dedicated to improving these deficiencies.

Images of marine organisms are mostly taken from the bottom of the sea, the background of the images often contains reefs, corals and algae, which interferes with the recognition of the marine organisms themselves. A few images of marine life are taken from beaches, fishing boats and other scenes, the change of scenes also affects the identification of marine life. At the same time, due to the irresistible factors of camera angle and distance, images of the same subcategory show diverse global features, so paying too much attention to the global information is not conducive to correct classification. Examples of the three different scenarios are shown in Figure 1.

In this paper, to reduce the interference of intra-category diverse global information and useless background information, we propose a novel Token-Selective Vision Transformer (TSVT) for fine-grained image classification of marine organisms, which selects discriminative tokens layer by layer and gradually excludes interfering tokens. We propose a localized attention mechanism called Token-Selective Self-Attention (TSSA) to explore contextual information in discriminating regions and enhance the interaction amongst selected tokens. Influenced by the idea of clustering, for each discriminative token, only the other discriminative tokens related to it are selected for information interaction, then the class token integrates the information of these discriminative tokens for

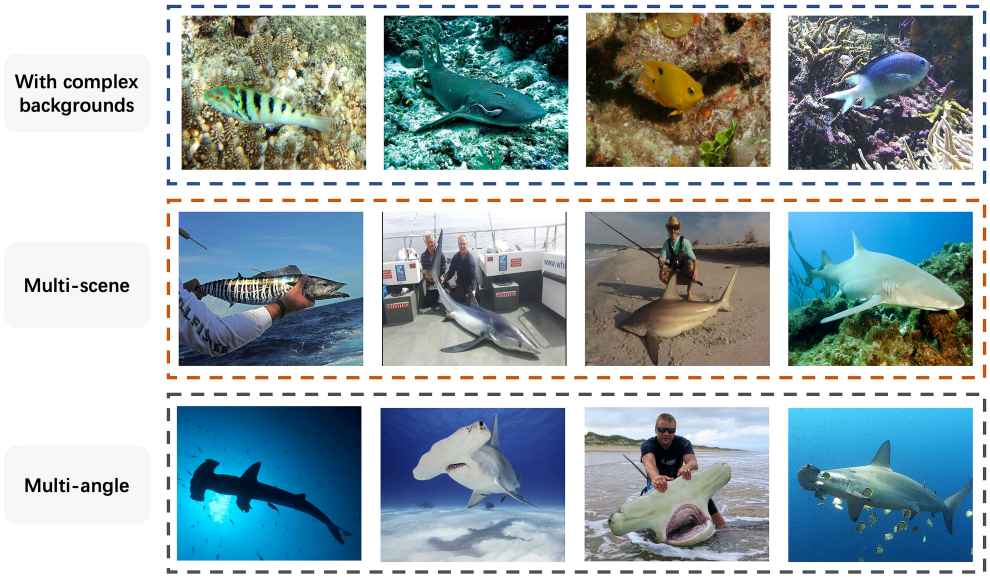


FIGURE 1
Some examples of marine life images. Three rows sequentially represent images with complex backgrounds, images of multiple scenes, and images of marine life taken from multiple angles.

classification. Finally, we verify the efficacy of TSVT for fine-grained image classification of marine organisms on three marine biological datasets.

In summary, our work has the following three contributions:

- We propose TSVT, a novel Vision Transformer framework for fine-grained image classification of marine organisms that excludes background interference and refines the range of distinguishing regions layer by layer.
- We propose Token-Selective Self-Attention (TSSA), which removes the interference of irrelevant tokens, and then establish the association of selected tokens in local regions and extract the most discriminative features.
- We conduct experiments on three different datasets to verify the effectiveness of our method, and show that TSVT achieves state-of-the-art performance. Additionally, we perform comparative experiments on TSSA's parameters to further explore the impact of applying TSSA to different layers, using different methods to select tokens and selecting different numbers of tokens on model performance.

2 Related work

2.1 Fine-grained image classification

2.1.1 CNN for fine-grained image classification

The fine-grained image classification methods based on CNN are mainly divided into two categories: localization methods and feature-encoding methods.

The basic idea of localization methods is to locate discriminative local regions first, and perform feature extraction on these regions, then cascade the extracted features and then again feed them to the sub-network for classification. Earlier localization methods (Zhang et al., 2014; Lin D. et al., 2015) rely on additional manual annotation information such as object bounding boxes and part annotation to help the network find the region with the most representative features. However, since such annotations are time-consuming and labor-intensive, more weakly supervised methods which only require image-level labels are preferred. Some methods (Ge et al., 2019; Liu et al., 2020) use RPN to obtain discriminative bounding boxes and input the selected feature regions into the network to capture local features. In addition, there are also methods to locate discriminative regions by utilizing an attention mechanism: RA-CNN (Fu et al., 2017) proposed Recurrent Attention to select a series of distinguishing regions for attention mapping in a coarse-to-fine manner; MA-CNN (Zheng et al., 2017) adopted a Multi-Attention CNN structure to obtain multiple distinguishing regions in parallel; MAMC (Sun et al., 2018) directed the generated attention features to categories to help better classification; NTS-Net (Yang et al., 2018) used a collaborative learning method to accurately identify the feature information regions.

Feature-encoding methods obtain richer fine-grained features for classification in the form of high-level feature interactions and the design of loss functions. As the most representative method for high-level feature interaction, B-CNN (Lin T.-Y. et al., 2015) used two deep convolutional networks to extract features from the same image, and then performed outer product operations on the feature vectors to obtain bilinear features for classification. However, the large feature dimensions of this method leads to a very large number of parameters, which is not easy to drive during training. To solve this problem, C-BCNN (Gao et al., 2016) adopted tensor sketches to reduce the dimensions of high-dimensional features.

Other methods attempt to capture features at higher levels to obtain a more distinguishable feature representation. HBP (Yu et al., 2018) combined the features of different layers through bilinear pooling, and finally concatenated them for classification. The loss function plays the role of a conductor's baton in Deep Learning and model learning is driven by it. In fine-grained image classification tasks, there are corresponding approaches to the design of loss functions: MaxEnt (Dubey et al., 2018) provided a training routine that maximizes the entropy of the output probability distribution; MC-Loss (Chang et al., 2020) focused on different local areas of each channel in the feature map, which is more conducive to feature learning.

2.1.2 ViT for fine-grained image classification

Transformer (Vaswani et al., 2017) was first applied to solve the sequence to sequence problem in Natural Language Processing (NLP) and has achieved better results than both convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Subsequently, Transformer has been widely used in the field of computer vision. ViT (Dosovitskiy et al., 2021) was the first transformer-based model for image classification, which splits images into a number of patches and inputs them to the transformer layer, and then establishes the association between different patches with the help of MSA, the classification is finally carried out by using the class token. TransFG (He et al., 2022) was the first to verify the effectiveness of vision Transformer on fine-grained visual classification. The input of its last layer is the class token and some important tokens representing distinguishing features rather than all of the tokens. In addition, RAMS-Trans (Hu et al., 2021) locates and extracts discriminative areas based on attention weights, and then re-inputs them into ViT for classification using multi-scale features.

In this paper, we propose TSSA, which allows each token to select its own relevant tokens according to the attention weights for attention computation. We integrate the one-to-one selection of each token into the attention computation. Furthermore, we apply TSSA to different layers of ViT to narrow the selection range layer by layer, so as to gradually refine the distinguishing features, yielding the major difference between our work and previous methods.

2.2 Underwater image classification

Due to the influence of the complex imaging environment in the ocean, the underwater images appear blurred, low contrast and low resolution, therefore various image preprocessing methods (Qi et al., 2022; Zhou et al., 2022; Zhou et al., 2023a; Zhou et al., 2023b) such as image enhancement and image restoration are used first to improve classification results. Recently, significant progress has been made in underwater classification, thanks to the influence of deep learning and the creation of several methods for underwater organism detection (Chen et al., 2021; Wang et al., 2023a; Wang et al., 2023b). The research on underwater biological image classification can be mainly divided into two aspects, one is the learning of biological features, the other is the feature fusion of different levels or types. For the feature acquisition methods, the earlier artificial methods (Alsmadi et al., 2010; Alsmadi et al., 2011) were only effective for specific datasets or scenarios, subsequently universal methods based on deep learning were adopted to learn various features. DeepFish (Qin et al., 2016) first extracted the fish regions using matrix decomposition, and then refined and learned these regional features by Principal Components Analysis (PCA) (Jackson, 1993) and CNN respectively. MCNN (Prasenan and Suriyakala, 2023) segmented fish images by the firefly algorithm and extracted features from the segmented parts. However, these methods require a large amount of computation, therefore, to maintain the balance between classification effect and cost, a number of efficient improved CNN networks were proposed: FDCNet (Lu et al., 2018) used filtering deep convolutional neural networks to classify deep-sea species; deconvolutional neural network was applied to different squid classification (Hu et al., 2020). In addition, in order to solve the noise background problem, AdaFish (Zhang et al., 2022) adopted adversarial learning to reduce the interference of background on classification.

Some methods (Kartika and Herumurti, 2016; Gomez Chavez et al., 2019) have obtained some limited improvement in classification accuracy by learning only a single feature such as fish color or coral texture, therefore combining multi-level or multi-part information to complete classification is another direction of underwater image classification. One method (Cui et al., 2018) integrated the texture and shape features of plankton to improve CNN performance; another method (Mathur et al., 2020) combined the characteristics of different parts of fish through cross convolutional layer pooling for prediction; whilst yet another method used a multi-level residual network (Prasetyo et al., 2022) which fused high and low level information through depth separable convolution was also proposed and achieved a good classification effect.

3 Methodology

3.1 Preliminary: vision transformer

The inputs of ViT are a sequence of serialized tokens. First, an image with resolution $H \times W$ is first split into fixed-size patches x_p, each of size $P \times P$, so the number of patches N is equal to $\frac{H}{P} \times \frac{H}{W}$. Each patch is transformed into a token x_{pt} by a patch embedding layer consisting of linear projection. In addition to patch tokens, there is a dedicated class token x_{cls} for final classification in the classification task. So all tokens include patch tokens and the class token. The above tokens only contain pixel information, and position encoding adds corresponding position information x_{pos} to each token to determine the position of each patch in the original image. All tokens are then fed into the transformer encoder, and the inputs of the transformer encoder x_0 are represented in Eq. 1:

$$x_0 = [x_{cls}; x_{pt}^1; x_{pt}^2; \ldots; x_{pt}^N] + x_{pos} . \quad (1)$$

Transformer encoder is the core of ViT and contains l transformer layers of MSA and Multi-Layer Perceptron (MLP) blocks, as well as residual connections after every block. The output of the l_{th} layer is represented as follows:

$$x_l^* = \mathrm{MSA}(\mathrm{LN}(x_{l-1})) + x_{l-1} \tag{2}$$

$$x_l = \mathrm{MLP}(\mathrm{LN}(x_l^*)) + x_l^*, \tag{3}$$

where $x_{l-1}$and x_l denote the encoded image representation of the $l-1_{th}$ and l_{th} transformer layers, x_l^* is the output of the MSA block after residual connection, LN represents layer normalization, and the class token of the last transformer layer is used for category prediction through MLP.

3.2 Overall architecture

Marine life images of the same subcategories present different global information such as posture and viewpoint, so an over-reliance on global information and a lack of attention to local information are not conducive to the correct classification. In addition, due to the complexity of the seabed environment, images of marine organisms often contain complex backgrounds such as reefs and corals, which will also affect the identification of marine organisms. In order to address the above issues, we first consider eliminating the interference of irrelevant factors such as the background, and locating the marine organisms themselves, then further locating the distinguishing areas. In this manner we propose TSVT, which selects tokens layer by layer for more accurate classification. By doing so, the number of tokens selected by the latter layer is further reduced on the basis of the preceding layer so as to more accurately refine the distinguishing areas and reduce the computational cost. To this end, we design a local attention module named TSSA, in which distinguishing tokens only interact with the other distinguishing tokens selected according to the attention weights, and the interference of background tokens is eliminated to obtain the purest distinguishing feature information for classification with the class token.

The framework of our TSVT is shown in Figure 2, where, the first eight transformers remain unchanged according to the settings of ViT, while the last four layers are Token-Selective Transformer Layer (TS Transformer Layer). It is different from the standard transformer layer in that it replaces the original MSA with TSSA. The number of tokens selected in each layer is different, and the local scope of attention is also different. The class token of the last layer aggregates the most discriminating features in the local regions and completes category prediction through MLP.

3.3 Token-selective self-attention

Fine-grained image classification requires focusing on local discriminating regions, but the complex background of marine biological images interferes with accurate localization of these regions. To solve the above issue, we propose to eliminate the interference of background tokens to the greatest extent and apply local attention to the selected important discriminating tokens.

All tokens can be divided into two categories: discriminating region tokens that play a positive effect in classification and background interfering tokens that play a negative effect in classification. Discriminating region tokens and background

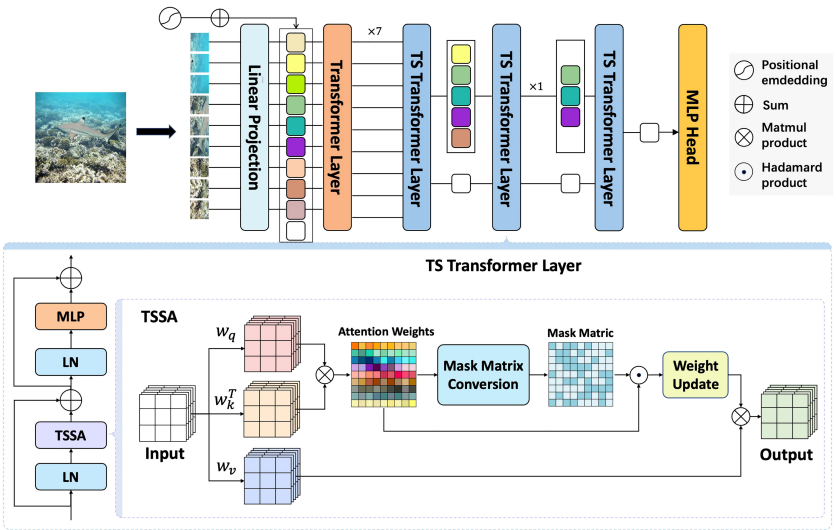


FIGURE 2
The framework of our proposed TSVT and the details of our designed TSSA. An image is first split into a number of patches, each of which is mapped into a feature vector by Linear Projection and combined with learnable position embedding. Contextual links between tokens are then established in the Transformer Layers, and the selection of tokens representing the discriminating regions is performed layer by layer in the latter four TS Transformer Layers with the number of selected tokens in each layer decreasing from the previous layers. In the TS Transformer Layer, TSSA is a sparse selective attention mechanism that generates a mask based on the similarity between tokens so as to limit the attention computation between non-relevant tokens.

tokens are clustered separately for information interaction in TSSA to ensure that discriminating tokens are no longer mixed with the interference information of background tokens, and then the class token integrates the information of distinctive tokens for the final classification.

The correlation between tokens can be reflected by attention weights. Previous work (Wang et al., 2021; He et al., 2022) has proved that attention weights can be a good indicator for token selection. The attention weights of each head in each transformer layer $A \in \mathbb{R}^{(N+1)\times(N+1)}$ can be written as follows:

$$A = \text{softmax}(\frac{Q \cdot K^T}{\sqrt{d_k}}) = [a_0, a_1, a_2 \dots a_N], \tag{4}$$

$$a_i = [a_{i,0}, a_{i,1}, a_{i,2} \dots \dots a_{i,N}], i \in (0, N)\,. \tag{5}$$

According to the attention weights, the information of the token is weighted and summed to obtain the calculation result of the attention symbolized as *Attention*. The following formula is the calculation process of MSA:

$$Attention = \text{softmax}(\frac{Q \cdot K^T}{\sqrt{d_k}}) \cdot V, \tag{6}$$

where Q, K and V are all obtained by the linear transformations of tokens, all of which represent information about the token itself; d_k represents the dimensionality of K; softmax is a normalized exponential function; a_{ij} represents the degree of correlation between the i_{th} token and the j_{th} token, that is, token i as Q and token j as K for the calculation in Eq. 4; a_i represents the set of correlation degrees between the i_{th} token and all tokens; and $\cdot$ represents the general matrix product.

Only the largest m elements in each row of attention weights are selected, the selected elements remain unchanged, and the remaining unselected elements are all set to zero, thus generating new selective attention weights, which represent the degree of correlation between each token and its most relevant m tokens. In the computation of attention, the distinguishing tokens interact with each other and the distinguishing features are strengthened.

In the implementation, to ensure parallel computing, a mask matrix M with the same shape as the attention weights is first generated, we set the m_{th} largest element α_i in each row of attention weights as the threshold to determine whether the elements at different positions of mask matrix are one or zero. The process of mask matrix conversion is represented as:

$$M_{(i,j)} = \begin{cases} 1 & A_{(i,j)} \geq \alpha_i, \\ 0 & otherwise, \end{cases} \tag{7}$$

where (i, j) represents the position of each element in the mask matrix the and attention weights in $(n + 1) \times (n + 1)$ positions.

Then the selective attention weights A_s are obtained by computing the Hadamard product of the mask matrix and the attention weights, as follows:

$$A_s = A \odot M, \tag{8}$$

where $\odot$ is the calculation symbol for Hadamard product.

Without changing the relevance of the different tokens, we further update the elements a^s in the selective attention weights so that the sum of the elements in each row is equal to one, which further increases the proportion of discriminative information in the class token. Take the first row of A_s as an example, each element of this row $a^{s'}$ is computed as:

$$a^{s'}_{0,i} = \frac{a^s_{0,i}}{\sum_{j=1}^{N} a^s_{0,j}}\,. \tag{9}$$

The new selective attention weights $A_s^{'}$ represent the correlation between tokens in local areas, and then after the calculation in Eq. 10, the information between these tokens interacts and the output Z of TSSA is obtained. In the final TS Transformer Layer, the class token combines the token information through MLP for category predictions.

$$Z = A_s^{'} \cdot V\,. \tag{10}$$

The selective attention weights of each token-selective transformer layer are updated on the basis of the previous layer, and the number of selected tokens m of each layer is gradually reduced narrowing and refining the distinguishing feature regions layer by layer.

We apply TSSA to the deep layers of the model without destroying the globality of the shallow layers, and the local information based on the global basis is extracted for classification. Starting from the first token-selective transformer layer, the distinguishing tokens only aggregate important tokens related to them, so that the class token associated with these distinguishing tokens can minimize the interference of the background tokens. Our model is actually a trade-off between globality and locality, on the basis of not losing the globality, it can accurately locate the discriminating area and extract local features.

4 Experiments

In this section, we mainly introduce the experimental process and analyze the experiment results. First, we introduce the three marine biological datasets used in experiments, and briefly introduce the specific settings. Then, we verify the efficacy of TSVT by ablation study and analyze the experiment results.

4.1 Datasets

We validated the effectiveness of TSVT on three datasets of marine organisms, namely ASLO-Plankton (Sosik and Olson, 2007), Sharks[1], and WildFish (Zhuang et al., 2018). ASLO-Plankton consists of 22 categories of marine plankton images, its training set is unbalanced, and the number of images in different subcategories conforms to the long-tail distribution; Sharks contains images of 14 shark species, where the background of the images is complex and the differences between images are subtle; WildFish is a large-scale marine fish dataset with 1000 categories

and 54459 images in total, and we randomly select images of 200 categories from WildFish to form a new dataset WildFish200. The statistics of the three datasets are shown in Table 1.

4.2 Implementation details

The input image size of the ASLO-Plankton, WildFish200 and Sharks datasets is 448×448 pixels, the size of each patch is 16×16. We set the batch size on the three datasets to 8. SGD optimizer is employed with a momentum of 0.9. The learning rate is initialized as 0.03 and we adopt cosine annealing as the scheduler of optimizer. TSVT imports the pre-trained ViT-B_16 on ImageNet21k as the pretrained model. We complete the construction of the whole model using PyTorch and run all experiments on four NVIDIAGTX 1070 GPUs in one computer.

4.3 Comparison with the state-of-the-arts

Our method performs on par with a number of CNN-based methods: B-CNN (Lin T.-Y. et al., 2015), NTS-Net (Yang et al., 2018), TASN (Zheng et al., 2019), MC Loss (Chang et al., 2020), and the recent transformer variants: ViT (Vaswani et al., 2017), RAMS-Trans (Hu et al., 2021), TransFG (He et al., 2022) on ASLO-Plankton, Sharks and WildFish200. The experiment results are shown in Table 2. It can be seen from the results that ViT-based methods have a higher classification accuracy than CNN-based methods. Meanwhile, TSVT reaches 74.3%, 90.4% and 94.7% top-1 accuracy on ASLO-Plankton, Sharks and WildFish200 respectively, which achieves higher accuracy in the identification of marine organisms compared with other methods. The main reason for the improvement is that our method further eliminates background interference, accurately locates the discriminating areas, thus enlarging the differences between categories.

TABLE 1 Statistics of ASLO-Plankton, Sharks and WildFish200 datasets.

Dataset	Classes	Training	Testing
ASLO-Plankton	22	743	3300
Sharks	14	743	749
WildFish200	200	7929	3523

4.4 Ablation study

We verify the efficacy of our proposed TSSA on the three datasets, and further explore the impact of applying TSSA to different layers, using different methods to select tokens and selecting different numbers of tokens on model performance.

4.4.1 Impact of applying TSSA to different layers

We applied TSSA to the shallow layers (1-4), middle layers (5-8) and deep layers (9-12) of TSVT respectively, to explore the influence of token selection in different layers on model performance. The experiment results in the Table 3 show that applying TSSA to the deep layers achieves the best performance, whilst starting token selection in the shallow layers achieves worse performance. A possible reason is that the attention weights in shallow layers cannot highlight the key points that should be paid attention to, which is not enough to be used as the indicator for selecting tokens. On the contrary, with the deepening of layers, the feature information is accumulated, and the model starts to notice discriminating regions. At this time, further eliminating background and other interference can make the discriminative local features account for a larger proportion of final features used for classification. Global information needs to be strengthened by layers of accumulation, premature destruction of the association among all tokens at shallow layers is not conducive to extracting global features of the model. Therefore, establishing the association among all tokens at the shallow layers first, and then discarding some tokens at the deep layers is a trade-off between global information and local information, which is beneficial for classification.

When TSSA is applied to the deep layers, the classification performance of the model is improved. So we further explore the impact of applying TSSA to different deep layers. In different

TABLE 2 Comparison of TSVT and state-of-the-art methods on three datasets of marine organisms.

Method	Backbone	Accuracy(%)		
		ASLO-Plankton	Sharks	WildFish200
B-CNN	VGG-16	61.9	76.2	82.1
NTS-Net	ResNet-50	69.4	84.5	87.3
TASN	ResNet-50	70.0	85.2	88.7
MC Loss	ResNet-50	69.6	86.3	86.2
ViT	ViT-B_16	72.6	88.9	93.5
RAMS-Trans	ViT-B_16	73.1	89.2	93.8
TransFG	ViT-B_16	73.7	89.1	94.1
TSVT (Ours)	ViT-B_16	74.3	90.4	94.7

TABLE 3 Ablative experiments on applying TSSA to different layers.

Layers	ASLO-Plankton	Sharks	WildFish200
1-4	69.5	85.7	92.5
5-8	71.0	88.9	93.4
9-12	74.3	90.4	94.7

ablative experiments, the number of selected tokens decreases from the first TS transformer layer and the number in the final layer remains the same. As shown from the Table 4, the classification accuracy is constantly improved with the increase of the number of layers. The best effect is achieved when TSSA is applied to layers 8-12, which indicates that the model has been able to accurately locate the distinguishing regions from the 8_{th} layer, and the smaller the reduction of tokens between layers, the better the classification performance of the model.

4.4.2 Impact of the number of selected tokens

TSVT performs token selection layer by layer, and the latter layer continues to select tokens based on those selected in the previous layer in order to pinpoint discriminative regions hierarchically. In the experiments, we set a parameter p about the selection proportion to indicate the number of selected tokens, which is the ratio of the number of selected tokens to the number of all tokens. We studied the influence of the parameter p on the model, and the experiment results are shown in Table 5. When p is 0.7, TSVT achieves the best performance on the three datasets. As the p value increases from 0.7 to 0.9, the accuracy decreases, probably because too many background tokens are not discarded, leading to discriminative information being mixed with interference information. When the value of p is smaller than 0.7, the accuracy also decreases, which is because the number of tokens is too small and too many important tokens are discarded. When the value of p is smaller than 0.2, the number of selected tokens in the last layer is less than 1, so we did not conduct related experiments. In conclusion, TSVT is sensitive to the number of selected tokens.

4.4.3 Impact of token-selective methods

We select important tokens according to the attention weights. In this part, we select tokens randomly at layers 9-12 with the selection ratio $p = 0.7$ for comparison, which further verifies the efficacy of our selection method. The two methods of random selection and selection according to attention weights are respectively applied in TSSA for experiments. As can be seen from Table 6, the accuracy of the former method decreases by

TABLE 4 Ablative experiments on applying TSSA to different deep layers.

Layers	ASLO-Plankton	Sharks	WildFish200
12	73.2	88.9	93.7
11-12	73.6	89.4	94.3
10-12	73.4	90.0	94.3
9-12	74.3	90.4	94.7

TABLE 5 Ablation experiments on the number of selected tokens.

p	ASLO-Plankton	Sharks	WildFish200
0.9	73.2	89.1	93.8
0.8	72.9	89.4	94.4
0.7	74.3	90.3	94.7
0.6	72.9	89.9	94.3
0.5	72.1	88.5	93.7
0.4	71.3	88.1	91.1
0.3	69.2	87.9	88.7

3.6%, 1.6%, 0.6% respectively compared with the latter method (ours) on the three datasets. The reason is that some important distinguishing tokens are discarded in the process of random selection, and some tokens that interfere with classification accuracy may be selected for classification.

4.4.4 Visualization

In order to further verify the effectiveness of our method in locating discriminating regions, we use Grad-CAM (Selvaraju et al., 2017) to visualize the attention map generated from the attention weights of the final layer in TSVT and compare them with ViT. As shown in Figure 3, for images with complex backgrounds, ViT is easily affected by these backgrounds and focuses on objects irrelevant to classification, such as reefs and corals, while after excluding these interferences, TSVT easily locates marine organisms and their most distinctive features, such as patterns and spots on the fish. Taking the image in the first row and column as an example, ViT considers the human head as the discriminative region, while our method can accurately use the effective information of the hammerhead shark's head information to predict the category. In addition, for images where the fish are visually small due to the long shooting distance, TSVT can locate the positions of the small targets more accurately, whereas ViT sometimes cannot achieve such high precision positioning.

5 Conclusion

In this paper, in order to exclude the influence of the complex background of the seabed and accurately locate discriminating features, we propose a novel framework called TSVT for fine-grained image classification of marine organisms, which achieves the best performance on the three marine organism datasets compared with other state-of-the-art works. We propose a local attention mechanism called TSSA that excludes interfering tokens.

TABLE 6 . Ablative experiments on token-selective methods.

Selection Methods	ASLO-Plankton	Shark	WildFish200
random	70.7	88.8	94.1
max	74.3	90.4	94.7

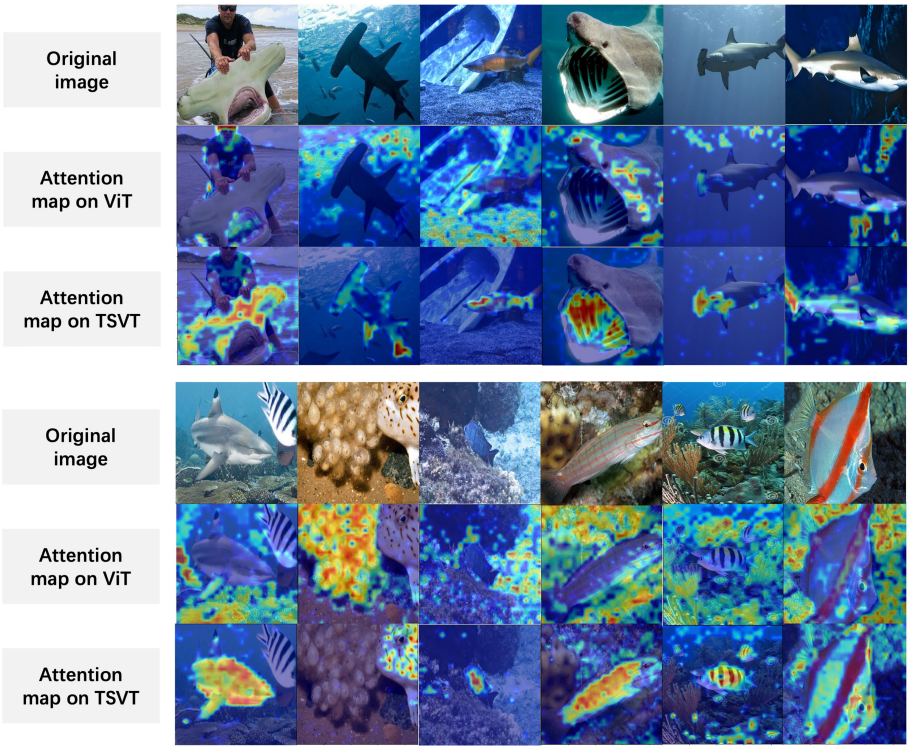


FIGURE 3
Visualization results on marine biological datasets, in which the first and fourth rows are six images in Sharks and WildFish datasets, the second and fifth rows are visualization of six images in the two datasets on ViT, and the third and sixth rows are visualization on TSVT.

Each discriminating token interacts with other discriminating tokens in the local area to extract positive fine-grained features to the greatest extent. Then, we explore the impact of applying TSSA to different layers, the number of selected tokens and token-selective methods on the performance of TSVT.

However, we still select key tokens through attention weights, which has the limitation that it must be applied to deep layers to ensure the reliability of the selection. Meanwhile, the number of key tokens in each image is not the same, so selecting tokens through more effective learning methods as well as setting learnable parameters to control the number of selected tokens is the future direction.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

GS, YW, and XW designed the study and wrote the draft of the manuscript with contributions from YX and BW. BW and LB collected the marine fish image datasets. YW and XW devised the method. GS and YX performed the experiments. All authors contributed to the experimental analysis and manuscript writing. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation of China (No. 32073029) and the Key Project of Shandong Provincial Natural Science Foundation (No. ZR2020KC027).

Acknowledgments

We thank the Intelligent Information Sensing and Processing Lab at Ocean University of China for their computing servers and collaboration during experiments. We kindly thank the Editor Dr. Xuemin Cheng for her efforts to handle this manuscript and all the reviewers for their constructive suggestions that helped us to improve our present manuscript.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alsmadi, M. K., Omar, K. B., Noah, S. A., Almarashdeh, I., Al-Omari, S., Sumari, P., et al. (2010). Fish recognition based on robust features extraction from size and shape measurements using neural network. *Comput. Sci.* 4, 1085–1091. doi: 10.3844/jcssp.2010.1088.1094

Alsmadi, M. K., Omar, K. B., Noah, S. A., et al. (2011). Fish classification based on robust features extraction from color signature using back-propagation classifier. *Comput. Sci.* 4, 52–58. doi: 10.3844/jcssp.2011.52.58

Branson, S., Van Horn, G., Belongie, S., and Perona, P. (2014). Bird species categorization using pose normalized deep convolutional nets. in. *Br. Mach. Vision Conference.* 2, 1–14.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. in. *Eur. Conf. Comput. Vision.* 2, 213–229. doi: 10.1007/978-3-030-58452-8_13

Chang, D., Ding, Y., Xie, J., Bhunia, A. K., Li, X., Ma, Z., et al. (2020). The devil is in the channels: mutual-channel loss for fine-grained image classification. *IEEE Trans. Image Process.* 4 (8), 4683–4695. doi: 10.1109/TIP.2020.2973812

Chen, T., Wang, N., Wang, R., Zhao, H., and Zhang, G. (2021). One-stage CNN detector-based benthonic organisms detection with limited training dataset. *Neural Networks* 4, 247–259. doi: 10.1016/j.neunet.2021.08.014

Cui, J., Wei, B., Wang, C., Yu, Z., Zheng, H., Zheng, B., et al. (2018). Texture and shape information fusion of convolutional neural network for plankton image classification. in. *OCEANS.* 5, 1–5. doi: 10.1109/OCEANSKOBE.2018.8559156

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). "An image is worth 16x16 words: transformers for image recognition at scale," in *In International Conference on Learning Representations*, Vol. 2, 4. 1–22.

Dubey, A., Gupta, O., Raskar, R., and Naik, N. (2018). Maximum-entropy fine grained classification. in. *Adv. Neural Inf. Process. Systems.* 4, 1–11.

Fu, J., Zheng, H., and Mei, T. (2017). "Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. in," in *IEEE Conference on Computer Vision and Pattern Recognition*, 3, 4438–4446.

Gao, Y., Beijbom, O., Zhang, N., and Darrell, T. (2016). "Compact bilinear pooling. in," in *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 4. 317–326.

Ge, W., Lin, X., and Yu, Y. (2019). "Weakly supervised complementary parts models for fine-grained image classification from the bottom up. in," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2 (3), 3034–3043.

Gomez Chavez, A., Ranieri, A., Chiarella, D., Zereik, E., Babić, A., and Birk, A. (2019). CADDY underwater stereo-vision dataset for human–robot interaction (HRI) in the context of diver activities. *Mar. Sci. Eng.* 5, 1–14. doi: 10.3390/jmse7010016

Guo, Z., Gu, Z., Zheng, B., Dong, J., and Zheng, H. (2022). "Transformer for image harmonization and beyond," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2,

He, J., Chen, J.-N., Liu, S., Kortylewski, A., Yang, C., Bai, Y., et al. (2022). "TransFG: a transformer architecture for fine-grained recognition," in *In AAAI Conference on Artificial Intelligence*, 4 (6–8), 852–860.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition. in," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2, 770–778.

Hu, Y., Jin, X., Zhang, Y., Hong, H., Zhang, J., He, Y., et al. (2021). "RAMS-trans: recurrent attention multi-scale transformer for fine-grained image recognition," in *In ACM International Conference on Multimedia*, 4 (8), 4239–4248.

Hu, J., Zhou, C., Zhao, D., Zhang, L., Yang, G., and Chen, W. (2020). A rapid, low-cost deep learning system to classify squid species and evaluate freshness based on digital images. *Fisheries Res.* 4, 1–10. doi: 10.1016/j.fishres.2019.105376

Jackson, D. A. (1993). Stopping rules in principal components analysis: a comparison of heuristical and statistical approaches. *Ecology* 4, 2204–2214. doi: 10.2307/1939574

Kartika, D. S. Y., and Herumurti, D. (2016). "Koi fish classification based on HSV color space," in *In International Conference on Information Communication Technology and Systems*, Vol. 5. 96–100.

Khosla, A., Jayadevaprakash, N., Yao, B., and Li, F.-F. (2011). Novel dataset for fine-grained image categorization:stanford dogs. in. *CVPR Workshop Fine-Grained Visual Categorization.* 2, 1–2.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013). "3D object representations for fine-grained categorization. in," in *IEEE International Conference on Computer Vision*, 2, 554–561.

Li, J., Xu, W., Deng, L., Xiao, Y., Han, Z., and Zheng, H. (2022). Deep learning for visual recognition and detection of aquatic animals: a review. *Rev. Aquaculture* 2, 1–24. doi: 10.1111/raq.12726

Li, J., Xu, C., Jiang, L., Xiao, Y., Deng, L., and Han, Z. (2019). Detection and analysis of behavior trajectory for sea cucumbers based on deep learning. *IEEE Access* 2, 18832–18840. doi: 10.1109/ACCESS.2019.2962823

Lin, T.-Y., RoyChowdhury, A., and Maji, S. (2015). "Bilinear CNN models for fine-grained visual recognition," in *IEEE International Conference on Computer Vision*, Vol. 2 (4–8), 1449–1457.

Lin, D., Shen, X., Lu, C., and Jia, J. (2015). "Deep LAC: deep localization, alignment and classification for fine-grained recognition," in *In IEEE Conference on Computer Vision and Pattern Recognition*, 3, 1666–1674.

Liu, C., Xie, H., Zha, Z.-J., Ma, L., Yu, L., and Zhang, Y. (2020). "Filtration and distillation: enhancing region attention for fine-grained visual categorization. in," in *AAAI Conference on Artificial Intelligence*, 2 (3), 11555–11562.

Liu, P., Zhang, C., Qi, H., Wang, G., and Zheng, H. (2022). "Multi-attention DenseNet: a scattering medium imaging optimization framework for visual data pre-processing of autonomous driving systems," in *IEEE Transactions on Intelligent Transportation Systems*, 2, 25396–25407.

Lu, H., Li, Y., Uemura, T., Ge, Z., Xu, X., He, L., et al. (2018). FDCNet: filtering deep convolutional network for marine organism classification. *Multimedia Tools Appl.* 4, 21847–21860. doi: 10.1007/s11042-017-4585-1

Mathur, M., Vasudev, D., Sahoo, S., Jain, D., and Goel, N. (2020).). crosspooled fishnet: transfer learning based fish species classification model. *Multimedia Tools Appl.* 5, 31625–31643. doi: 10.1007/s11042-020-09371-x

Prasenan, P., and Suriyakala, C. (2023). Novel modified convolutional neural network and FFA algorithm for fish species classification. *Combinatorial Optimization* 4, 1–23. doi: 10.1007/s10878-022-00952-0

Prasetyo, E., Suciati, N., and Fatichah, C. (2022). Multi-level residual network vggnet for fish species classification. *King Saud Univ. - Comput. Inf. Sci.* 5, 5286–5295. doi: 10.1016/j.jksuci.2021.05.015

Qi, Q., Li, K., Zheng, H., Gao, X., Hou, G., and Sun, K. (2022). SGUIE-net: semantic attention guided underwater image enhancement with multi-scale perception. *IEEE Trans. Image Process.* 4, 6816–6830. doi: 10.1109/TIP.2022.3216208

Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. (2016). DeepFish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing* 4, 49–58. doi: 10.1016/j.neucom.2015.10.122

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). "Grad-cam: visual explanations from deep networks *via* gradient-based localization," in *In IEEE International Conference on Computer*, 10, 618–626.

Shi, Z., Guan, C., Li, Q., Liang, J., Cao, L., Zheng, H., et al. (2022). "Detecting marine organisms *via* joint attention-relation learning for marine video surveillance," in *IEEE Journal of Oceanic Engineering*, 2, 959–974.

Simonyan, K., and Zisserman, A. (2015). "Very deep convolutional networks for large-scale image recognition. in," in *International Conference on Learning Representations*, 3, 1–14.

Sosik, H. M., and Olson, R. J. (2007). Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. *Limnology Oceanography: Methods* 8, 204–216. doi: 10.4319/lom.2007.5.204

Sun, M., Yuan, Y., Zhou, F., and Ding, E. (2018). "Multi-attention multi-class constraint for fine-grained image recognition. in," in *European Conference on Computer Vision*, 2, 805–821.

Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., et al. (2015). "Building a bird recognition app and large scale dataset with citizen scientists: the fine print in fine-grained dataset collection. in," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1, 595–604.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. in. *Adv. Neural Inf. Process. Systems.* 4 (8), 1–11.

Wang, N., Chen, T., Liu, S., Wang, R., Karimi, H. R., and Lin, Y. (2023b). Deep learning-based visual detection of marine organisms: a survey. *Neurocomputing* 1–32, 4. doi: 10.1016/j.neucom.2023.02.018

Wang, H., Sun, S., Bai, X., Wang, J., and Ren, P. (2023a). "A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes," in *IEEE Journal of Oceanic Engineering*, 4, 1–19.

Wang, N., Wang, Y., and Er, M. J. (2022). Review on deep learning techniques for marine object recognition: architectures and algorithms. *Control Eng. Pract.* 118, 1–18 2. doi: 10.1016/j.conengprac.2020.104458

Wang, J., Yu, X., and Gao, Y. (2021). Feature fusion vision transformer for fine-grained visual categorization. *arXiv preprint arXiv* :2107.02341 6.

Wei, X.-S., Xie, C.-W., Wu, J., and Shen, C. (2018). Mask-CNN: localizing parts and selecting descriptors for fine-grained bird species categorization. *Pattern Recognition* 704–714, 2. doi: 10.1016/j.patcog.2017.10.002

Yang, Z., Luo, T., Wang, D., Hu, Z., Gao, J., and Wang, L. (2018). "Learning to navigate for fine-grained classification," in *European Conference on Computer Vision*, 4 (8), 420–435.

Yu, C., Zhao, X., Zheng, Q., Zhang, P., and You, X. (2018). "Hierarchical bilinear pooling for fine-grained visual recognition," in *European Conference on Computer Vision*, 2 (4), 574–589.

Zhang, N., Donahue, J., Girshick, R., and Darrell, T. (2014). "Part-based r-CNNs for fine-grained category detection," in *European Conference on Computer Vision*, 2 (3), 834–849.

Zhang, Z., Du, X., Jin, L., Wang, S., Wang, L., and Liu, X. (2022). Large-Scale underwater fish recognition *via* deep adversarial learning. *Knowledge Inf. Syst.* 4, 353–379. doi: 10.1007/s10115-021-01643-8

Zheng, H., Fu, J., Mei, T., and Luo, J. (2017). "Learning multi-attention convolutional neural network for fine-grained image recognition," in *IEEE International Conference on Computer Vision*, 3, 5209–5217.

Zheng, H., Fu, J., Zha, Z.-J., and Luo, J. (2019). "Looking for the devil in the details: learning trilinear attention sampling network for fine-grained image recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8, 5012–5021.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., et al. (2021). "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2, 6881–6890.

Zhou, J., Sun, J., Zhang, W., and Lin, Z. (2023a). Multi-view underwater image enhancement method *via* embedded fusion mechanism. *Eng. Appl. Artif. Intell.* 4, 1–12. doi: 10.1016/j.engappai.2023.105946

Zhou, J., Yang, T., Chu, W., and Zhang, W. (2022). Underwater image restoration *via* backscatter pixel prior and color compensation. *Eng. Appl. Artif. Intell.* 4, 1–16. doi: 10.1016/j.engappai.2022.104785

Zhou, J., Zhang, D., and Zhang, W. (2023b). Cross-view enhancement network for underwater images. *Eng. Appl. Artif. Intell.* 4, 1–11. doi: 10.1016/j.engappai.2023.105952

Zhuang, P., Wang, Y., and Qiao, Y. (2018). "WildFish: a large benchmark for fish recognition in the wild," in *ACM International Conference on Multimedia*, 8, 1301–1309. doi: 10.1016/j.engappai.2023.105952

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 01 May 2023
DOI 10.3389/fmars.2023.1151112

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Ercan Avşar,
Technical University of Denmark, Denmark
Leonardo Bobadilla,
Florida International University,
United States

*CORRESPONDENCE
Zhibin Yu
yuzhibin@ouc.edu.cn

RECEIVED 25 January 2023
ACCEPTED 11 April 2023
PUBLISHED 01 May 2023

CITATION
Han L, Zhai J, Yu Z and Zheng B (2023) See you somewhere in the ocean: few-shot domain adaptive underwater object detection.
Front. Mar. Sci. 10:1151112.
doi: 10.3389/fmars.2023.1151112

COPYRIGHT
© 2023 Han, Zhai, Yu and Zheng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

See you somewhere in the ocean: few-shot domain adaptive underwater object detection

Lu Han[1,2], JiPing Zhai[1,2], Zhibin Yu[1,2]* and Bing Zheng[1,2]

[1]Department of Electronic Engineering, College of Information Science and Engineering, Ocean University of China, Qingdao, Shandong, China, [2]Sanya Oceanographic Institution, Ocean University of China, Sanya, China

The current data-driven underwater object detection methods have significantly progressed. However, there are millions of marine creatures in the oceans, and collecting a corresponding database for each species for similar tasks (such as object detection)is expensive. Besides, marine environments are more complex than in-air cases. Water quality, illuminations, and seafloor topography may lead to domain shifting with visual instability features of underwater objects. To tackle these problems, we propose a few-shot adaptive object detection framework with a novel two-stage training approach and a lightweight feature correction module to accommodate both image-level and instance-level domain shifting on multiple datasets. Our method can be trained in a source domain and quickly adapt to an unfamiliar target domain with only a few labeled samples. Extensive experimental results have demonstrated the knowledge transfer capability of the proposed method in detecting two similar marine species. The code will be available at: https://github.com/roadhan/FSCW

KEYWORDS

computer vision, underwater object detection, domain adaptive, few shot, deep learning

1 Introduction

In recent years, with the development of deep learning technology and the deterioration of the marine ecological environment, underwater optical object detection has attracted more and more attention. However, many problems still need to be solved in underwater object detection. On the one hand, the underwater environment is complex and changeable. Affected by the scattering and absorption of the water medium, the quality of the images is usually poor (Fu et al., 2023). These underwater factors would inevitably involve inconsistent visual features. On the other hand, underwater images are challenging to collect and have limited reusability. Suppose we need more samples to boost a deep-learning model to handle a detection task. Generally, a common method is to use another large-scale dataset (e.g., Microsoft Common Objects in Context(MSCOCO) dataset Lin

et al. (2014)) to boost the model and finetune the model with limited samples with new categories (Cai et al., 2022). This method can be particularly helpful for new dataset tasks if the large-scale dataset contains similar target categories (Zhu et al., 2021a) (e.g., the experience of motorbike detection can help bicycle detection in another task). However, there are domain shifts between different datasets due to differences in shots, environments, and objects themselves (Li et al., 2022a; Yu et al., 2022). These domain shifts prevent us from fully exploiting prior knowledge on large datasets. Therefore, in-air adaptive object detection algorithms are designed to solve such problems. Different underwater optical characteristics can also easily cause domain shifts (Liu et al., 2020), resulting in hue changes and discrepancies in visual features. Moreover, due to changes in the ecological environment of the new waters, similar species may also have different appearance characteristics. Therefore, under these domain shifts, datasets collected in one water body are unlikely to help detection tasks in another water environment.

Similar to in-air domain adaptive object detection (Wang et al., 2019), we can divide underwater domain shift into image-level domain shift and instance domain shift. Image-level shift refers to the shift of the image in terms of style, brightness, etc. As shown in Figure 1, we attribute water transparency and chromatic aberration to image-level shift underwater. Instance-level shift refers to the shift of the target in appearance and size. We group organisms of the same family or genus but different species as instance-level shifts underwater. Any domain shift will have a significant performance degradation on the underwater detection network. The green bounding box represents the undetected target. Regarding results in Figure 1, the detection network can hardly work well under image-level and instance-level shifts.

Unsupervised domain-adaptive object detection based on deep learning is generally considered a solution to this kind of problem (Chen et al., 2018; Saito et al., 2019; Shen et al., 2019). However, the current domain-adaptive object detection algorithms have several apparent flaws. First, these methods always need a large amount of target domain data for training (Wang et al., 2019), which is difficult to obtain in underwater scenes. Second, due to the algal blooms or river floods at different times, the environmental conditions of the offshore and river outlets may change unexpectedly and cause a changeable aquatic background.

Although many existing few-shot object detection methods can work with a few data, their feature extraction ability on the new domain will be significantly affected by the changeable aquatic background. This is because most existing few-shot object detection considered shared weights or a separately trained feature extraction module to extract the feature map of the new class. Since the model has yet to see the new domain, the feature extraction ability on the new domain would be insufficient (Li et al., 2022d). On the other hand, most domain-adaptive methods can adapt to a new domain with sufficient retraining on the source domain and target domain data (Wang et al., 2019). However, such methods usually need a large amount of target domain data. The lengthy retraining time also hinders further applications on underwater vision.

Inspired by the theory of few-shot learning (Kang et al., 2019) and transfer learning (Sun et al., 2021), we propose a fast few-shot domain-adaptive algorithm to tackle the challenge of underwater cross-domain object detection. Our contributions can be summarized as follows: 1) Aiming at the problem of insufficient ability of the backbone to extract features, as shown in Figure 2, we fused the two-branch algorithm into a single-branch object detection algorithm with a channel-level feature correction module to solve this problem. 2) Many existing domain adaptation algorithms need a long time to adjust to a new domain. We propose a two-stage domain adaptation training strategy, which only takes a short time to adapt to the new target domain. 3) We conduct exhaustive experiments on two datasets, demonstrating that our algorithm performs excellently on few-shot domain adaptation problems. Compared to other domain adaptation algorithms, our algorithm has two key advantages:

1) **Boosting the model with limited data.** Compared with unsupervised domain adaptation (UDA) object detection, which requires many unlabeled samples, our model only needs a small number of labeled samples to complete the training and achieve excellent performance during the target domain adaptation.
2) **Adapting new tasks with less time.** When our model encounters unfamiliar environments, it no longer needs to be trained on both the source and target domain data simultaneously. Instead, it only needs to be fine-tuned on a small number of labeled target domain data sets, which reduces the adaptation time.

2 Related work

General Object Detection refers to finding the object we need from the image and giving an accurate mark frame and category (Li et al., 2022a). Current deep learning-based object detection can be divided into two architectures: one-stage and two-stage methods. The two-stage methods are mainly based on the region convolutional neural network (R-CNN) series. They use a convolutional neural network (CNN) to generate region proposals where objects may exist and perform further category prediction and bounding box regression in the detection head module. The one-stage methods perform end-to-end bounding box regression and category prediction through the neural network. The one-stage methods include You Only Look Once (YOLO) (Redmon et al., 2016; Zhu et al., 2021b), RetinaNet (Lin et al., 2017b), etc. Usually, two-stage methods outperform one-stage methods in accuracy, but they have poorer inference speeds. Both two architectures require large datasets for training. Considering the real-time requirements of underwater object detection, we use YOLOv5 (Zhu et al., 2021b) as the baseline in this paper

Underwater Object Detection is a particular branch of object detection. Compared with general object detection tasks, underwater images often have problems such as blurring, color

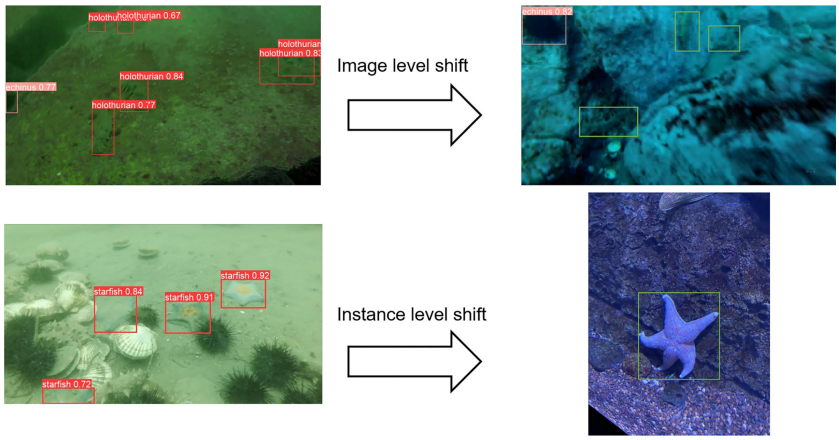


FIGURE 1
Two different underwater domain shifts and cross-domain performance degradation of detectors.

shifting, and costly data collection. To tackle these problems, (Lin et al., 2020) proposed an augmentation method called the region of interest mix-up (RoIMix), by fusing the proposed regions of different images to enhance the generalization of the detection network. (Fan et al., 2020a) proposed an underwater detection framework with feature enhancement and anchor refinement, which improves the ability of the detector to deal with underwater images of different scales. (Liang and Song, 2022) applying Self-Attention modules to the region of interest (RoI) features to improve underwater detector performance. However, the underwater objection detection methods often have to be deployed in an unseen underwater environment, which can lead to a domain shift. Unfortunately, the current underwater object detection algorithms have not yet considered the problem of adapting to different waters.

Domain Adaptation refers to reducing domain shift by training neural networks on source and target domain datasets. The current domain adaptive object detection is mainly based on unsupervised domain adaptation. According to the domain adaptation theory (Ganin et al., 2016), when performing neural network domain adaptation, the features extracted by the backbone must have domain invariant properties to adapt to a new domain. Ganin et al. 2016 used a gradient reversal layer with a domain classifier to constrain the backbone to extract features without domain shift to achieve this goal. This method is called domain adversarial training, which is still adopted by most domain adaptation methods. (Chen et al., 2018) divides domain shift into image-level and instance-level domain shift, and two adaptive components are designed to adapt to these two domain shifts, respectively. (Saito et al., 2019) designed a weak alignment model using adversarial alignment loss to address domain variance. (Kiran et al., 2022) proposes the domain transfer module (DTM) to transform the source image according to different target domain images, enabling the network to avoid catastrophic forgetting when performing multi-domain adaptation. (Li et al., 2022b) proposed a novel semantic conditional adaptation framework to reduce the cross-domain misclassification problem. The above works only focus on domain adaptation under large unsupervised samples and do not consider the problems encountered in few-shot domain adaptation. In the case of only a small number of samples, labeling samples do not add too much labor overhead. (Wang et al., 2019) considers the domain adaptation problem under the condition of small sample labeling. He proposed a two-layer module to adapt to the domain adaptive object detection problem under limited loose labeling. Loose labeling means that only part of each image is labeled, and more images are used to improve the target information of labeling. This method is promising for cases when image acquisition is easy but labeling is complex. Nevertheless, the reverse more or less applies in underwater object detection. Collecting underwater data is always expensive and time-consuming, but labeling objects is relatively easy. Unlike other domain adaptation methods, our model can quickly adapt to the target domain when there are only a few labeled samples in the target domain.

Few-shot learning refers to learning new categories with limited data. In the field of object detection, methods can be divided into two main branches: dual-branch methods and single-branch methods (Köhler et al., 2021). The dual-branch methods are shown in Figure 3A, and an auxiliary feature extraction module is used to extract the feature vector of the support set image. Support set vectors are then channel-level interacted with query set vectors. (Kang et al., 2019) use a pre-trained backbone on the basis of YOLO to extract the support set feature vector which will reweight the query set vector. (Fan et al., 2020b) on the basis of Faster-RCNN, use the shared weight backbone to extract the support set feature vector to complete the reweighting step and use the multi-relation detector to classify the target. (Lee et al., 2022) propose a method to refine the support information through an attention mechanism among support data before aggregating the query and support data. The single-branch methods are shown in Figure 3B. The single-branch methods are mainly based on transfer learning. (Wang et al., 2020) used the transfer learning theory to unfreeze the bounding box regression and the classification layer of Faster R-CNN achieves excellent performance. Sun et al. proposed a method (Sun et al., 2021) by controlling the form of intersection over the union (IoU) output with the Faster R-CNN of the unfreezing region proposal network (RPN) and region of interest (ROI) pooling layers and

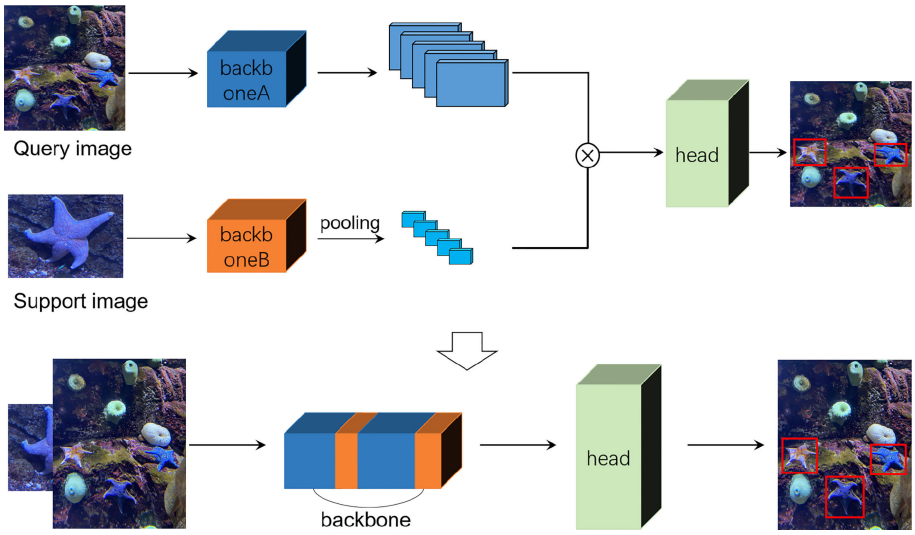


FIGURE 2
Improve the dual-branch structure to a single-branch structure that is more suitable for domain adaptation problems.

achieved the best performance at that time. Generally, the single-branch methods have only one backbone with fewer parameters and converge faster. Since the dual-branch methods considered meta-learning and more parameters, they can achieve better performance on few-shot learning. But their training speed is lower than single-branch cases. Since both two kinds of methods did not consider domain shifts, they will lead to a dramatic drop in performance when handling new samples from another domain. Our method combines the advantages of both approaches. To address this issue, we propose the feature correction module (FCM) (Figure 3C), which plays a similar role on the backbone B of dual-branch methods to enhance feature extraction ability with few samples. Furthermore, we use a two-stage fine-tuning method to make our model adjust itself to the features of the new domain.

3 Method

This section will briefly introduce our few-shot domain adaptive object detection algorithm. Due to the insufficient samples in the underwater target domain, the existing domain adaptation methods cannot achieve good results. The main reason for the poor performance in cross-domain object detection tasks is that the feature extraction ability of the backbone can hardly work in new domains (Li et al., 2022d). To solve this problem, we propose a solution. Firstly, we can overcome the overfitting problem of few-shot by introducing a two-stage training strategy. The proposed strategy can also reduce the need for repeated training on the source domain, shortening the time to adapt to the new domain. Secondly, by introducing a feature correction module, we

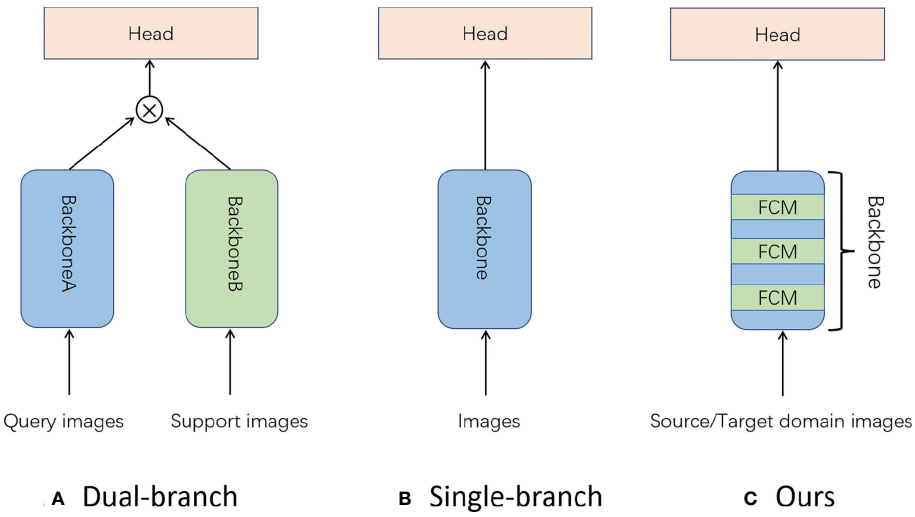


FIGURE 3
The comparison among two typical few-shot learning structures and ours.

further enhance the feature extraction ability of the backbone on new domains. Since the feature correction module only contains quite a few trainable parameters, it only takes a little for training. When only a few labeled samples are in the target domain, our method can quickly adapt to the target domain and achieve excellent performance.

3.1 Problem definition

We follow and extend the definition of "n-shot learning" given by (Kang et al., 2019). Suppose we have k images with labels in the source domain. We can define these images and labels in the source domain as $D_s = \{(X_{s1}, Y_{s1}), \ldots, (X_{sk}, Y_{sk})\}$. Similarly, we can define the images and labels in the target domain as $D_t = \{(X_{t1}, Y_{t1}), \ldots, (X_{tm}, Y_{tm})\}$. Since the target domain data is often less than the source domain, we have $k >> m$. Here D_s and D_t represent the source domain and target domain data, respectively. X and Y represent the images and the corresponding target labels. Let $num()$ denote the number of instances in a domain. In Kang et al.'s work (Kang et al., 2019), they defined n-shot ($num(X) = n$) as available samples (instances) in a domain. In the case of instance-level domain shift, the shape of the target will change significantly with the region. Thus, we followed this definition to evaluate instance-level domain shifts as $n - shot_{instance} = num(X_{t1}) + \ldots + num(X_{tm})$, where the $num()$ is the number of instances of an image. Since the main factor to cause image-level domain shifts is the environment (not the objects), we further define $n - shot_{image} = num'(X_t))$, where the $num'()$ means the number of images.

3.2 Two-stage fine-tuning method

Most existing few-shot learning approaches consider only adjusting the classification and the bounding Box regression header without changing the parameters of the backbone (Wang et al., 2020). Such an operation can correct new few-shot samples in a short time. However, the underwater domain shifts will also affect the backbone rather than the header. Inspired by Li et al.'s work (Li et al., 2022d), which proposed a two-stage fine-tuning strategy to correct a cross-domain classification task, we further extend the fine-tuning method to solve a cross-domain object detection problem.

Since different layers of the backbone network can extract different scales of features (Lin et al., 2017a), we focus on the domain correction of the backbone. Furthermore, to reduce the fine-tuning cost and accelerate the re-training speed, we insert some feature extraction modules (FCMs, please refer to Section 3.3 for detail) into the backbone and only update these feature extraction modules in the fine-tuning phase. As a result, our two-stage fine-tuning strategy can reduce the number of trainable parameters to solve the overfitting problem of few-shot. Our two-stage training method reduces the number of parameters by 42% compared to direct training YOLOv5, while our newly added FCM only increases the number of parameters by 0.00278%. The training method is shown in Figure 4. To overcome the underwater cross-domain challenge, our method includes two stages:

Base training: Our first stage is only performed on the source domain training dataset. In order to ensure a fair comparison, except for the modification of the network module we will mention in Section 3.3, the training hyperparameters remain the same as those of YOLOv5. We did not perform any hyperparameter tuning. The joint loss function is:

$$L_{total} = \lambda_1 L_{cls} + \lambda_2 L_{obj} + \lambda_3 L_{box} \tag{1}$$

where λ_1, λ_2 and λ_3 is the custom hyperparameter. Among them, both L_{cls} and L_{obj} using binary cross entropy (BCE) loss for classification and foreground detection, respectively:

$$L_{BCE} = -\frac{w}{N}\sum_{n=1}^{N}[y_n \cdot \log F(x)_{x\sim P_s(x)} + (1 - y_n) \cdot \log(1 - F(x)_{x\sim Ps(x)})] \tag{2}$$

where w is a hyperparameter, x and y represents different images and labels. $P_{s/t}$ represents our network to obtain data from the source or target domain at different training stages. represents the number of samples. F represents the model. L_{box} uses CIoU loss (Zheng et al., 2020):

$$L_{CIoU} = IoU - \left(\frac{\rho^2(b^s, b^s_{gt})}{c^2} + \alpha v\right) \tag{3}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\frac{w}{h}\right)^2 \tag{4}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{5}$$

where ρ represents the Euclidean distance between b^s and b^s_{gt}, $b^{s/t}$ and $b^{s/t}_{gt}$ represents the detected bounding box and ground truth on the

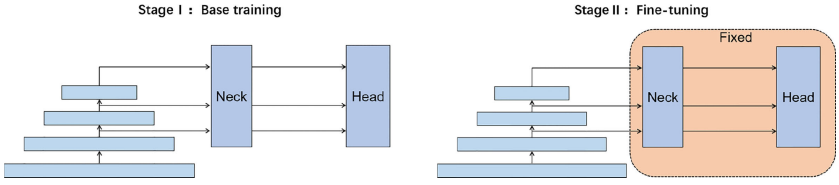


FIGURE 4
Our two-stage training approach.

source domain dataset or target domain dataset. *IoU* represents the intersection over the union.

Fine-tuning: Our second stage (fine-tuning) is performed on a small amount of labeled target domain data. In this stage, we freeze the neck and head modules of the detection network and only perform gradient updates on the backbone, the function is:

$$\frac{\partial L_{total}}{\partial net_b^t} = \frac{\partial L_{total}}{\partial net_h^t} \cdot \frac{\partial net_h^t}{\partial net_b^t} \tag{6}$$

$$W(net_h^t) \equiv W(net_h^s) \tag{7}$$

Among them, $net_h^{t/s}$ represents the neck and head network modules on the target domain dataset or source domain dataset, and net_b^t represents the backbone module on the target domain dataset. W represents the network weight.

Since the target domain dataset is adopted in the fine-tuning stage, our BCE loss and CIoU loss function are changed accordingly to:

$$L_{BCE} = -\frac{w}{N}\sum_{n=1}^{N}[y_n \cdot \log F(x)_{x\sim P_t(x)} + (1-y_n)\cdot \log(1 - F(x)_{x\sim P_t(x)})] \tag{8}$$

$$L_{CIoU'} = IoU - \left(\frac{\rho^2(b^t, b_{gt}^t)}{c^2} + \alpha v\right) \tag{9}$$

3.3 Lightweight feature correction module

In the field of few-shot learning, feature reweighting for dual-branch object detection is a popular solution (Köhler et al., 2021). In dual-branch few-shot object detection, the channel reweighting of the support set vector to the query set vector plays a key role in few-shot learning. Following this idea, we aim to build a reweighting module in our single backbone to help our model quickly adapt to new samples. However, since the backbone has yet to see this new category, it cannot accurately extract information. Therefore, we design a channel-level feature rectification module that can replace the feature interaction stage in two-stage few-shot training. We insert it into the backbone so the backbone can perform channel correction on the generated feature vector according to the image domain information in the new domain during the training process.

In the backbone network, a common view is that we can extract the different scales of features from different layers (Lin et al., 2017a; Li et al., 2022d). Inspired by this point, we uniformly insert the FCM into the backbone network to address the instance-level and image-level domain shifts.

The Feature Correction Module (FCM) we designed is shown in the lower part of Figure 5, and then we insert it into CSPDarknet53, which is the backbone of YOLOv5, as shown in the upper part of Figure 5. In each FCM, there are two branches. The first branch saves the raw input feature maps, and the second generates a reweighting vector to correct the feature maps of the first branch. Suppose the input feature map size is $h*w*c$. In the second branch, the feature map will first go through a global avg-pooling operation to obtain a $1*1*c$ vector followed by c groups depthwise convolution. Next, we feed the output $1*1*c$ vector to a sigmoid activation layer to normalize and reweight vectors. At last, the reweighted vectors will multiply the feature maps of the first branch to obtain the final outputs

4 Experiment

In this section, we will introduce the experimental results of our method and other methods in different scenarios. The experimental results are represented by the mean average precision (mAP) with an IOU threshold of 0.5. The mAP is determined by Precision and Recall. Precision represents the accuracy of the detected samples, and Recall represents the proportion of correctly detected samples among all correct samples.

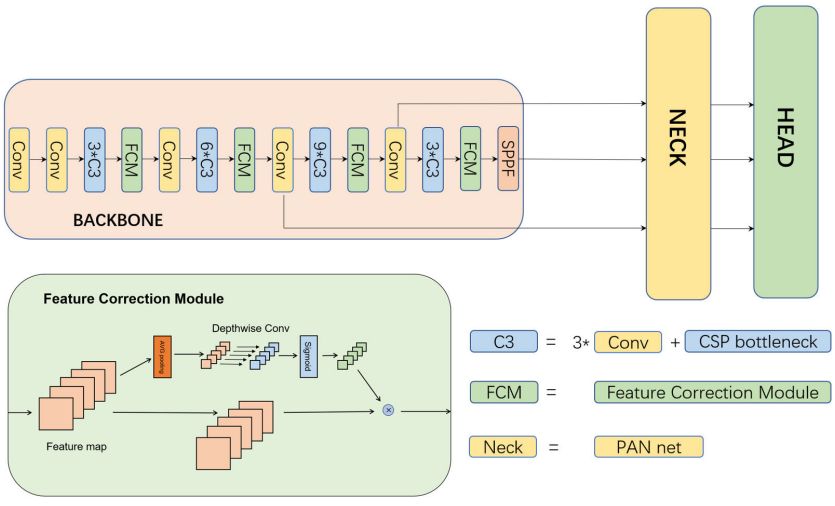


FIGURE 5
Our model structure and the specific implementation of the feature correction module.

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

Among them, TP refers to the true positive, which means the detected real samples; FN refers to the false negative, which means the correct samples that were not detected; and FP refers to the false positive, which means the falsely detected samples. The AP is determined by the area under the Precision-Recall (PR) curve, and we use the interpolation method to calculate it:

$$AP = \sum_{i=1}^{n-1} (Recall_{i+1} - Recall_i) \cdot Precision_{interp}(Recall_{i+1}) \tag{12}$$

Here mAP refers to the average value of each type of AP:

$$mAP = \sum_{i=1}^{k} AP_i \tag{13}$$

The mAP50 used in the following experiments means that the mAP score with an IOU threshold of 0.5. The adaptation time refers to the time required for each method to achieve the optimal effect in the target domain, and the time unit is hours (h).

4.1 Datasets

S-UODAC2020: This dataset was processed by Song et al. (Song et al., 2021). They used the style transfer model WCT2 (Yoo et al., 2019) to process the original UODAC2020 dataset into seven common underwater domains for evaluating domain adaptation, and each domain type has 791 images. type1-type6 is the source domain, and type7 is the target domain.

URPC2022[1]: URPC contains 9,000 images. The original dataset contains four categories, such as starfish. Here we only take the starfish category for analysis.

Aquarium[2]: The dataset consists of 638 images collected from two aquariums in the United States, which also contain the starfish class. Since the paired categories in the two data sets only include the starfish category, we use the starfish class from these two datasets (URPC and Aquarium) for cross-domain testing, in which URPC2022 is the source domain and Aquarium is the target domain.

4.2 Implementation details

Our code is based on official YOLOv5x(PyTorch)[3] with COCO dataset pre-training weights. Except for our proposed FCM, we do not adopt any other modules to modify the network. We adopt the Stochastic Gradient Descent (SGD) optimizer with a 0.01 learning rate and a 16 batch size. We set the picture size to 640 on the long side. All training time statistics are performed with a graphic card of GTX1080ti (11G).

4.3 Benchmark comparison

In Table 1, we compared two UDA methods including SCL (Shen et al., 2019) and SCAN (Li et al., 2022b) on the S-UODAC2020 dataset. The four columns (holothurian, echinus, scallop and starfish) in Table 1 represent the AP50 values of each category in the dataset, and mAP50 represents the average value of all categories. The time column represents the adaptation time of the algorithm when encountering a new domain, and the unit is hours. For the baseline, we used the network freeze strategy (freeze backbone) recommended by YOLOv5 to solve the few-shot problem (YOLOv5 w/ft). Since the dataset mainly includes image-level domain shifts, the number of targets in each picture is large, we adopt *shot* = *num* (X_t), and the performance results under ten shots are shown in Table 1. We can find that the UDA methods have poor accuracy under 10-shot. The two UDA methods also take a long time to adapt to each domain. Our method overcomes this problem with the only additional cost of labeling a few samples, which does not consume too much human effort.

In Table 2, we also compared methods such as SCL, SCAN, and SIGMA (Li et al., 2022c) on the URPC2022 and Aquarium dataset settings. The number of targets in the images of these datasets is relatively balanced, and there are image-level and instance-level domain offsets at the same time, so we strictly use the method to count. We provide the results under 3-shot and 10-shot in Table 2. The experimental results of "YOLOv5 w/ft" shown in Table 2 freezing the backbone module and fine-tuning the header module are better than "YOLOv5 w/o ft" but worse than our results. That means freezing the backbone module and fine-tuning the header module (YOLOv5 w/ft) can correct the domain shift to a certain extent, but the efficiency is lower than ours (freezing the head module and updating the backbone).

It can be seen that the classic UDA methods (SCL, SCAN, and SIGMA) cannot work with a small number of samples, and their time to adapt to the unfamiliar domain is much longer than our method, so they cannot quickly adapt to the unfamiliar domain.

4.4 Ablation analysis

To validate each component of our method, we design an ablation study on the S-UODAC dataset, as shown in Table 3. The "bb-ft" represents our migration learning strategy, and the "FCM" denotes the feature correction module. The four columns before the mAP column in Table 3 represent the AP50 values of each category in the dataset, and mAP50 represents the average value of all categories. Both the feature correction module and the migration learning strategy can significantly improve the

1 http://www.urpc.org.cn/

2 https://universe.roboflow.com/data-science-day-dry-run/aquarium-6cfzm/dataset/1.

3 https://github.com/ultralytics/yolov5

TABLE 1 Our comparison results with other methods on the S-UODAC dataset.

method	holothurian	echinus	scallop	starfish	mAP50	Time
SCL	0.491	0.725	0.589	0.345	0.546	13.2h
SCAN	0.399	0.745	0.469	0.252	0.466	6.9h
YOLOv5 w/ft	0.604	0.780	0.707	0.587	0.669	0.19h
Ours	0.613	0.804	0.722	0.685	0.706	0.19h

TABLE 2 Our comparison results with other methods on the URPC2022 and Aquarium dataset.

method	3-shot		10-shot	
	mAP50	Time	mAP50	Time
SCL	0.349	14.3h	0.478	15.6h
SCAN	0.545	5.1h	0.607	6.2h
SIGMA	0.636	6.6h	0.652	6.5h
YOLOv5 w/o ft	0.516	–	0.516	–
YOLOv5 w/ft	0.685	0.1h	0.714	0.14h
Ours	0.710	0.09h	0.736	0.11h

TABLE 3 Our ablation experiments on the S-UODAC dataset.

method	bb-ft	FCM	holothurian	echinus	scallop	starfish	mAP50
Benchmark			0.425	0.803	0.647	0.519	0.599
Ours	✓		0.621	0.783	0.703	0.617	0.681
		✓	0.580	0.798	0.725	0.608	0.678
	✓	✓	0.613	0.804	0.722	0.685	0.706

✓ represents the training method using the column.

performance of the baseline model. We achieve the best result when these two components work simultaneously.

In Table 4, we tested with the activation functions in FCM and found that the sigmoid function performs slightly better than rectified linear unit (ReLU) in performance. For the three challenging categories, the sigmoid function leads to significant improvements. We conclude that this is because the sigmoid normalizes the vector between 0 and 1, which helps the final reweighting of our feature correction module. We also found that the FCM module with the sigmoid function converges faster than the case with the ReLU function. The result also verifies the point of attention mechanisms in recent years (Vaswani et al., 2017).

We visualize the results of the ablation experiments. The green bounding box in the figure refers to the correct sample missed by the detector. Figure 6A results from the benchmark training only on the source domain. The model missed many instances when we performed a cross-domain test. The results in Figure 6A also show the shortcomings of current detectors in cross-domain detection performance. Figure 6B shows the two-stage training method's result. We can see that the fine-tuning process can significantly reduce the number of missed samples, but some samples are still undetected. Figure 6C is the result of using the two-stage training method and FCM at the same time. It can be seen that our method has only one missed target and no false detections. Based on the result in Figure 6, we can conclude that both the proposed two-stage training method and FCM can efficiently resist the performance degradation from the cross-domain detection task.

To further verify the attention improvement, Figure 7 shows some examples using the Gradient-weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al., 2017) image under different datasets. Gradient-weighted Class Activation Mapping can reflect which part of the image the neural network pays

TABLE 4 Performance of different activation functions on the S-UODAC dataset.

activation	holothurian	echinus	scallop	starfish	mAP50
ReLU	0.596	0.814	0.697	0.683	0.698
Sigmoid	0.613	0.804	0.722	0.685	0.706

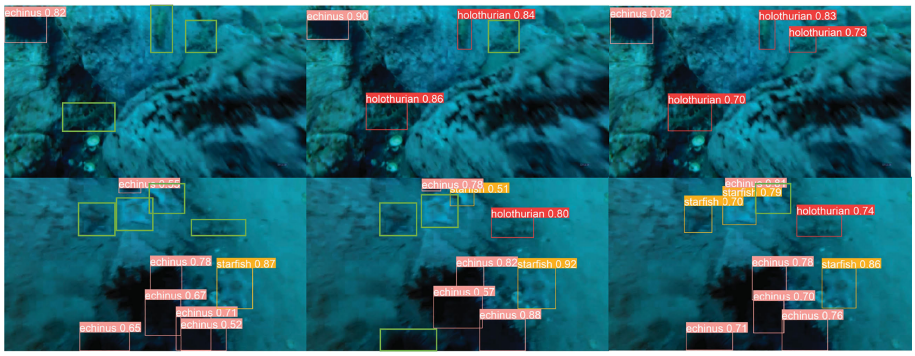


FIGURE 6
Visualization results of ablation experiments.

attention to when detecting and recognizing a certain type of object. The redder the color of the heat map, the more the network pays attention to this part. Figure 7A contains three raw images; Figure 7B shows the results of YOLOv5 trained only on the source domain. We can see that many target areas are inactivated during the detection process. In other words, the network has not paid attention to these areas. Figure 7C represents the Grad-CAM results of our method. All target regions are accurately activated after fine tuning with our approach. The heat map visualization results indicate that our method can better locate the object in the new domain. The heat map visualization results can also prove the above point of view. Figure 7B (freezing the backbone module and fine-tuning the header module) performs worse than Figure 7C (freezing the header module and fine-tuning the backbone module). Our network paid attention to these targets without missing the original detected samples, indicating that the extracted features are offset from the actual feature space when the backbone is not adapted to the target domain.

When we select the final weight, we adopt the "early stop" strategy, which allows us to obtain the training weight when the loss of the verification set is the smallest. Figure 8 is the loss curve image during our fine-tuning process. In the "early stop" strategy, a commonly used parameter is "patience". Assuming its value is n, it means that if the result of the k_{th} epoch training is still the best

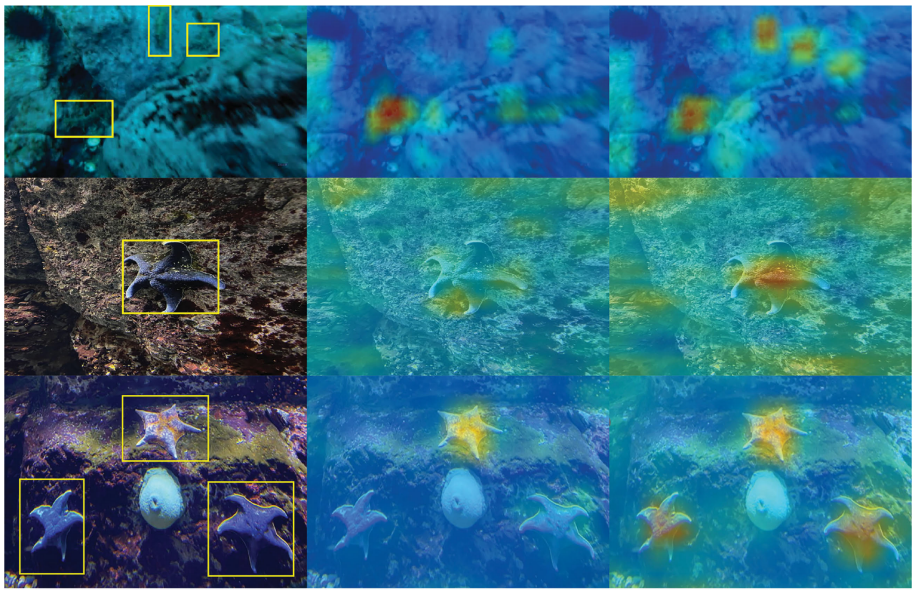


FIGURE 7
Our Grad-CAM images under different datasets.

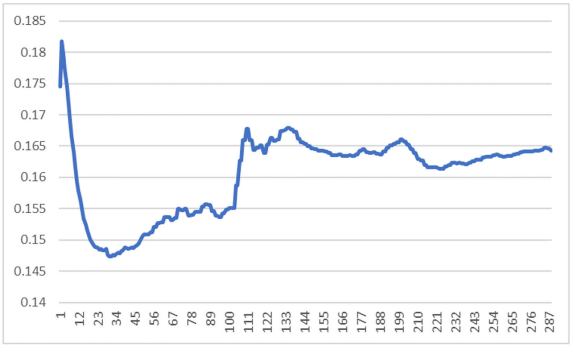


FIGURE 8
Our loss curve chart.

after n epochs, stop the training. The weight of the k_{th} epoch is selected as the final weight. We set n to 250. From Figure 8, we can find that our method quickly converged in about 30_th epoch. Then the curve gradually grew up. Since the lowest point is clear, a large enough can easily locate the lowest point.

We also test our model in a short underwater video to prove the superiority of our method for object detection in unfamiliar waters. Figure 9 shows the detection result of one frame. The left side is our method, and the right is the fine-tuning results after pre-training on a large-scale dataset (COCO) of YOLOv5. Our approach is significantly ahead of the comparison method in both recall and precision, and our fine-tuning uses the first frame of the video. More details can be found in our GitHub project.

5 Discussion

Currently, deep learning has achieved remarkable results in computer vision and has also produced good results in underwater computer vision, such as underwater observation and underwater image processing. However, its data-driven models also have limitations. As discussed in the article, deep learning models have shown a significant performance drop in test scenarios in an unfamiliar environment with different data distributions from the training set. Previous works Chen et al. (2018); Ganin et al. (2016) have shown that the main reason for cross-domain performance degradation in tasks such as classification and object detection is that the backbone cannot extract domain-invariant features.

In the field of underwater vision, we have an urgent need for domain adaptation algorithms:

- Underwater images are affected by plankton and river flooding disasters, often resulting in large changes in image colors.
- In different water domains, due to environmental influences, biological morphology often has certain changes.
- Many different species of the same family have certain differences in appearance, which also brings about domain shifts.

Regardless of the data domain in which the target category appears, humans can accurately capture the invariant features in different domains to complete classification and labeling. Inspired by this point, many researchers trained the backbone through domain adversarial training and other strategies, which can make the backbone extract domain invariant features. However, this training method requires a large number of target domain samples, which is very difficult to obtain in underwater scenarios. Unfortunately, we often need more training samples to adapt to the test scenario, especially when underwater data collection is challenging.

We propose a few-shot domain adaptation object detection algorithm based on a two-stage training strategy and an FCM module, which can quickly adapt to the target domain with only a small number of annotated samples, not only solving the defects of previous domain adaptation work under few-shot but also being more suitable for underwater scene applications. However, our method still has some drawbacks. When the algorithm adapts to the target domain, it does not consider catastrophic forgetting. Because we only use target domain samples to fine-tune the network rather than jointly training with source domain samples, this inevitably leads to a performance drop in the source domain.

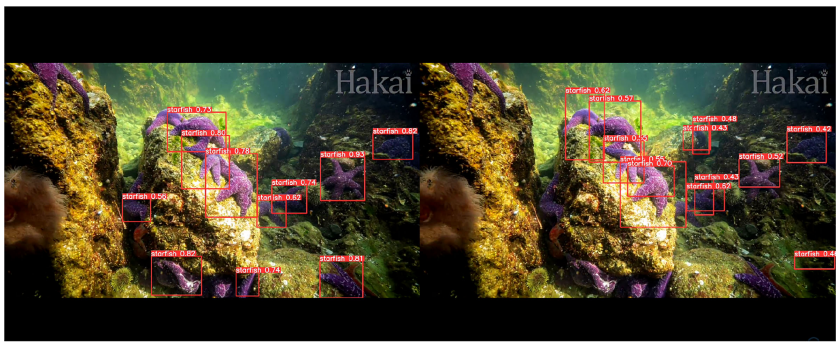


FIGURE 9
Demo on a YouTube video, the confidence threshold is 0.4 and the IOU threshold is 0.45.

Our current solution to this problem is to retain weight files for each domain so that they can be used at any time.

6 Conclusion

This paper proposes a novel few-shot domain adaptive object detection framework. Our algorithm can transfer the object knowledge information from the source domain to the target domain, achieving a situation where only a small number of annotated target domain samples are used. At the same time, our algorithm also inspires unsupervised few-shot domain adaptive object detection, such as exploring the use of an image-to-image translation model to generate a small number of target domain samples for training using our method.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://github.com/roadhan/FSCW.

Author contributions

LH completed most of the work in this paper. JZ completed the synthesis of the datasets and the typesetting of the paper. ZY handled the work of revising the article, and BZ provided guidance and funding for this research. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 62171419), the Project of Sanya Yazhou Bay Science and Technology City (Grant No. SCKJ-JYRC-2022-102) and Hainan Province Science and Technology Special Fund, China (ZDYF2022SHFZ318).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Cai, L., Zhang, Z., Zhu, Y., Zhang, L., Li, M., and Xue, X. (2022). "BigDetection: a large-scale benchmark for improved object detector pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 4777–4787.

Chen, Y., Li, W., Sakaridis, C., Dai, D., and Van Gool, L. (2018). "Domain adaptive faster r-CNN for object detection in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3339–3348.

Fan, B., Chen, W., Cong, Y., and Tian, J. (2020a). "Dual refinement underwater object detection network," in *Proceedings of the European Conference on Computer Vision*. 275–291.

Fan, Q., Zhuo, W., Tang, C.-K., and Tai, Y.-W. (2020b). "Few-shot object detection with attention-RPN and multi-relation detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4013–4022.

Fu, C., Liu, R., Fan, X., Chen, P., Fu, H., Yuan, W., et al. (2023). Rethinking general underwater object detection: datasets, challenges, and solutions. *Neurocomputing* 517, 243–256. doi: 10.1016/j.neucom.2022.10.039

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* 17, 2096–2030.doi: 10.48550/arXiv.1505.07818

Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J., and Darrell, T. (2019). "Few-shot object detection *via* feature reweighting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8420–8429.

Kiran, M., Pedersoli, M., Dolz, J., Blais-Morin, L.-A., Granger, E., et al. (2022). Incremental multi-target domain adaptation for object detection with efficient domain transfer. *Pattern Recognition* 129, 108771. doi: 10.1016/j.patcog.2022.108771

Köhler, M., Eisenbach, M., and Gross, H.-M. (2021). Few-shot object detection: a comprehensive survey. *arXiv preprint arXiv* 2112, 11699. doi: 10.1109/TNNLS.2023.3265051

Lee, H., Lee, M., and Kwak, N. (2022). "Few-shot object detection by attending to per-sample-prototype," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2445–2454.

Li, W.-H., Liu, X., and Bilen, H. (2022d). "Cross-domain few-shot learning with task-specific adapters," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7161–7170.

Li, W., Liu, X., Yao, X., and Yuan, Y. (2022b). "SCAN: cross domain object detection with semantic conditioned adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 6. 7.

Li, W., Liu, X., and Yuan, Y. (2022c). "SIGMA: semantic-complete graph matching for domain adaptive object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5291–5300.

Li, B., Wang, C., Reddy, P., Kim, S., and Scherer, S. (2022a). "AirDet: few-shot detection without fine-tuning for autonomous exploration," in *Proceedings of the European Conference on Computer Vision*. 427–444.

Liang, X., and Song, P. (2022). "Excavating roi attention for underwater object detection," in *Proceedings of the IEEE International Conference on Image Processing*. 2651–2655.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2117–2125.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*. 2980–2988.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "Microsoft COCO: common objects in context," in *Proceedings of the European Conference on Computer Vision*. 740–755.

Lin, W.-H., Zhong, J.-X., Liu, S., Li, T., and Li, G. (2020). "ROIMIX: proposal-fusion among multiple images for underwater object detection," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*. 2588–2592.

Liu, H., Song, P., and Ding, R. (2020). WQT and DG-YOLO: towards domain generalization in underwater object detection. *arXiv preprint arXiv* 2004, 06333. doi: 10.48550/arXiv.2004.06333

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 779–788.

Saito, K., Ushiku, Y., Harada, T., and Saenko, K. (2019). "Strong-weak distribution alignment for adaptive object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6956–6965.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). "Grad-CAM: visual explanations from deep networks *via* gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*. 618–626.

Shen, Z., Maheshwari, H., Yao, W., and Savvides, M. (2019). SCL: towards accurate domain adaptive object detection *via* gradient detach based stacked complementary losses. *arXiv preprint arXiv* 1911, 02559. doi: 10.48550/arXiv.1911.02559

Song, P., Dai, L., Yuan, P., Liu, H., and Ding, R. (2021). Achieving domain generalization in underwater object detection by image stylization and domain mixup. *arXiv preprint arXiv* 2104, 02230.

Sun, B., Li, B., Cai, S., Yuan, Y., and Zhang, C. (2021). "FSCE: few-shot object detection *via* contrastive proposal encoding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7352–7362.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst.* 30:6000–6010. doi: 10.5555/3295222.329534

Wang, X., Huang, T. E., Darrell, T., Gonzalez, J. E., and Yu, F. (2020). Frustratingly simple few-shot object detection. *arXiv preprint arXiv* 2003, 06957. doi: 10.48550/arXiv.2003.06957

Wang, T., Zhang, X., Yuan, L., and Feng, J. (2019). "Few-shot adaptive faster r-CNN," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7173–7182.

Yoo, J., Uh, Y., Chun, S., Kang, B., and Ha, J.-W. (2019). "Photorealistic style transfer *via* wavelet transforms," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9036–9045.

Yu, F., Wang, D., Chen, Y., Karianakis, N., Shen, T., Yu, P., et al. (2022). "SC-UDA: style and content gaps aware unsupervised domain adaptation for object detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 382–391.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D. (2020). "Distance-IoU loss: faster and better learning for bounding box regression," in *Proceedings of the AAAI Conference on Artificial Intelligence*. 12993–13000.

Zhu, C., Chen, F., Ahmed, U., Shen, Z., and Savvides, M. (2021a). "Semantic relation reasoning for shot-stable few-shot object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8782–8791.

Zhu, X., Lyu, S., Wang, X., and Zhao, Q. (2021b). "TPH-YOLOv5: improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2778–2788.

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 01 May 2023
DOI 10.3389/fmars.2023.1113669

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Yangfan Wang,
Ocean University of China, China
Xiaoling Zhang,
University of Electronic Science and Technology of China, China
Ibrar Ahmad,
University of Peshawar, Pakistan

*CORRESPONDENCE
Lili Zhan
skd992016@sdust.edu.cn

RECEIVED 01 December 2022
ACCEPTED 07 April 2023
PUBLISHED 01 May 2023

CITATION
Yasir M, Zhan L, Liu S, Wan J, Hossain MS, Isiacik Colak AT, Liu M, Islam QU, Raza Mehdi S and Yang Q (2023) Instance segmentation ship detection based on improved Yolov7 using complex background SAR images.
Front. Mar. Sci. 10:1113669.
doi: 10.3389/fmars.2023.1113669

COPYRIGHT
© 2023 Yasir, Zhan, Liu, Wan, Hossain, Isiacik Colak, Liu, Islam, Raza Mehdi and Yang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Instance segmentation ship detection based on improved Yolov7 using complex background SAR images

Muhammad Yasir[1], Lili Zhan[2]*, Shanwei Liu[1], Jianhua Wan[1], Md Sakaouth Hossain[3], Arife Tugsan Isiacik Colak[4], Mengge Liu[2], Qamar Ul Islam[5], Syed Raza Mehdi[6] and Qian Yang[7]

[1]College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao, China, [2]College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China, [3]Department of Geological Sciences, Jahangirnagar University, Dhaka, Bangladesh, [4]National University International Maritime College Oman, Sahar, Oman, [5]Department of Electrical and Computer Engineering, College of Engineering, Dhofar University, Salalah, Oman, [6]Department of Marine Engineering, Ocean College, Zhejiang University, Zhoushan, Zhejiang, China, [7]People's Liberation Army (PLA) Troops No.63629, Beijing, China

It is significant for port ship scheduling and traffic management to be able to obtain more precise location and shape information from ship instance segmentation in SAR pictures. Instance segmentation is more challenging than object identification and semantic segmentation in high-resolution RS images. Predicting class labels and pixel-wise instance masks is the goal of this technique, which is used to locate instances in images. Despite this, there are now just a few methods available for instance segmentation in high-resolution RS data, where a remote-sensing image's complex background makes the task more difficult. This research proposes a unique method for YOLOv7 to improve HR-RS image segmentation one-stage detection. First, we redesigned the structure of the one-stage fast detection network to adapt to the task of ship target segmentation and effectively improve the efficiency of instance segmentation. Secondly, we improve the backbone network structure by adding two feature optimization modules, so that the network can learn more features and have stronger robustness. In addition, we further modify the network feature fusion structure, improve the module acceptance domain to increase the prediction ability of multi-scale targets, and effectively reduce the amount of model calculation. Finally, we carried out extensive validation experiments on the sample segmentation datasets HRSID and SSDD. The experimental comparisons and analyses on the HRSID and SSDD datasets show that our model enhances the predicted instance mask accuracy, enhancing the instance segmentation efficiency of HR-RS images, and encouraging further enhancements in the projected instance mask accuracy. The suggested model is a more precise and efficient segmentation in HR-RS imaging as compared to existing approaches.

KEYWORDS

computer vision, object detection, instance segmentation, HR-RS, YOLOv7, SSDD, HRSID, SAR Complex background images

1 Introduction

SAR is a microwave imaging sensor built on electromagnetic wave scattering properties that may be used in all weather conditions and has some ability to penetrate clouds and the ground. With the ongoing exploitation of maritime resources as well as the increased attention being paid to the monitoring of marine ships, it has special benefits in marine monitoring, mapping, the military, and all of these fields(Li et al., 2022; Liu et al., 2022; Kong et al., 2023; Yasir et al., 2023a; Yasir et al., 2023b). SAR ship detection technique is therefore very important for protecting marine ecosystems, maritime law enforcement, and territorial sea security. Ocean ship monitoring has received a lot of attention (Zhang et al., 2020b; Chen et al., 2021; Xu et al., 2022a; Zhang et al., 2023). Synthetic aperture radar (SAR) is more suited for monitoring ocean ships than optical sensors (Zeng et al., 2021; Zhang and Zhang, 2021a; Xu et al., 2022b; Zhang and Zhang, 2022c) because of its ability to operate in all weather conditions (Zhang and Zhang, 2021b). Ship monitoring is a key maritime task that is crucial for ocean surveillance, national defense security, fisheries management, etc. identification Ship in the SAR picture is a significant area of remote sensing research because it relies on target detection technology, which is in high demand (Wang et al., 2018; Chang et al., 2019; Qian et al., 2020; Su et al., 2022). Ship identification in satellite RS pictures has grown in importance as a research area recently (Nie et al., 2020). The marine transportation sector is now developing extremely quickly. The number of maritime infractions has increased as a result of the quick expansion in ship numbers and shipping volume. Automated ship identification plays an increasingly essential role in maritime surveillance, monitoring, and traffic supervision as well as in the regulation of illegal fishing and freight transit. It can assist in gathering information about ship dispersion. HR-RS pictures are given by a variety of airborne and spaceborne sensors, including Gaofen-3, TerraSAR-X, RADARSAT-2, Ziyuan-3, Sentinel-1, Gaofen-2, and unmanned aerial vehicles (UAV), owing to the quick development of imaging technology in the domain of RS. These HR pictures are being used in the military and the domains of the national economy, such as traffic control, marine management, urban monitoring, and ocean surveillance (Mou and Zhu, 2018; Cui et al., 2019; Su et al., 2019; Sun et al., 2021b). The HR RS pictures are especially well suited for object identification and segmentation in areas like military precision strikes and maritime transportation safety (Su et al., 2019; Wang et al., 2019; Zhang et al., 2020a). Instance segmentation, which may be characterized as a technology that addresses both the issue of object identification and semantic segmentation, has emerged as a significant, sophisticated, and challenging area of research in machine vision. Parallel to semantic segmentation, it has both pixel-level classification and object identification properties, where dissimilar instances must be located even if they belong to the same type (Xu et al., 2021). Since the two-stage object identification algorithm's introduction, other convolutional neural network-based object detection and segmentation methods have appeared, including the R-CNN, Faster R-CNN (Ren et al., 2015), and Mask R-CNN (He et al., 2017).

Deep learning innovation demonstrates inspiring outcomes recently in several fields, including object identification (Zhang et al., 2019a; Zhang et al., 2020c; Zhang et al., 2021a), image classification (Liu et al., 2021b; Zhou et al., 2022a; Zhou et al., 2022b), Segmentation (Liu et al., 2021b; Zhou et al., 2021; Zong and Wan, 2022; Zong and Wang, 2022), and so on (Zhou et al., 2019; Liu et al., 2021a; Wu et al., 2022; Yin et al., 2022; Zhu and Zhao, 2022). Recently, despite the existence of many excellent algorithms, like the path aggregation network (Liu et al., 2018), Mask Score R-CNN (Wang et al., 2020a), Cascade Mask R-CNN (Dai et al., 2016), and segmenting objects by locations (Wang et al., 2020b) and so on (Zhang and Zhang, 2019; Zhang et al., 2019b; Zhang et al., 2021b; Shao et al., 2022; Zhang and Zhang, 2022a; Zhang and Zhang, 2022b; Zhang and Zhang, 2022c; Zhang and Zhang, 2022d), common issues, such as erroneous segmentation edges and the development of global relations, still exist. The extension of the model will lead to dimensional disasters if the long-range dependencies are represented by dilated convolution or by expanding the number of channels. YOLOv7 serves as the basic foundational framework for the development of a framework model for RS picture object identification and instance segmentation in order to get over CNNs' limitations in terms of their capacity to extract spatial information. Detecting and segmenting ships in SAR images is difficult because of the complexity and variety of the images themselves, which include speckle noise, shadows, and cluttered backgrounds. These elements make it challenging to reliably identify ships among other objects in the image and to define the ship's boundaries.

In addition, different from moving targets such as aircraft and vehicles, ship targets often dock side by side near the port, so it is difficult for general detection methods to accurately distinguish each target, resulting in a large number of missing targets. Meanwhile, Ship case segmentation can not only accurately obtain the position of the object, but also effectively achieve the shape information of the target, which can further promote the research of SAR ship recognition. However, at present, a large number of studies only focus on the SAR ship targets detection and do not further achieve the target-level instance segmentation. It is specifically affected by the following factors, (1) the complexity of the instance segmentation model is high, often reaching hundreds of megabytes, which is difficult to be applied. (2) The running efficiency of the instance segmentation algorithm is relatively low, and the initial training of the model takes a long time. (3) There is not enough sample data to train the model, which makes the performance of existing deep learning methods insufficient. In our study, we utilized various data augmentation techniques, such as random flipping, rotation, and scaling, to generate additional samples from the limited dataset. These techniques effectively increase the diversity of the training samples and help prevent overfitting.

To overcome this problem, we propose an improved version of the YOLOv7 object detection algorithm that incorporates an ELAN-Net backbone and feature pyramid network (FPN) to boost the model's capability to extract relevant features from SAR images in complex backgrounds. Our suggested algorithm achieves state-of-the-art effectiveness on two benchmark datasets, demonstrating its effectiveness in addressing the research problem

of accurate ship identification and segmentation in complex SAR pictures. The main contributions in this paper are outlined in the following order:

Λ An upgraded YOLOv7 model has been proposed for instance segmentation ship detection.

Λ An effective feature extraction module has been developed and added to the improved backbone network, enhancing the network's focus on target features and making the process of feature extraction more efficient.

Λ The feature pyramid module is optimized with feature fusion to increase the accuracy of multi-scale target segmentation and further improve the speed of image processing to boost the identification and segmentation performance of the network for multi-scale ship targets.

Λ Two ship datasets, an SSDD dataset, and an HRSID dataset are used to evaluate the efficiency of the suggested technique. To test the model's robustness, two ship datasets are run (which contain images with different scales, resolutions, and scenes).

The paper is structured as follows: Part 2 explains the materials and experimental setup and demonstrates how the study acts as an organizing foundation for the remaining portions of the research. Part 3 provides a description of the research project's results and analyses. It has also shown the model's potential by comparing it with other innovatively made versions. The ablation study is described in Section 4, and Section 5 concludes the paper.

2 Related work

2.1 Deep learning-based instance segmentation

Instance segmentation in SAR photos has the advantage of combining semantic segmentation with object identification. Using semantic segmentation, each pixel of the input picture is separated into logical groups according to where the ship targets are located. It offers a better description and perception of the ship targets because of the more complex interpretation technique. As the first attempt at segmenting CNN, Mask R-CNN (Lin et al., 2017b) adds a mask branch that is analogous to the classification and regression branch in Faster R-CNN in order to forecast the segmentation mask for each region of interest (RoI). Mask Scoring R-CNN (Wang et al., 2020a) utilizes the product of the classification score and the IoU score of the mask to construct the mask score in order to increase the quality of an instance. Cascade Mask R-CNN is created by combining Mask R-CNN and Cascade R-CNN (Chen et al., 2019b). Each cascade framework adds a mask branch to complete the instance segmentation task, combining the best features of the two approaches. In order to improve identification accuracy, Hybrid Task Cascade (Chen et al., 2019b) proposes integrating the concurrent structures of identification and segmentation, which leverage semantic segmentation branches to build a spatial context for the bounding box. In recent years, a number of one-stage algorithms, notably YOLACT (Bolya et al., 2019) and SOLO (Wang et al., 2020b), have appeared that correspond to object identification methods. In addition, a few approaches such as BlendMask (Chen et al., 2020) and PolarMask (Xie et al., 2020) are built on an item identification network without anchors. Due to their speed benefits, these one-stage techniques are frequently utilized in the domain of autonomous vehicle operation and facial detection. However, in some complex ship identification tasks, the identification technique can only assess a ship's length and contour when they are important details for the particular type of ship. Improvements to the current algorithms for segmenting SAR images by an instance are not currently being made in a substantial way. The HRSID (Lin et al., 2017a) dataset was recently created for the segmentation of ship instances in SAR images.

2.2 SAR images-based ship detection

SAR can continually monitor the planet, in contrast to optical sensors, which are inoperable at night. Because SAR images do not contain information about color, texture, shape, or other aspects, they show ships differently than optical images do. Furthermore, the SAR image has a lot of noise; as a result, identifying SAR images might be difficult for researchers without the appropriate skills. Because there is a dearth of data on tagged SAR ships as an outcome, it is more challenging to identify ships from SAR images. In order to find ships in SAR images, several deep-learning techniques have been used (Sun et al., 2021a; Liu et al., 2022; Sun et al., 2022; Yasir et al., 2022). (Fan et al. 2019b) implemented a multi-level features extractor into the Faster R-CNN for polarimetric SAR ship identification. A dense attention pyramid network was created to identify SAR ships by densely connecting each feature map to the attention convolutional module (Cui et al., 2019). For pixel-by-pixel ship identification in polarimetric SAR photos, a fully convolutional network has been created (Fan et al., 2019a). The feature pyramid structure contained a split convolution block and an embedded spatial attention block (Gao et al., 2019). Against a complex background, the feature pyramid structure can identify ship items with accuracy. Wei et al. (Wei et al., 2020) created a high-resolution feature pyramid structure for ship recognition that combined high-to-low-resolution features. The challenge of ships of various sizes and crowded berthings has been addressed by the development of a multi-scale adaptive recalibration structure (Chen et al., 2019a). A one-stage SAR target identification approach was suggested by Hou et al. (Hou et al., 2019) to address the low confidence of candidates and false positives. (Kang et al. 2017) proposed a method integrating CFAR with faster R-CNN. The object proposals produced by the faster R-CNN used in this method for extracting small objects served as the protective window of the CFAR. Zou et al. (Zou et al., 2020) integrated YOLOv3 with a generative adversarial network with a multi-scale loss term to increase the accuracy of SAR ship recognition. In order to identify and recognize ships in complex-scene SAR images, Xiong et al. (Xiong et al., 2022) suggested a lightweight model that integrated several attention mechanisms into the YOLOv5-n lightweight model.

Results from using CNN methods to identify ships in SAR imagery are impressive. However, there are still two significant

areas of work that need to be addressed. One of these involves methodically combining the most recent advancements in computer vision to connect optical and SAR images. The other seeks to broaden the use of ship identification to further applications, such as instance segmentation. The two SAR image components were combined as part of this study to enhance the images' suitability for RS applications, which is another goal of the investigation.

3 Proposed improved methodology

3.1 Overall structure of our model

In addition to classifying and locating the object of interest in an image, instance segmentation also labels each pixel that is a component of the particular object instance. It enhances the identification process by associating the bounding box and mask with the object. As a result, instance segmentation will help us identify ships more accurately and will also help us deal with crowded sceneries and detect partially occluded ships. Semantic segmentation-based bottom-up and identification-based top-down techniques have been the main focus of case segmentation research for a very long time. The majority of CNN-based models and their derivation models, including RCNN, have been used for computer vision tasks such as object identification, tracking, segmentation, and classification. Faster RCNN (Chen et al., 1993) is improved by a cutting-edge technique known as Mask RCNN (He et al., 2017), which also does instance segmentation using region proposals. Additionally, it locates every instance of the target object down to the pixel level in an image.

YOLO is a single-stage object detector that can forecast a particular object in each area of the feature maps without the aid of the cascaded location classification stage. YOLO categorizes and locates the object using bounding boxes and a particular Convolution Neural Networks (CNN) network. It splits the image into an S×S; S ∈ ℤ+ grid and identifies an object as a grid cell if its focal point crosses one. A one-stage detection method called YOLO may recognize objects instantly and is very quick (Redmon et al., 2016). The YOLOV7 algorithm, which is now the most sophisticated in the YOLO series, balances the conflict between the quantity of parameters, the amount of calculation, and the performance. It also outperforms earlier iterations of the YOLO series in terms of accuracy and speed. In this paper, we used the improved Yolov7 for segmentation ship detection, and Figure 1 illustrates the outline of the method recommended in the research.

The 1024x1024 SAR images are concurrently supplied to the network feature extraction at the input end, as shown in Figure 1. In order to successfully manage the framework training, the proper ship target labeling must be delivered. The entire deep framework is divided into three sections: the backbone structure, which is primarily used to extract features from the input picture; the feature pyramid, which is used to scale the extracted features and strengthen the expression of the target feature; and the network prediction layer, which predicts the target at three scales. Finally, post-processing techniques like maximum value suppression (NMS) are used to acquire the results of the identification output.

3.2 Improved backbone networks

The two new modules that are added to the backbone structure in this research are as follows: SiLu function is used by the MP-Conv module, the E-ELAN module, and its activation function. The SiLU activation function used by the MP-Conv module is known to be more computationally efficient and effective than the traditional ReLU activation function. By incorporating the SiLU function, the MP-Conv module can better capture relevant features from SAR images, leading to improved object detection performance. Meanwhile, The MP-Conv module adopts the way of double-branch fusion to carry out super downsampling of convolution blocks, which on the one hand improves the operational efficiency of target feature extraction, on the other hand, it can fuse and enhance target feature expression. The E-ELAN module is designed

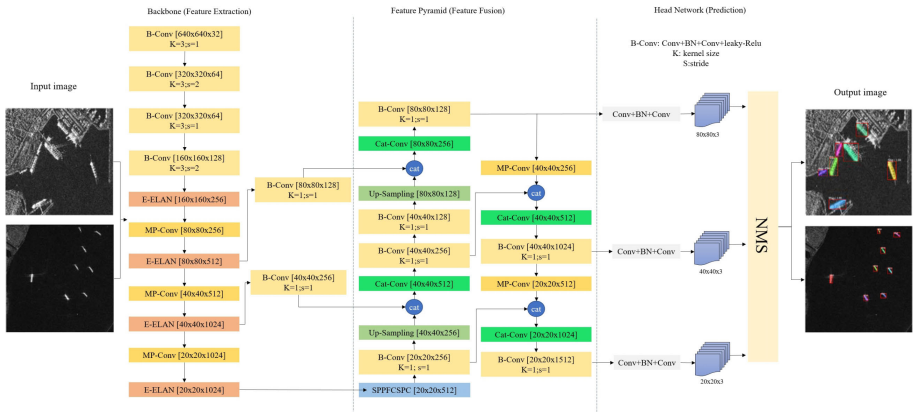


FIGURE 1
The overall structure of the proposed ship detection and segmentation model. E-ELAN, MP-Conv, Cat-Conv, and SPPSPC are some improved modules.

to boost the capability of the algorithm to retrieve spatial information from the SAR image. This is achieved by incorporating an attention mechanism that selectively weighs the feature maps based on their relevance to the final prediction. By selectively weighing the feature maps, the E-ELAN module can help the model focus on the most relevant information, leading to improved detection and segmentation performance. In addition, the E-ELAN module can stack more blocks by considering the shortest gradient path, so as to enhance the feature extraction capability of the network without significantly increasing the complexity of the model.

The E-ELAN module is an effective network structure, as shown in Figure 2A, that enables the network to learn more features and has stronger robustness by managing the shortest and longest gradient routes. The ELAN module has two branches specifically: The first branch involves using a 1x1 convolution to adjust the number of channels. The second branch, which is more difficult, first passes through a 1x1 convolution module to alter the number of channels. Then, run four 3x3 convolution modules to extract features.

The reason for selecting the fourth B-Conv as the branch for channel concatenating in Figure 2 is that we conducted extensive experiments and found that this branch provides the best performance for ship detection. Specifically, we found that by selecting the fourth B-Conv branch, the network can effectively capture features at different scales and resolutions, which is critical for accurate ship instance segmentation detection in complex background SAR images.

Two branches of the MP-Conv (Max-Pooling Convolution) module, as seen in Figure 2B, are employed for downsampling. A Max-pool, or maximal pooling, is used on the first branch. The result of maximizing is downsampling and a 1x1 convolution to change the number of layers. The second branch initially performs a 1x1 convolution to change the number of layers before passing through a convolution block with a 3x3 convolution kernel and a 2 stride. Downsampling is another application for this convolution block. In the end, the two branches' results are combined, the number of layers equals the number of input layers, but the spatial resolution is decreased by a factor of 2.

In summary, the proposed model structure is designed to enhance the model's ability to extract relevant features from SAR images and to incorporate spatial information through the attention mechanism. These improvements contribute to improved object detection and segmentation performance, as demonstrated in our experiments.

3.3 Improved neck networks

Figure 3 displays the detailed structures of two enhanced modules in the neck network. Figure 3A illustrates how similar the Cat-conv module is to the E-ELAN (Encoder Enhanced Layer Aggregation Network) module, with the exception that it chooses a different number of outputs for the second branch. Three outputs are chosen by the E-ELAN module for final addition, and five channels are chosen by the Cat-conv module for contact. The Cat-conv structure utilized in this article can assist the entire pyramid framework in aggregating multi-scale features, increasing the multi-scale representation of ship targets, which have remarkable multi-scale features in SAR images.

In order to increase the receptive field more efficiently and further promote the algorithm to adapt to different resolution images, we optimize to design of the SPPSPC (Spatial Pyramid Pooling with Spatial Pyramid Convolution) module to replace the original SPP module. As seen in Figure 3B, the first branch has four branches following the Max-pool operation. Through maximal pooling, it obtains various receptive fields. These four distinct branches signify the network's ability to process a variety of objects. That is to say, it has four receptive fields for each of its four separate scales of maximum pooling, which are utilized to differentiate between large and small targets. In this way, the SPPSPC module designed in this paper combines and optimizes the feature reorganization, which can effectively increase the accuracy of the algorithm while greatly reducing the amount of

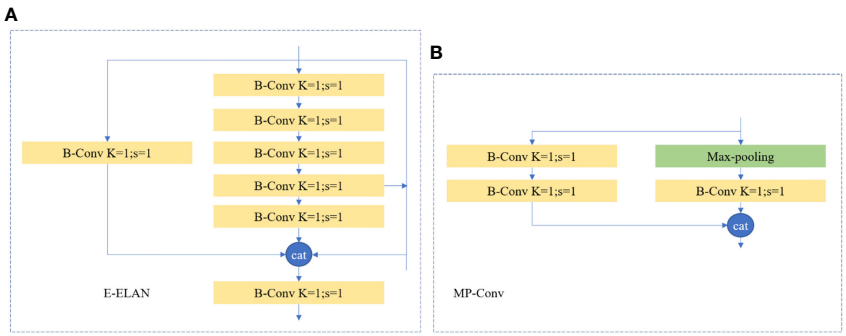


FIGURE 2
The detailed structures of two improved modules in the backbone network. **(A)** The E-ELAN module. **(B)** The MP-Conv module. "Conv" means the ordinary convolution-2D layer, "BN" means the batch normalization layer, "Max-Pooling" means the max pooling-2D layer; "k" is the kernel size, and "s" is the sliding step.

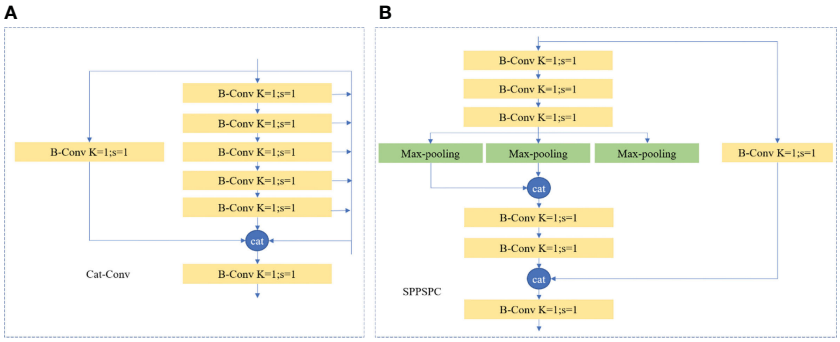


FIGURE 3
The detailed structures of two improved modules in the neck network. **(A)** The Cat-Conv module. **(B)** The SPPSPC module. "Conv" means the ordinary convolution-2D layer, "BN" means the batch normalization layer, "Max-Pooling" means the max pooling-2D layer; "k" is the kernel size, and "s" is the sliding step.

computation. The loss function used in our proposed network is a combination of three loss functions: the localization loss, the confidence loss, and the segmentation loss. The localization loss measures the difference between the predicted bounding box and the ground truth bounding box. The confidence loss measures the objectness score and the background score. Finally, the segmentation loss measures the pixel-wise difference between the predicted mask and the ground truth mask. The overall loss function is a weighted sum of these three loss functions, and it is optimized using the stochastic gradient descent (SGD) algorithm.

4 Experimental result and discussions

4.1 Dataset overview

4.1.1 HRSID dataset

The High-Resolution SAR Images Dataset for Ship Detection and Instance Segmentation (HRSID) provided by Wei et al. (Lin et al., 2017b) is made up of images from 99 Sentinel-1B imageries, 36 TerraSAR-X, and 1 TanDEM-X imagery. The resolutions of the 800 x 800-pixel images, which contain 16951 ships and 5604 sliced SAR images, range from 1 to 15 meters.

4.1.2 SSDD dataset

The first and most important stage in applying deep learning algorithms to recognize ships is the construction of a substantial and comprehensive dataset. As a result, the experiment makes use of the SSDD (Li et al., 2017) dataset, which contains 1160 SAR pictures from Sentinel-1 TerraSAR-X, and RadarSat-2 with resolutions ranging from 1m to 15m and polarizations in HV, HH, VH, and VV (Table 1). Scenes of offshore ships and inshore ships are both present in the collection as background elements.

4.2 Implementation setting

The experiments are all run on an Intel Core i9-9900KF CPU and an NVIDIA Geforce GTX 2080Ti GPU utilizing CUDA 10.1 CUDNN 7.6.5 and PyTorch 1.7.0. In each experiment, the initial learning rate is set to 0.01, the final one-cycle learning rate is set to 0.001, the momentum is set to 0.937, the optimizer weight decay is set to 0.0005, and the ship detection confidence is set to 0.7. We use the Stochastic Gradient Descent (SGD) algorithm for learning optimization. The ship instance segmentation task in this research also requires labeling the object instance as supervision information and sending it to the suggested deep learning framework for learning optimization, unlike the general detection task. In order to more thoroughly assess the proposed model, we separated the entire training set into the test set and the training set in a 7:3 ratio. We then compared the detection results with the true value annotation to assess how well the algorithm performed.

4.3 Evaluation metrics

The traditional methods for quantitatively and thoroughly assessing the effectiveness of object detectors are the estimate metrics precision (p), recall (r), intersection of union (IoU), and average precision (AP) (Everingham et al., 2010). The expert

TABLE 1 Information about the SAR imageries in detail for construction.

Dataset	Image (num)	Size (Pixel)	Satellite	Resolution (m)
HRSID (Lin et al., 2017b)	5604	800 x 800	Sentinel-1B/TerraSAR-X /TanDEM-X	1-15
SSDD (Li et al., 2017)	1160	800 x 800	RadarSat-2/TerraSAR-X/Sentinel-1	1-15

The first two SAR image examples in Figure 4 show offshore ships, whereas the last two in the row, respectively, show ships docking in ports and large ships and show the cluster-distributed tiny ships in the canal.

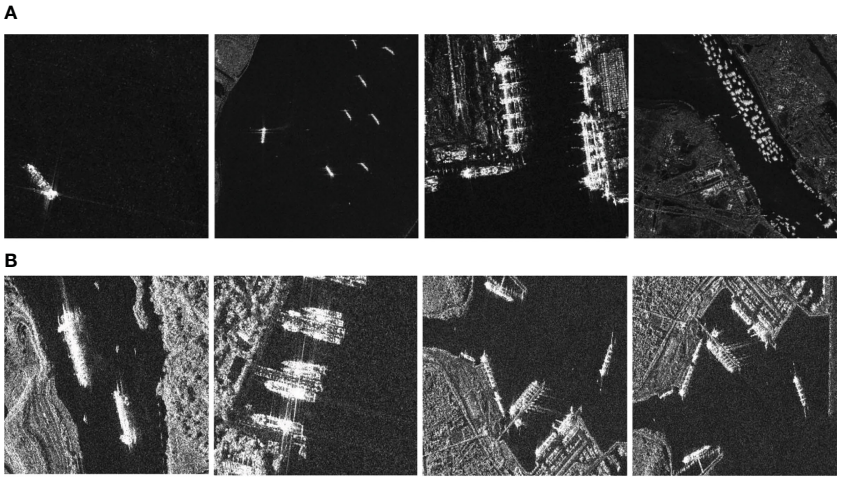


FIGURE 4
Photos are shown from the dataset used in the current paper. **(A)** some photos from the HRISD dataset and, **(B)** some photos from SSDD datasets.

annotation of the object's geographic coordinates is referred to as the ground truth in supervised learning for object identification and instance segmentation. The percentage of overlap between the expected outcome and the actual result serves as a proxy for the correlation between two variables; a higher level of overlap denotes a stronger connection and a more precise prediction. Eq (1) states that the bounding box IoU is determined by the percent of overlap between the predicted bounding box and the ground truth bounding box.

The efficiency of various techniques is evaluated using a number of recognized indicators, such as AP, r, p, and IoU, and these indications are particularly specified in the following Eq (1–5) since SAR photo object identification tasks are comparable:

$$IoU_{bbox} = \frac{Bbox_p \cap Bbox_g}{Bbox_p \cup \mathrm{Bbox}_g} \tag{1}$$

In object identification tasks, AP is a frequently used indicator that compares the proportion of properly recognized items to the total number of objects in the picture. Another often-used metric is r, which compares the fraction of successfully recognized items to the total number of objects in the picture. It is determined as the ratio of true positives (items that have been accurately identified) to the sum of true positives and false negatives (objects that were present in the image but not detected).

p is an indicator that calculates the proportion of successfully detected items concerning the total number of detected objects in the picture. It is calculated by dividing the number of true positives by the total number of true positives and false positives. IoU (Intersection over Union) is an indicator that calculates the ratio of the intersection of two bounding boxes to the union of two bounding boxes to determine the similarity between two bounding boxes (Bbox p and Bbox g). These indicators (AP, r, p, IoU) are extensively employed in the domain of SAR picture object identification to evaluate and compare the efficacy of various methodologies.

The rate of overlap between the ground mask and predicted mask, as shown in equation (2), determines the mask IoU in a manner similar to how segmentation precision is calculated.

$$IoU_{mask} = \frac{Mask_p \cap Mask_g}{Mask_p \cup \mathrm{Mask}_g} \tag{2}$$

The IoU may also be used to assess segmentation tasks such as object recognition in SAR images. The IoU is determined using equation (2), which is comparable to the calculation for IoU of bounding boxes that has been previously described. The IoU mask is the ratio of the predicted mask (Mask p) and the ground truth mask (Mask g) intersection to the union of the two masks. IoU is also known as the Jaccard Index in the context of image segmentation, which is a standard statistic for evaluating the performance of image segmentation algorithms. A high IoU score implies that the predicted mask and the ground truth mask have a high degree of overlap, indicating that the model is accurate.

During classification, algorithms may incorrectly recognize the surroundings and the objects. True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) are the four categorization findings, where TP stands for the number of successfully categorized positive samples, TN for correctly classed negative samples, FN for correctly classified missed positive samples, and FP for correctly classified false alarms in the background. These criteria establish p and r, as shown by equations (3, 4).

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \tag{3}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{4}$$

In classification tasks, the four categorization findings are used to evaluate the algorithm's performance. Precision and recall, two often used indicators in classification tasks, are calculated using TN,

FN, TP, and FP. The equation (3) is used to calculate precision, it calculates the fraction of correctly identified positive samples to the total number of positive samples. A high accuracy score suggests that the algorithm has a low number of false positives, indicating that it accurately identifies a large majority of positive samples.

The AP is established using recall and precision measurements. If the horizontal coordinate is the r value and the vertical coordinate is the precision value, as shown in equation (5), then the area under the recall-precision curve is the AP value in the Cartesian coordinate system:

$$AP = \int_0^1 P(R)dR \tag{5}$$

The mathematical average of all categories in a dataset with multiple classes is defined as the mean AP (mAP). The AP measure is extensively used to assess the effectiveness of object identification systems. The area under the recall-precision curve, which is a plot of recall *vs.* accuracy, is what it is. According to equation (5), the AP value in the Cartesian coordinate system is the definite integral of the accuracy value with respect to the recall value, ranging from 0 to 1. A greater AP value suggests that the algorithm is doing well, as seen by a larger area under the recall-precision curve.

Mean Average Precision (mAP) is a statistic used to assess the effectiveness of multi-class object identification systems. It is the average of all the AP values in a dataset. It provides an overall measure of the algorithm's performance across all classes in the dataset. A greater mAP number implies that the method performs better across all classes in the dataset.

4.4 Visualization experiment of proposed algorithm

Due to various incident angles of the radar signal, environmental conditions, polarization techniques, etc., the preprocessing SAR images include clutter noise that interferes with the feature of ships and prohibits ship identification and instance segmentation using CNN. Therefore, while building a SAR dataset for ship identification and instance segmentation, ships should be totally and precisely labeled as opposed to creating an optical RS dataset for object recognition and instance segmentation (Waqas Zamir et al., 2019). In current research work, we have established an effective and reliable algorithm for building an HR-RS dataset for CNN-based ship identification and instance segmentation. Instance segmentation's impacts on low-resolution SAR pictures may be limited in order to escape missing annotation and incorrect annotation brought on by artificial structures that resemble ships (Wang et al., 2019), which are displayed as highlighted spots in low-resolution SAR images. High-resolution remote sensing pictures are utilized to create the dataset, and the images are sliced into 800 x 800 size segments for optimal function development, such as multi-scale training.

The results of ship identification instance segmentation for SAR images using the proposed model are shown in Figures 5, 6. The ground truth mask results are shown in the first row of the figure, and the projected instance outcomes are outcomes presented in the second row. Figures 5, 6 demonstrate how our model's output is suitable for our goal of segmenting instances in HR-RS images. As missed and false alarms increase in our model, instance segmentation is carried out on the mask branch. Finally, these synthetic targets can be detected and segmented quite well, and the segmentation outcomes produced by our model are very close to reality. With the help of our model, the instance segmentation task in HR-RS images was completed successfully.

4.5 Ablation studies

We performed ablation experiments to assess the efficacy of various components in their suggested ship instance segmentation

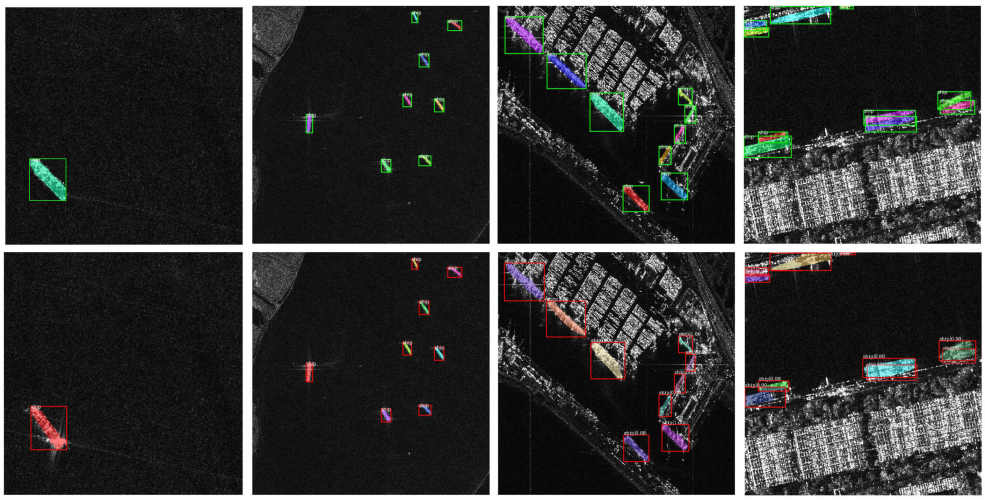


FIGURE 5
Outcomes of the proposed approach instance segmentation in the HRSID dataset (first row show the ground truth and second row is the predicted instance outcomes).

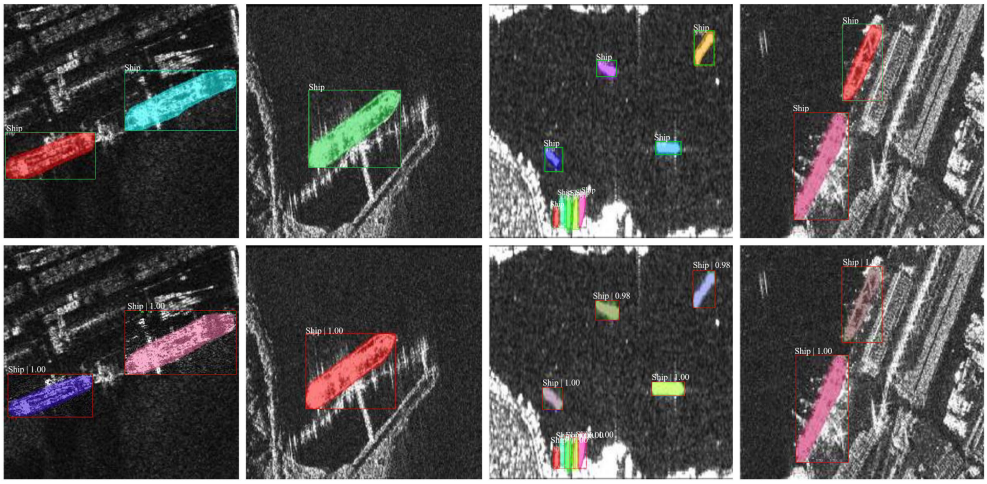


FIGURE 6
Results of the proposed model's instance segmentation in the SSDD detection dataset (the first row show the ground truth and the second row shows the predicted instance outcomes).

detection model. Table 2 shows the findings of the ablation research. As the default model, the writers used the YOLOv7 model with an input size of 640x640 pixels. The standard model had an AP of 57.8, with an AP50 of 83.7 and an AP75 of 69.5. Also we have added E-ELAN, an edge enhancement module, to the basic model in the first ablation trial. With the inclusion of E-ELAN, the AP increased to 59.4, with an AP50 of 89.6 and an AP75 of 71.9.Then we have added MP-Conv, a multi-path convolution module, to the basic model in the second ablation analysis. The inclusion of MP-Conv increased the AP to 60.7, with an AP50 of 83.9 and an AP75 of 69.8. Cat-Conv, a channel attention transfer convolution module, was added to the baseline model in the third ablation trial. Cat-Conv increased the AP to 62.3, with an AP50 of 83.1 and an AP75 of 68.3. Also we have added SPPSPC, a spatial pyramid pooling module, and convolution to the baseline model in the fourth ablation trial. SPPSPC increased the AP to 63.5, with an AP50 of 87.8 and an AP75 of 73.5. In the last, the authors added all of the previously stated modules (E-ELAN, MP-Conv, Cat-Conv, and SPPSPC) to the baseline model in the fifth and concluding ablation trial. The finished model had the greatest AP of 69.7, as well as an AP50 of 94.9 and an AP75 of 86.5. The authors discovered that incorporating all four modules greatly enhanced the baseline model's performance, particularly in terms of accuracy and recall, showing the efficacy of their suggested model for real-time ship instance segmentation recognition in complicated backdrop SAR images.

4.6 Comparison with other state-of-the-art techniques

Figures 7 and 8 show the qualitative outcomes of our model and the comparable algorithm on the SSDD and HRSID dataset, individually, to further validate the efficiency of instance segmentation and ship identification. Row 1 displays the ground-truth mask, while rows 2 to 6 display the results of Faster R-CNN, Cascade R-CNN, Mask R-CNN, and Hybrid Task Cascade, respectively. When compared to existing instance segmentation techniques, the results of our improved model can accurately recognize and separate artificial targets in a variety of scenes, as shown in row 7. The expected instance masks, in particular, precisely cover these contrived objectives. As a result of our model's nearly complete elimination of false alarms and missed detections, our mask branch consistently accomplishes superior instance segmentation. When contrast to bounding box identification approaches like Faster R-CNN, Mask R-CNN,

TABLE 2 The ablation experiment study.

Model	Input size	E-ELAN	MP-Conv	Cat-Conv	SPPSPC	AP	AP50	AP75	APS	APM	APL
Yolov7	640x640	–	–	–	–	57.8	83.7	69.5	57.3	60.6	24.5
	640x640	✓	–	–	–	59.4	89.6	71.9	59.1	60.6	39.7
	640x640	–	✓	–	–	60.7	83.9	69.8	56.9	61.2	30.4
	640x640	–	–	✓	–	62.3	83.1	68.3	60.7	63.5	47.8
	640x640	–	–	–	✓	63.5	87.8	73.5	65.5	67.4	45.5
	640x640	✓	✓	✓	✓	69.7	94.9	86.5	73.4	76.8	58.6

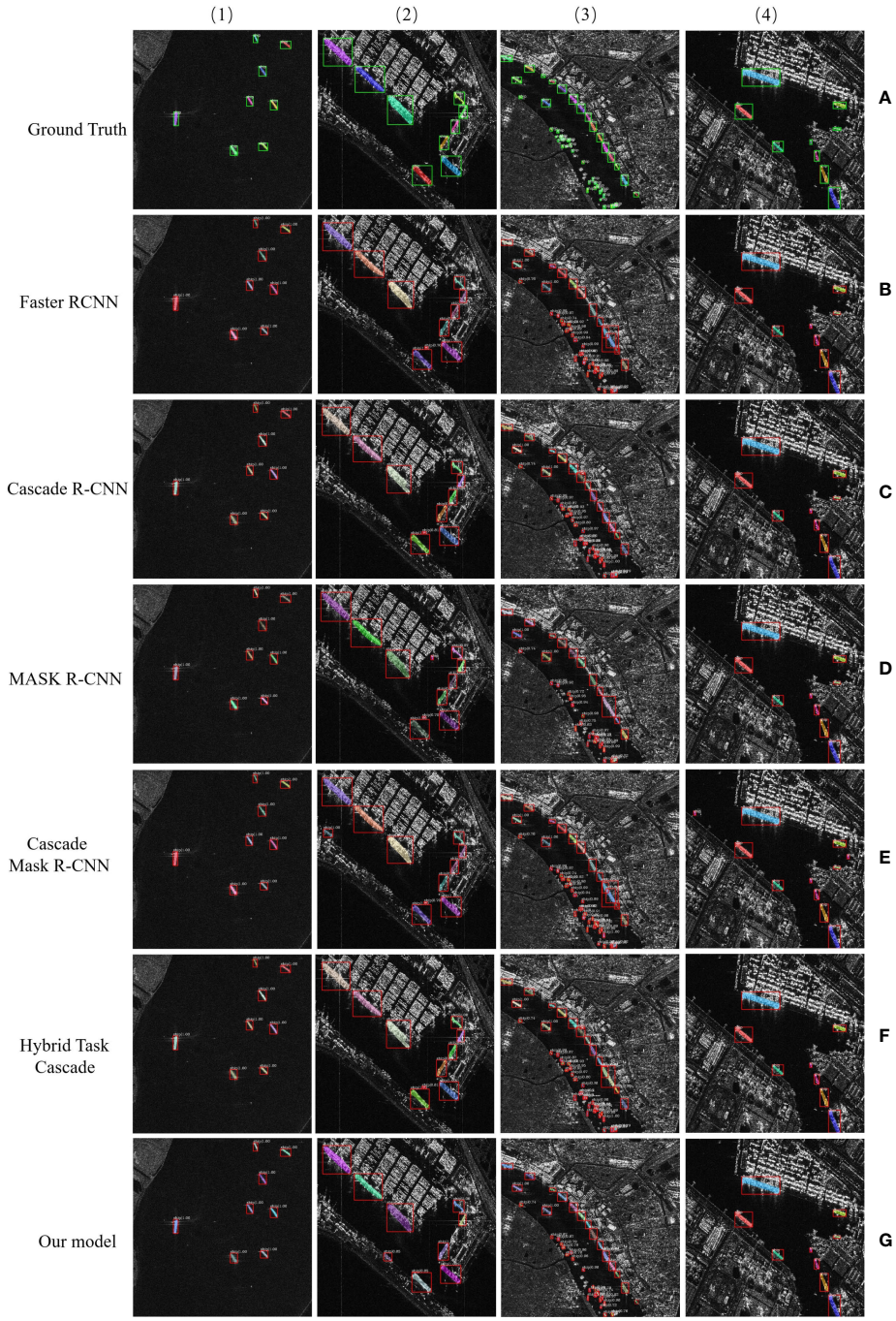


FIGURE 7
Outcomes of CNN-based techniques for visual ship identification instance segmentation using the HRSID dataset. Outcomes from **(A)** illustrate the ground truth, **(B)** the Faster-R-CNN technique, **(C)** the Cascade R-CNN, **(D)** the Mask R-CNN, **(E)** the Cascade Mask R-CNN, **(F)** the Hybrid Task Cascade, and **(G)** the results from our proposed method.

Cascade Mask R-CNN, Hybrid Task Cascade, and Cascade R-CNN, instance segmentation outcomes are more closely connected to the shape of the original targets. Additionally, separate instances within the same category can be distinguished using the instance segmentation. The ships in Figures 7, 8 stand out because to their dissimilar colors, and in addition, the suggested model, when compared to other instance segmentation approaches, has no false alarms and no missed targets detection while also producing better results for mask segmentation. The results from the HRSID and SSDD dataset show that our technique is appropriate for instance segmentation in HR-RS photos and outperforms existing instance segmentation strategies when it comes to mask segmentation.

To quantitatively assess the achievement of instance segmentation, we compared the suggested approach with other cutting-edge approaches on the HRSID and SSDD in Tables 3 and

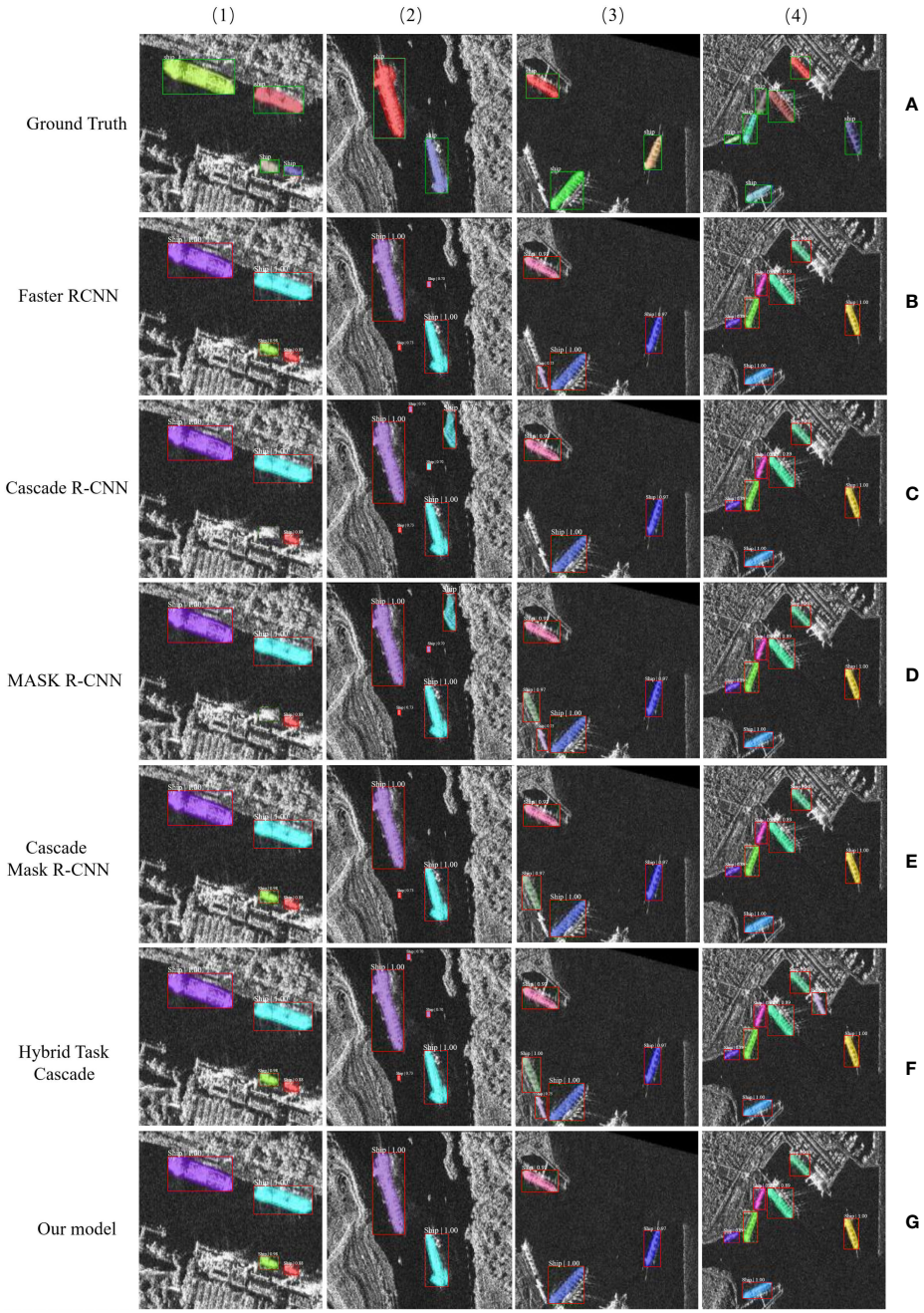


FIGURE 8
Outcomes of CNN-based techniques for visual ship identification instance segmentation using the SSDD. Results from **(A)** illustrate the ground truth, **(B)** the Faster-R-CNN technique, **(C)** the Cascade R-CNN, **(D)** the Mask R-CNN, **(E)** the Cascade Mask R-CNN, **(F)** the Hybrid Task Cascade, and **(G)** the results from our proposed method.

4. Faster R-CNN, Mask R-CNN, Cascade R-CNN, Cascade Mask R-CNN, and Hybrid Task Cascade are some of these techniques. Tables 3 and 4 show that the suggested strategy achieves the maximum ap of 69.7%. Hybrid Task Cascade and our model outperform Faster R-CNN, Cascade R-CNN, Mask R-CNN, Cascade Mask R-CNN, and Cascade R-CNN by 6.3%, 3.2%, 4.5%, 0.8%, and 2.4%, respectively. In summary, the recommended method has superior instance segmentation effectiveness and better precise predicted instance masks on the HRSID dataset compared to other instance segmentation algorithms. The reduced parameter count, and computational expense are due to the use of the SiLU activation function, which is more computationally efficient than the traditional ReLU activation function. Additionally, the E-ELAN module selectively weighs the feature maps, further reducing the computational expense without compromising performance. The AP50 score of our model is 94.9%, which is also 10.2% higher than Faster R-CNN, 9.3% higher than Cascade R-CNN, 7.4% higher than Mask R-CNN, 8.2% higher than Cascade Mask R-CNN, and 7.3% higher than Hybrid Task Cascade. Our model achieves an AP75 score of 86.5%, which is an

TABLE 3 Comparing to various cutting-edge methods on the HRSID dataset.

Methods	Backbone	Time (ms)	Model (Size)	AP	AP50	AP75	APS	APM	APL
Faster R-CNN(Ren et al., 2015)	ResNet-50 +FPN ResNet-101 +FPN	52.6 64.2	330M 482M	64.9 63.4	84.6 84.7	71.5 71.6	65.1 65.7	66.2 67.3	17.8 25.3
Cascade R-CNN(Cai and Vasconcelos, 2019)	ResNet-50 +FPN ResNet-101 +FPN	73.9 85.5	552M 704M	67.8 66.5	85.8 85.6	77.6 77.3	68.6 68.2	68.8 69.7	29.9 28.8
Mask R-CNN (He et al., 2017)	ResNet-50 +FPN ResNet-101 +FPN	53.7 62.9	351M 503M	66.8 65.2	87.3 87.5	74.9 74.0	67.9 67.3	67.8 69.3	18.4 24.3
Cascade Mask R-CNN(Cai and Vasconcelos, 2019)	ResNet-50 +FPN ResNet-101 +FPN	73.0 87.1	615M 768M	68.7 68.9	86.1 86.7	76.6 76.8	69.4 69.8	68.5 70.6	21.5 22.9
Hybrid Task Cascade (Chen et al., 2019b)	ResNet-50 +FPN ResNet-101 +FPN	118.9 134.6	639M 791M	67.1 67.3	88.4 87.6	79.8 79.3	70.3 70.8	72.6 73.6	39.0 32.8
Our Model	ELAN-Net	87	403M	69.7	94.9	86.5	73.4	76.8	58.6

improvement of 14.9% over Faster R-CNN, 9.3% over Cascade R-CNN, 12.5% over Mask R-CNN, 9.7% over Cascade Mask R-CNN, and 7.2% over Hybrid Task Cascade. Mask segmentation has proven to be more precise and superior to other state-of-the-art techniques, such as segmentation utilizing the HRSID dataset. The efficacy of large medium, and small targets on the HRSID dataset has also improved, according to APS, APM, and APL.

Table 3 shows that our model achieves a 70.3% AP, which represents an improvement of 11.7% compared to Faster R-CNN, 9.2% compared to Cascade R-CNN, 13.8% compared to Mask R-CNN, 10.3% compared to Cascade Mask R-CNN, and 2.5% compared to Hybrid Task Cascade. In summary, the recommended model has superior instance segmentation effectiveness and more precise predicted instance masks when compared to previous instance segmentation algorithms on the SSDD dataset. The AP50 score of our model is also 94.7%, which is an improvement of 15.7% over Faster R-CNN, 3.3% over Cascade R-CNN, 4% over Mask R-CNN, 7.7% over Cascade Mask R-CNN, and 2% over Hybrid Task Cascade. Our model obtains an AP75 of 76.5 percent, which is an improvement of 11% over Faster R-CNN, 9.8% over Cascade R-CNN, 10.8% over Mask R-CNN, 8.8% over Cascade Mask R-CNN, and 1.6% over Hybrid Task Cascade. It has been proven that segmentation using the mask will be more accurate and superior than segmentation using other cutting-edge techniques, such as segmentation on the SSDD dataset. According to APL, APM, and APS, the HRSID dataset's small, medium, and

TABLE 4 Comparing to various cutting-edge methods on the SSDD dataset.

Methods	Backbone	Time (ms)	Model (Size)	AP	AP50	AP75	APS	APM	APL
Faster R-CNN(Ren et al., 2015)	ResNet-50+FPN ResNet-101+FPN	55.5 66.1	330M 482M	57.5 58.6	78.1 79.0	64.2 65.5	42.8 43.6	57.8 58.1	62.7 61.6
Cascade R-CNN (Cai and Vasconcelos, 2019)	ResNet-50+FPN ResNet-101+FPN	61.9 70.2	552M 704M	60.7 61.1	90.2 91.4	67.8 66.7	46.4 45.7	61.7 61.4	66.4 61.3
Mask R-CNN (He et al., 2017)	ResNet-50+FPN ResNet-101+FPN	63.0 72.3	351M 503M	55.3 56.5	91.3 90.7	64.8 65.8	41.8 41.1	55.7 54.4	59.9 60.2
Cascade Mask R-CNN(Cai and Vasconcelos, 2019)	ResNet-50+FPN ResNet-101+FPN	85.6 93.8	615M 768M	60.2 59.7	88.5 87.2	66.8 67.7	47.5 46.2	63.5 63.0	66.4 65.7
Hybrid Task Cascade (Chen et al., 2019b)	ResNet-50+FPN ResNet-101+FPN	153.2 168.5	639M 791M	68.7 67.8	91.2 92.6	75.5 74.9	52.2 54.6	68.9 67.8	70.5 73.8
Our Model	ELAN-Net	96	403M	70.3	94.7	76.5	55.9	70.2	75.1

large target efficacy has also enhanced. We achieve the similar achievement as our model on the NWPU VHR-10 dataset under several AP indicators, and some AP indicators even outperform it.

Tables 3, 4 show how our model performs better with fewer parameters and less computational expense. The proposed model incorporates several improvements to the YOLOv7 backbone architecture, including the addition of an ELAN-Net backbone and FPN, the SiLU activation function, and the E-ELAN module. These improvements allow the model to more effectively extract and use relevant features from SAR images, resulting in improved detection and segmentation performance. Moreover, the proposed model achieves this improved performance while using fewer parameters and less computational expense compared to other modern models, as shown in Tables 3 and 4. The reduced parameter count and computational expense are due to the use of the SiLU activation function, which is more computationally efficient than the traditional ReLU activation function. Additionally, the E-ELAN module selectively weighs the feature maps, further reducing the computational expense without compromising performance.

Furthermore, with comparable model sizes and levels of computational complexity, our models outperform the Mask Scoring R-CNN and Mask R-CNN. Comparing our models to Hybrid Task Cascade and Cascade Mask R-CNN, we find that our models outperform them while consuming less processing power and having a smaller model size. Our network is therefore better than other modern algorithms in terms of model size and processing complexity.

In order to assess the detectors' capacities to locate the ship in complex situations and to test their capacity to deliver adequately observable results, some complex scenarios are added to the datasets. The findings demonstrate that complex situations, like those containing nearby ships and small ships scattered in a cluster, continue to provide a challenge to detectors. The generated mask may accurately show the distribution of ships with their concrete shape pixel-by-pixel with regard to the visual identification outcomes in instance segmentation, laying the groundwork for further instance segmentation investigations. As a result, when compared to other cutting-edge techniques, our model creates instance masks that are more precise and improves the performance of instance segmentation in HR-RS images.

The object detection of RS images has been shown to have problems by CNN. YOLOv7 was actually created as the fundamental detecting network, whereas the ELAN-Net backbone network was designed for advancement. The results of our studies demonstrate that the enhanced algorithm we built would considerably improve the identification efficiency of small-scale items in RS pictures and can increase the accuracy of multi-scale object segmentation. The HRSID and SSDD datasets were used for our investigation because there are no established, open remote sensing mask datasets available, and there might only be a few different varieties. We also need to conduct further research to improve and advance the model inference speed. However, using fuzzy preprocessing techniques to images is also necessary because the processed images are frequently affected by unknown factors (Versaci et al., 2015). Our next study will focus on solving the aforementioned issues, and in order to test our new models, we will first look for and create more RS mask datasets with a wider range of object classes. Additionally, we will use more accurate and representative datasets. The next phase of our research will involve creating a lightweight framework model that will speed up inference without sacrificing identification accuracy.

In summary, our proposed model achieves better performance with fewer parameters and less computational expense by incorporating several improvements to the YOLOv7 backbone architecture, and by using the SiLU activation function and the E-ELAN module to more effectively extract and use relevant features from SAR pictures.

5 Conclusions

The field of aerospace and remote sensing (RS) domains is heavily influenced by instance segmentation and object recognition tasks, which have a wide range of potential applications in various real-world scenarios. In recent times, the importance of ship identification in RS satellite images has increased. While most current algorithms identify ships using rectangular bounding boxes, they do not segment pixels. As a result, our research offers an enhanced YOLOv7 one-stage detection technique for ship segmentation and identification in RS imagery, capable of accurately recognizing and segmenting ships at the pixel level. We have redesigned the network structure to adapt to the task of ship target segmentation and added two feature optimization modules to the backbone network to increase the robustness of network feature extraction. In addition, we improved the network feature fusion structure and enhanced the prediction capability of multi-scale targets by optimizing the model acceptance domain. Based on the experimental outcomes on the SSDD and HRSID datasets, our model demonstrates improved accuracy in predicting instance masks, promoting the success of instance segmentation in HR-RS imaging and encouraging further advancements in mask prediction accuracy. Our proposed method outperforms existing methods for segmenting ships in remote sensing images, and we plan to extend our research to the segmentation of objects in drone images. While our proposed approach has limitations in handling extremely small or crowded ship instances, we acknowledge this limitation and suggest further optimization of the network architecture and training strategies. Additionally, we have not yet explored the potential of other advanced techniques such as depthwise separable convolution neural network, balance learning, and attention mechanisms, which could be interesting directions for future research. In summary, our proposed approach provides a more precise and effective solution for ship segmentation and identification in RS imagery, and our future work will focus on extending the application of our proposed method to other remote sensing scenarios.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

MY, LZ. Methodology, MY, and SL. software, MY, SL, and LZ. Validation, WJ, SL, and LZ. Formal analysis, MH, QI, and AC. The investigation, ML. Resources, LZ. Data curation, MY, SM, QI, and QY. Writing-original draft preparation, MY. Writing-review and editing, JW, LZ, and SL. Visualization, SL, LZ. Supervision, JW, and SL. Project administration, JW. Funding acquisition, LZ. All authors contributed to the article and approved the submitted version.

Funding

This work is supported by Global atmospheric aerosol dataset development (HX20220168).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Bolya, D., Zhou, C., Xiao, F., and Lee, Y. J. (2019). "Yolact: real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*. 9157–9166.

Cai, Z., and Vasconcelos, N. (2019). Cascade r-CNN: high quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 1483–1498. doi: 10.1109/TPAMI.2019.2956516

Chang, Y.-L., Anagaw, A., Chang, L., Wang, Y. C., Hsiao, C.-Y., and Lee, W.-H. (2019). Ship detection based on YOLOv2 for SAR imagery. *Remote Sens.* 11, 786. doi: 10.3390/rs11070786

Chen, S.-W., Cui, X.-C., Wang, X.-S., and Xiao, S.-P. (2021). Speckle-free SAR image ship detection. *IEEE Trans. Image Process.* 30, 5969–5983. doi: 10.1109/TIP.2021.3089936

Chen, C., He, C., Hu, C., Pei, H., and Jiao, L. (2019a). MSARN: a deep neural network based on an adaptive recalibration mechanism for multiscale and arbitrary-oriented SAR ship detection. *IEEE Access* 7, 159262–159283. doi: 10.1109/ACCESS.2019.2951030

Chen, S., Mulgrew, B., and Grant, P. M. (1993). A clustering technique for digital communications channel equalization using radial basis function networks. *IEEE Trans. Neural Networks* 4, 570–590. doi: 10.1109/72.238312

Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., et al. (2019b). "Hybrid task cascade for instance segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4974–4983.

Chen, H., Sun, K., Tian, Z., Shen, C., Huang, Y., and Yan, Y. (2020). "Blendmask: top-down meets bottom-up for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8573–8581.

Cui, Z., Li, Q., Cao, Z., and Liu, N. (2019). Dense attention pyramid networks for multi-scale ship detection in SAR images. *IEEE Trans. Geosci. Remote Sens.* 57, 8983–8997. doi: 10.1109/TGRS.2019.2923988

Dai, J., He, K., and Sun, J. (2016). "Instance-aware semantic segmentation *via* multi-task network cascades," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3150–3158.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision* 88, 303–338. doi: 10.1007/s11263-009-0275-4

Fan, Q., Chen, F., Cheng, M., Lou, S., Xiao, R., Zhang, B., et al. (2019a). Ship detection using a fully convolutional network with compact polarimetric SAR images. *Remote Sens.* 11, 2171. doi: 10.3390/rs11182171

Fan, W., Zhou, F., Bai, X., Tao, M., and Tian, T. (2019b). Ship detection using deep convolutional neural networks for PolSAR images. *Remote Sens.* 11, 2862. doi: 10.3390/rs11232862

Gao, F., Shi, W., Wang, J., Yang, E., and Zhou, H. (2019). Enhanced feature extraction for ship detection from multi-resolution and multi-scene synthetic aperture radar (SAR) images. *Remote Sens.* 11, 2694. doi: 10.3390/rs11222694

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*. 2961–2969.

Hou, B., Ren, Z., Zhao, W., Wu, Q., and Jiao, L. (2019). Object detection in high-resolution panchromatic images using deep models and spatial template matching. *IEEE Trans. Geosci. Remote Sens.* 58, 956–970. doi: 10.1109/TGRS.2019.2942103

Kang, M., Leng, X., Lin, Z., and Ji, K. (2017). "A modified faster r-CNN based on CFAR algorithm for SAR ship detection," in *2017 International Workshop on Remote Sensing with Intelligent Processing (RSIP): IEEE*. 1–4.

Kong, W., Liu, S., Xu, M., Yasir, M., Wang, D., and Liu, W. (2023). Lightweight algorithm for multi-scale ship detection based on high-resolution SAR images. *Int. J. Remote Sens.* 44, 1390–1415. doi: 10.1080/01431161.2023.2182652

Li, J., Qu, C., and Shao, J. (2017). "Ship detection in SAR images based on an improved faster r-CNN," in *2017 SAR in Big Data Era: Models, Methods and Applications (BIGSARDATA): IEEE*. 1–6.

Li, K., Zhang, M., Xu, M., Tang, R., Wang, L., and Wang, H. (2022). Ship detection in SAR images based on feature enhancement swin transformer and adjacent feature fusion. *Remote Sens.* 14, 3186. doi: 10.3390/rs14133186

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2117–2125. doi: 10.48550/arXiv.1708.02002

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). Focal loss for dense object detection. *Proc. IEEE Int. Conf. Comput. Vision*, 2980–2988. doi: 10.48550/arXiv.1708.02002

Liu, S., Kong, W., Chen, X., Xu, M., Yasir, M., Zhao, L., et al. (2022). Multi-scale ship detection algorithm based on a lightweight neural network for spaceborne SAR images. *Remote Sens.* 14, 1149. doi: 10.3390/rs14051149

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8759–8768.

Liu, R., Wang, X., Lu, H., Wu, Z., Fan, Q., Li, S., et al. (2021a). SCCGAN: style and characters inpainting based on CGAN. *Mobile Networks Appl.* 26, 3–12. doi: 10.1007/s11036-020-01717-x

Liu, Y., Zhang, Z., Liu, X., Wang, L., and Xia, X. (2021b). Efficient image segmentation based on deep learning for mineral image classification. *Advanced Powder Technol.* 32, 3885–3903. doi: 10.1016/j.apt.2021.08.038

Mou, L., and Zhu, X. X. (2018). Vehicle instance segmentation from aerial image and video using a multitask learning residual fully convolutional network. *IEEE Trans. Geosci. Remote Sens.* 56, 6699–6711. doi: 10.1109/TGRS.2018.2841808

Nie, X., Duan, M., Ding, H., Hu, B., and Wong, E. K. (2020). Attention mask r-CNN for ship detection and segmentation from remote sensing images. *IEEE Access* 8, 9325–9334. doi: 10.1109/ACCESS.2020.2964540

Qian, X., Lin, S., Cheng, G., Yao, X., Ren, H., and Wang, W. (2020). Object detection in remote sensing images based on improved bounding box regression and multi-level features fusion. *Remote Sens.* 12, 143. doi: 10.3390/rs12010143

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* 28. doi: 10.48550/arXiv.1506.01497

Shao, Z., Zhang, X., Zhang, T., Xu, X., and Zeng, T. (2022). RBFA-net: a rotated balanced feature-aligned network for rotated SAR ship detection and classification. *Remote Sens.* 14, 3345. doi: 10.3390/rs14143345

Su, N., He, J., Yan, Y., Zhao, C., and Xing, X. (2022). SII-net: spatial information integration network for small target detection in SAR images. *Remote Sens.* 14, 442. doi: 10.3390/rs14030442

Su, H., Wei, S., Yan, M., Wang, C., Shi, J., and Zhang, X. (2019). "Object detection and instance segmentation in remote sensing imagery based on precise mask r-CNN," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium: IEEE*. 1454–1457.

Sun, Z., Dai, M., Leng, X., Lei, Y., Xiong, B., Ji, K., et al. (2021a). An anchor-free detection method for ship targets in high-resolution SAR images. *IEEE J. Selected Topics Appl. Earth Observations Remote Sens.* 14, 7799–7816. doi: 10.1109/JSTARS.2021.3099483

Sun, Z., Lei, Y., Leng, X., Xiong, B., and Ji, K. (2022). "An improved oriented ship detection method in high-resolution SAR image based on YOLOv5," in *2022 Photonics & Electromagnetics Research Symposium (PIERS): IEEE*. 647–653.

Sun, Z., Leng, X., Lei, Y., Xiong, B., Ji, K., and Kuang, G. (2021b). BiFA-YOLO: a novel YOLO-based method for arbitrary-oriented ship detection in high-resolution SAR images. *Remote Sens.* 13, 4209. doi: 10.3390/rs13214209

Versaci, M., Calcagno, S., and Morabito, F. C. (2015). "Fuzzy geometrical approach based on unit hyper-cubes for image contrast enhancement," in *2015 IEEE international conference on signal and image processing applications (ICSIPA): IEEE*. 488–493.

Wang, X., Kong, T., Shen, C., Jiang, Y., and Li, L. (2020a). "Solo: segmenting objects by locations," in *European Conference on Computer Vision*. 649–665.

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020b). "CSPNet: a new backbone that can enhance learning capability of CNN," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 390–391.

Wang, J., Lu, C., and Jiang, W. (2018). Simultaneous ship detection and orientation estimation in SAR images based on attention module and angle regression. *Sensors* 18, 2851. doi: 10.3390/s18092851

Wang, Y., Wang, C., Zhang, H., Dong, Y., and Wei, S. (2019). Automatic ship detection based on RetinaNet using multi-resolution gaofen-3 imagery. *Remote Sens.* 11, 531. doi: 10.3390/rs11050531

Waqas Zamir, S., Arora, A., Gupta, A., Khan, S., Sun, G., Shahbaz Khan, F., et al. (2019). "Isaid: a large-scale dataset for instance segmentation in aerial images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 28–37.

Wei, S., Su, H., Ming, J., Wang, C., Yan, M., Kumar, D., et al. (2020). Precise and robust ship detection for high-resolution SAR imagery based on HR-SDNet. *Remote Sens.* 12, 167. doi: 10.3390/rs12010167

Wu, Y., Sheng, H., Zhang, Y., Wang, S., Xiong, Z., and Ke, W. (2022). Hybrid motion model for multiple object tracking in mobile devices. *IEEE Internet Things J.* doi: 10.1109/JIOT.2022.3219627

Xie, E., Sun, P., Song, X., Wang, W., Liu, X., Liang, D., et al. (2020). "Polarmask: single shot instance segmentation with polar representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12193–12202.

Xiong, B., Sun, Z., Wang, J., Leng, X., and Ji, K. (2022). A lightweight model for ship detection and recognition in complex-scene SAR images. *Remote Sens.* 14, 6053. doi: 10.3390/rs14236053

Xu, X., Feng, Z., Cao, C., Li, M., Wu, J., Wu, Z., et al. (2021). An improved swin transformer-based model for remote sensing object detection and instance segmentation. *Remote Sens.* 13, 4779. doi: 10.3390/rs13234779

Xu, X., Zhang, X., Shao, Z., Shi, J., Wei, S., Zhang, T., et al. (2022a). A group-wise feature enhancement-and-Fusion network with dual-polarization feature enrichment for SAR ship detection. *Remote Sens.* 14, 5276. doi: 10.3390/rs14205276

Xu, X., Zhang, X., Zhang, T., Yang, Z., Shi, J., and Zhan, X. (2022b). Shadow-Background-Noise 3D spatial decomposition using sparse low-rank Gaussian properties for video-SAR moving target shadow enhancement. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2022.3223514

Yasir, M., Jianhua, W., Mingming, X., Hui, S., Zhe, Z., Shanwei, L., et al. (2022). Ship detection based on deep learning using SAR imagery: a systematic literature review. *Soft Computing* 1-22. doi: 10.1007/s00500-022-07522-w

Yasir, M., Jianhua, W., Mingming, X., Hui, S., Zhe, Z., Shanwei, L., et al. (2023a). Ship detection based on deep learning using SAR imagery: a systematic literature review. *Soft Computing* 27, 63–84. doi: 10.1007/s00500-022-07522-w

Yasir, M., Shanwei, L., Xu, M., Sheng, H., Hossain, M. S., Colak, A. T. I., et al. (2023b). Multi-scale ship target detection using SAR images based on improved Yolov5. *Front. Mar. Science*. doi: 10.3389/fmars.2022.1086140

Yin, M., Zhu, Y., Yin, G., Fu, G., and Xie, L. (2022). Deep feature interaction network for point cloud registration, with applications to optical measurement of blade profiles. *IEEE Trans. Ind. Informatics*. doi: 10.1109/TII.2022.3220889

Zeng, X., Wei, S., Shi, J., and Zhang, X. (2021). A lightweight adaptive roi extraction network for precise aerial image instance segmentation. *IEEE Trans. Instrumentation Measurement* 70, 1–17. doi: 10.1109/TIM.2021.3121485

Zhang, Y., Guo, L., Wang, Z., Yu, Y., Liu, X., and Xu, F. (2020c). Intelligent ship detection in remote sensing images based on multi-layer convolutional feature fusion. *Remote Sens.* 12, 3316. doi: 10.1016/j.isprsjprs.2020.05.016

Zhang, J., Lin, S., Ding, L., and Bruzzone, L. (2020a). Multi-scale context aggregation for semantic segmentation of remote sensing images. *Remote Sens.* 12, 701. doi: 10.3390/rs12040701

Zhang, H., Luo, G., Li, J., and Wang, F.-Y. (2021a). C2FDA: coarse-to-fine domain adaptation for traffic object detection. *IEEE Trans. Intelligent Transportation Syst.* 23, 12633–12647. doi: 10.1109/JSTARS.2021.3102989

Zhang, S., Wu, R., Xu, K., Wang, J., and Sun, W. (2019a). R-CNN-based ship detection from high resolution remote sensing imagery. *Remote Sens.* 11, 631. doi: 10.3390/rs11060631

Zhang, T., Zeng, T., and Zhang, X. (2023). Synthetic aperture radar (SAR) meets deep learning. *Remote Sens.*15, 2.

Zhang, T., and Zhang, X. (2021b). "Integrate traditional hand-crafted features into modern CNN-based models to further improve SAR ship classification accuracy," in *2021 7th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR): IEEE*. 1–6.

Zhang, T., and Zhang, X. (2019). High-speed ship detection in SAR images based on a grid convolutional neural network. *Remote Sens.* 11, 1206. doi: 10.3390/rs11101206

Zhang, T., and Zhang, X. (2021a). Injection of traditional hand-crafted features into modern CNN-based models for SAR ship classification: what, why, where, and how. *Remote Sens.* 13, 2091. doi: 10.3390/rs13112091

Zhang, T., and Zhang, X. (2022a). A full-level context squeeze-and-excitation ROI extractor for SAR ship instance segmentation. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2022.3166387

Zhang, T., and Zhang, X. (2022b). HTC+ for SAR ship instance segmentation. *Remote Sens.* 14, 2395. doi: 10.3390/rs14102395

Zhang, T., and Zhang, X. (2022c). A mask attention interaction and scale enhancement network for SAR ship instance segmentation. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2022.3189961

Zhang, T., and Zhang, X. (2022d). A polarization fusion network with geometric feature embedding for SAR ship classification. *Pattern Recognition* 123, 108365. doi: 10.1016/j.patcog.2021.108365

Zhang, T., Zhang, X., Liu, C., Shi, J., Wei, S., Ahmad, I., et al. (2021b). Balance learning for ship detection from synthetic aperture radar remote sensing imagery. *ISPRS J. Photogrammetry Remote Sens.* 182, 190–207.

Zhang, T., Zhang, X., Shi, J., and Wei, S. (2019b). Depthwise separable convolution neural network for high-speed SAR ship detection. *Remote Sens.* 11, 2483. doi: 10.3390/rs11212483

Zhang, T., Zhang, X., Shi, J., and Wei, S. (2020b). HyperLi-net: a hyper-light deep learning network for high-accurate and high-speed ship detection from synthetic aperture radar imagery. *ISPRS J. Photogrammetry Remote Sens.* 167, 123–153. doi: 10.1109/TGRS.2022.3167569

Zhou, W., Liu, J., Lei, J., Yu, L., and Hwang, J.-N. (2021). GMNet: graded-feature multilabel-learning network for RGB-thermal urban scene semantic segmentation. *IEEE Trans. Image Process.* 30, 7790–7802. doi: 10.1109/TIP.2021.3109518

Zhou, W., Lv, Y., Lei, J., and Yu, L. (2019). Global and local-contrast guides content-aware fusion for RGB-d saliency prediction. *IEEE Trans. Systems Man Cybernetics: Syst.* 51, 3641–3649. doi: 10.1109/TSMC.2019.2957386

Zhou, W., Wang, H., and Wan, Z. (2022b). Ore image classification based on improved CNN. *Comput. Electrical Eng.* 99, 107819. doi: 10.1016/j.compeleceng.2022.107819

Zhou, G., Yang, F., and Xiao, J. (2022a). Study on pixel entanglement theory for imagery classification. *IEEE Trans. Geosci. Remote Sens.* 60, 1–18.

Zhu, H., and Zhao, R. (2022). Isolated Ni atoms induced edge stabilities and equilibrium shapes of CVD-prepared hexagonal boron nitride on the Ni (111) surface. *New J. Chem.* 46, 17496–17504. doi: 10.1039/D2NJ03735A

Zong, C., and Wan, Z. (2022). Container ship cell guide accuracy check technology based on improved 3D point cloud instance segmentation. *Brodogradnja: Teorija i praksa brodogradnje i pomorske tehnike* 73, 23–35. doi: 10.21278/brod73102

Zong, C., and Wang, H. (2022). An improved 3D point cloud instance segmentation method for overhead catenary height detection. *Comput. electrical Eng.* 98, 107685. doi: 10.1016/j.compeleceng.2022.107685

Zou, L., Zhang, H., Wang, C., Wu, F., and Gu, F. (2020). Mw-acgan: generating multiscale high-resolution SAR images for ship detection. *Sensors* 20, 6673. doi: 10.3390/s20226673

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 08 May 2023
DOI 10.3389/fmars.2023.1133881

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Huimin Lu,
Kyushu Institute of Technology, Japan
Younggun Cho,
Inha University, Republic of Korea

*CORRESPONDENCE
Zhibin Yu
yuzhibin@ouc.edu.cn

RECEIVED 29 December 2022
ACCEPTED 13 April 2023
PUBLISHED 08 May 2023

CITATION
Xin Z, Wang Z, Yu Z and Zheng B (2023) ULL-SLAM: underwater low-light enhancement for the front-end of visual SLAM.
Front. Mar. Sci. 10:1133881.
doi: 10.3389/fmars.2023.1133881

COPYRIGHT
© 2023 Xin, Wang, Yu and Zheng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

ULL-SLAM: underwater low-light enhancement for the front-end of visual SLAM

Zhichao Xin, Zhe Wang, Zhibin Yu* and Bing Zheng

Key Laboratory of Ocean Observation and Information of Hainan Province, Faculty of Information Science and Engineering, Sanya Oceanographic Institution, Ocean University of China, Sanya, Hainan, China

Underwater visual simultaneous localization and mapping (VSLAM), which can provide robot navigation and localization for underwater vehicles, is crucial in underwater exploration. Underwater SLAM is a challenging research topic due to the limitations of underwater vision and error accumulation over long-term operations. When an underwater vehicle goes down, it may inevitably enter a low-light environment. Although artificial light sources could help to some extent, they might also cause non-uniform illumination, which may have an adverse effect on feature point matching. Consequently, the capability of feature point extraction-based visual SLAM systems could only sometimes work. This paper proposes an end-to-end network for SLAM preprocessing in an underwater low-light environment to address this issue. Our model includes a low-light enhancement branch specific with a non-reference loss function, which can achieve low-light image enhancement without requiring paired low-light data. In addition, we design a self-supervised feature point detector and descriptor extraction branch to take advantage of self-supervised learning for feature points and descriptors matching to reduce the re-projection error. Unlike other works, our model does not require pseudo-ground truth. Finally, we design a unique matrix transformation method to improve the feature similarity between two adjacent video frames. Comparative experiments and ablation experiments confirm that the proposed method in this paper could effectively enhance the performance of VSLAM based on feature point extraction in an underwater low-light environment.

KEYWORDS

self-supervised learning, VSLAM, feature point matching, underwater low-light enhancement, end-to-end network

1 Introduction

In recent years, vision-based state estimation algorithms have emerged as a compelling strategy for detecting indoor García et al. (2016), outdoor Mur-Artal and Tardós (2017); Campos et al. (2021), and underwater Rahman et al., 2018; Rahman et al., 2019b environments using monocular, binocular, or multi-cameras. Meanwhile, simultaneous localization and mapping (SLAM) techniques can provide robots with real-time self-localization and constructing a map in an unknown environment, making SLAM vital in path planning, collision avoidance, and self-localization tasks. Specifically, visual SLAM provides an effective solution for many navigation applications Bresson et al. (2017), where it is responsible for detecting unknown environments and assisting in decision-making, planning, and obstacle avoidance. Furthermore, in recent years, the use of autonomous underwater vehicles (AUVs) or remotely operated underwater vehicles (ROVs) for marine species migration Buscher et al. (2020) and coral reef monitoring Hoegh-Guldberg et al. (2007), submarine cable and wreck inspection Carreras et al. (2018), deep-sea exploration Huvenne et al. (2018), and underwater cave exploration have received increasing attention Rahman et al., 2018; Rahman et al., 2019b.

However, unlike the terrestrial environment, the light source conditions are often limited during deep-sea exploration. As a result, underwater vehicles can only perform illumination detection through the airborne light source, which leads to the underexposure of underwater captured images. Furthermore, due to the limited space of the aircraft, the installation distance between the airborne lens and the light source is often too close, which will also lead to uneven exposure of the image or even overexposure. Meanwhile, photos captured underwater suffer from low contrast and color distortion problems due to strong scattering and absorption phenomena. Therefore, providing robust feature points for tracking, matching, and localization for feature point extraction-based visual SLAM systems is complex and challenging. As a result, direct execution of currently available vision-based SLAM often fails to achieve satisfactory and robust results.

To solve the problem of feature point matching, SuperPoint DeTone et al. (2018) expressed keypoints detection as a classification problem and realized the feature point detection method based on deep learning in this way. UnSuperPoint Christiansen et al. (2019) converted the keypoints detection problem into regression, and the detection head outputs the offset ratio of the keypoints in each patch relative to the reference coordinates, thereby improving the effect of feature point detection. Although these methods have achieved fair results in non-underwater general scenes, there is no particular design for underwater low-light scenes.

In recent years, deep learning-based Low-Light-Image-Enhancement(LLIE) has achieved impressive success since the first seminal work Lore et al. (2017). LLNet Lore et al. (2017) employed a variant of stacking sparse denoising autoencoders to brighten and denoise low-light images simultaneously. Zero DCE Li et al. (2021) achieved zero-reference learning through non-reference loss functions and treats light enhancement as an image-specific curve estimation task; it takes low-light images as input and produces high-order curves as output while achieving fast calculations. EnlightenGAN Jiang et al. (2021) adopted an attention-guided U-Net as the generator and used a global-local discriminator to ensure that the augmented results look like authentic typical light images. Although these works can achieve likely results in in-air low-light environments, these existing low-light enhancement networks did not consider the uneven illumination issues during the underwater exploration. Since there is no guarantee to keep the feature points from two adjacent frames consistent, an image-level low-light enhancement model may improve human visual perception but may be useless for feature point matching (Figure 1). Data collection is another underwater challenge. Some existing low-light image enhancement networks Lore et al. (2017); Li et al. (2021); Jiang et al. (2021) need a training data set by fixing multiple cameras to adjust the camera's exposure time or taking images at different times of the day. It would be difficult to take underwater images at different times of the same scene along with an underwater robot.

To address these issues, we propose a front-end network framework for underwater monocular SLAM based on low-light feature point extraction with siamese networks in Figure 2, named ULL-SLAM. Our ULL-SLAM can improve the performance of monocular SLAM in underwater low-light environments. This unsupervised end-to-end network architecture can effectively improve feature-matching performance, thereby obtaining better and more robust SLAM results. Our network can accomplish both low-light image enhancement and feature point extraction, and both are optimized together to enhance the low-light image enhancement network toward favorable feature point extraction and matching. Continuous image frames are input during training, and the network constrains the image enhancement followed by continuous frames to improve the performance of feature point extraction and matching between consecutive frames. Meanwhile, the image enhancement network and the feature point extraction network share the same backbone to improve the inference speed of the model and make the model capable of deployment on embedded devices. Furthermore, we have independently packaged the low-light feature point extraction network of ULL-SLAM, which can help audiences to transplant into any SLAM architecture based on feature point extraction and obtain performance gains. Finally, we evaluate our method on multiple underwater datasets. The proposed method outperforms existing methods in position estimation and system stability. In summary, our main contributions are as follows:

- We propose a mean frame loss and a temporal-spatial consistency loss to improve the ability of feature point extraction among several adjacent frames and keep the enhanced features from the adjacent frames consistent.
- We propose an adaptive low-light enhancement network with an uneven brightness loss, which can adjust the brightness of an image with an arbitrary low-light level.
- We adopt the method of the siamese network to train the network's ability to extract feature points through homography transformation. The siamese network enables interest point scores and positions to be learned automatically.

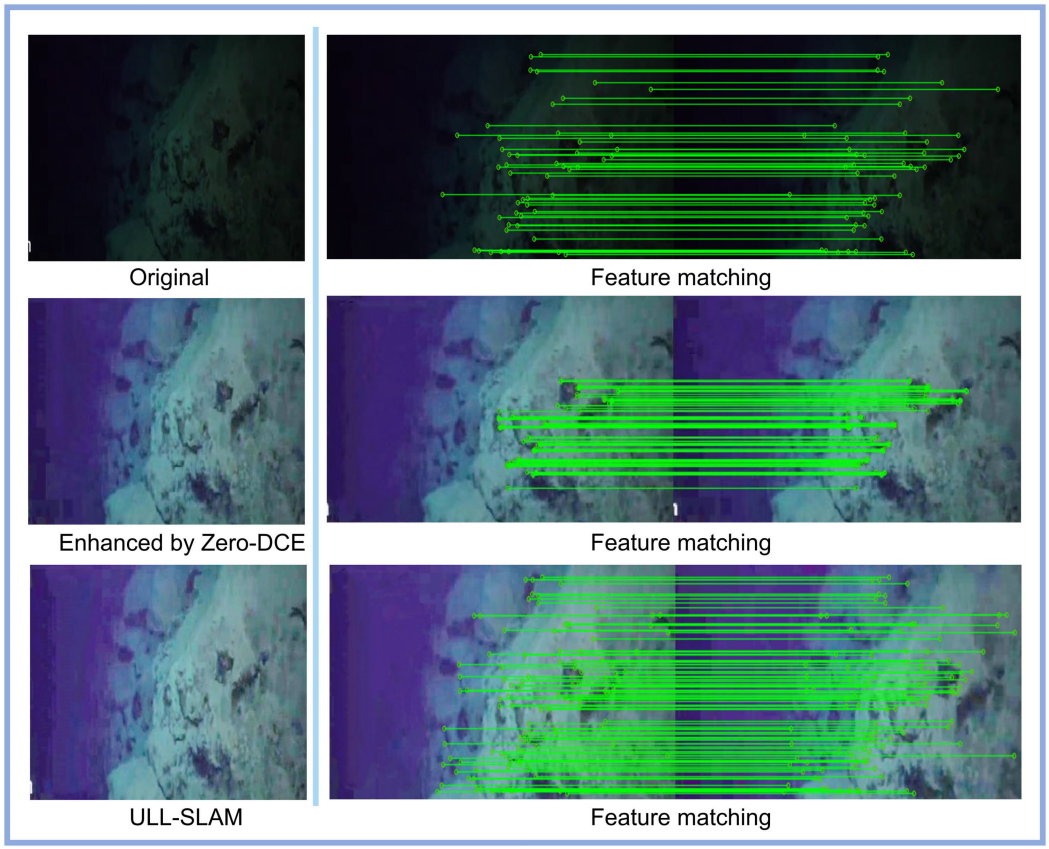


FIGURE 1
An image-level low-light enhancement preprocessing module (e.g., Zero-DCE Li et al. (2021)) can improve human visual perception. However, it is unlikely to improve feature point matching performance between two adjacent frames in an underwater video. The proposed ULL-SLAM, which includes a video-level low-light enhancement module, can effectively extract the feature points between two adjacent frames.

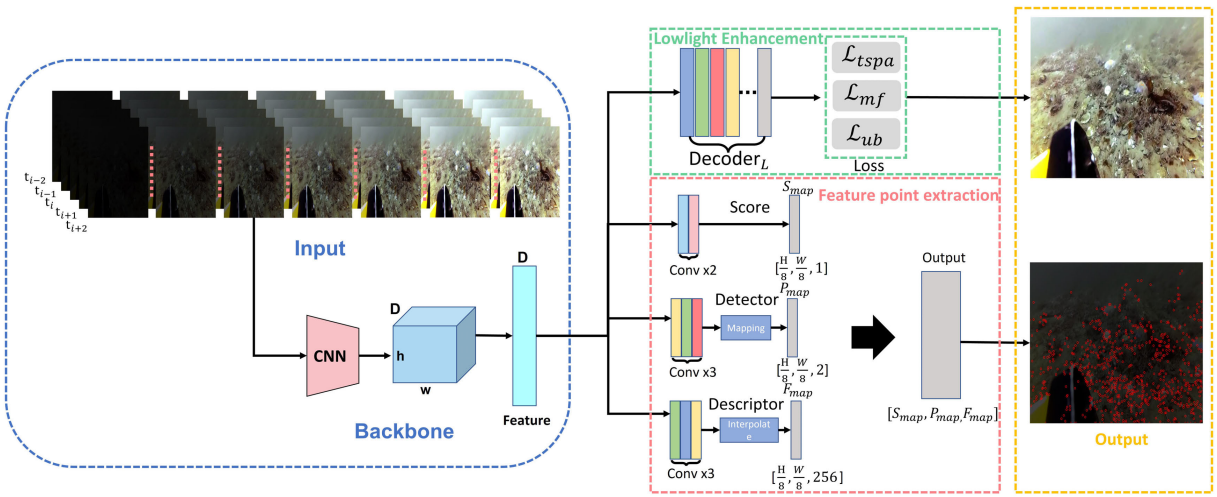


FIGURE 2
The overview framework of the proposed method. The green box is the low-light image enhancement branch, and the red box is the feature extraction branch. The two parts share the same backbone (in the blue box), and the orange box is the output result of the model.

2 Related work

2.1 Low light image enhancement

There are four types of popular low-light image enhancement: 1) supervised learning, 2) reinforcement learning, 3) unsupervised learning, and 4) zero-shot learning. MBLLEN Lv et al. (2018) extracted effective feature representation through a feature extraction module, an enhancement module, and a fusion module, which improves the performance of low-light image enhancement. Ren et al. Ren et al. (2019) designed a more complex end-to-end network, including an encoder-decoder network for image content enhancement and a recursive neural network for image edge enhancement. To reduce the computational burden, Li et al. Li et al. (2018) proposed LightenNet, a lightweight model for low-light image enhancement. LightenNet takes the low-light image as input to estimate its illuminance pattern. It can enhance the image by dividing the input image by the illuminance graph. In the absence of paired training data, Yu et al. Yu et al. (2018) used adversarial reinforcement learning to study the exposure of photos, which they named DeepExposure. First, the input image is segmented into sub-images based on exposure. For each sub-image, local exposures are sequentially learned through a reinforcement learning-based policy network, and the reward evaluation function is approximated by adversarial learning. EnligthenGAN Jiang et al. (2021) is based on an unsupervised learning method and addresses the problem that training a deep model on paired data may lead to overfitting and thus limit the model's generalization ability. Supervised learning, reinforcement learning, and unsupervised learning methods either have limited generalization ability or suffer from unstable training. Zhang et al. Zhang et al. (2019) proposed a zero-shot learning method called ExCNet, which is used for backlit image in painting. It first uses a network to estimate the S-curve that best fits the input image. Once the S-curve is estimated, guided filters separate the input image into a base layer and a detail layer. The estimated S-curve then adjusts the base layer. However, most of these works are image-level models. Applying an image-level model for video preprocessing may cause features to be inconsistent between two adjacent frames. In many low-light underwater cases, the unique illumination from the underwater vehicle could be more likely to cause uneven brightness distribution than in-air cases. Unlike these works, our model includes two loss functions to ensure the enhanced underwater images can practically improve the feature points matching efficiency as well as the VLSAM performance.

2.2 Underwater SLAM

Nowadays, the popular visual SLAM system is normally based on the feature description method Rublee et al. (2011). VINS Qin et al. (2018); Qin and Shen (2018) proposed a general monocular fusion framework containing IMU information. Unlike the non-underwater environment, conventional navigation and positioning communication methods cannot be used typically underwater (such as GPS). Hence, the visual information of the underwater robot itself provides an essential guarantee for robot navigation. In the absence of GPS to generate ground truth for camera poses, a recent work employs Colmap's Schönberger and Frahm (2016); Schönberger et al. (2016) SFM (structure-from-motion, SFM) based method to generate relatively accurate camera trajectories. To evaluate underwater SLAM performance, UW-VO Ferrera et al. (2019) uses the reconstructed trajectories as ground truth trajectory values. Due to the good properties of sound propagation in water, some sonar-based methods Rahman et al., 2018; Rahman et al., 2019a; Rahman et al., 2019b, SVIN Rahman et al. (2018) and SVin2 Rahman et al. (2019b)), incorporate additional sparse depth information from sonar sensors for more accurate position estimation. No matter which kind of feature point-SLAM system is used, the premise of its work is to be able to extract feature points. However, in deep-sea exploration, the feature points cannot be easily extracted due to the low brightness of underwater imaging and insufficient illumination. Besides, sonar sensor-based solutions Rahman et al., 2018; Rahman et al., 2019b) remain expensive, and we aim to propose a general underwater SLAM framework based on purely visual information in deep-sea low-light environments.

3 Methodology

3.1 Overall framework

Feature point extraction and matching play a key role in VSLAM process. Unfortunately, many existing low-light image enhancement works are not designed for continuous frames. An image-level preprocessing may improve human visual perception, but it may be useless for feature point extraction and matching. Moreover, the artificial illumination used for deep-sea exploration may easily cause uneven illumination. The ULL-SLAM front-end feature point extraction network uses a self-supervised siamese network training framework to learn all four tasks simultaneously; the process is shown in Figure 2. The learning tasks of the network are mainly divided into two branches: low-light image enhancement and feature point extraction. The two branches share the same backbone to reduce the model's training time and improve the model's inference speed, thereby ensuring that the model runs on embedded devices in real-time. The low-light image enhancement branch is responsible for enhancing the input original low-light image, and the feature point extraction branch uses the siamese network to predict the two detected feature points of the same input image.

The proposed enhancement network does not directly perform an image-to-image mapping from the low-light image to the enhanced image but rather estimates an enhancement curve from the low-light image to the enhanced image by the network, and applies the estimated enhancement curve to the low-light image to complete the low-light enhancement of the original image. Therefore, in order to make the estimated enhancement curve more accurate, images with different exposure levels of the same image are used when feeding them into the network, which is why the input part of the network frame has 7 images with different exposure levels at the same moment, as shown in Figure 3. In order to ensure the color imbalance that may occur between the front and back frames after underwater continuous



FIGURE 3
The images used for network training increase in brightness from left to right. Images with different exposure levels are used to improve the generalization of the augmentation network and to enhance the detection and matching ability of the feature point detection network.

frame image enhancement (e.g., the image scenes between the front and back frames do not differ much, but the enhancement effect has changed), the images at the five moments of t_i, t_{i-1}, t_{i-2}, t_{i+1}, t_{i+2} at the input end of the network are to ensure that the texture information, color, etc. between the front and back frames of continuous frame image enhancement do not become distorted, and at the same time can complete the Feature point matching, this part is explained in detail in the ablation experiment (Figure 4) of the loss function.

The first step is to perform a spatial transformation (rotation, scaling, tilt, etc.) on the input image through random homography T. Through the siamese network A, output the feature points fraction a, the position a, and the descriptor sub-information a. In the second step, the input image passes through the siamese network B, and then the output result is transformed by the same random homography T to obtain the feature point score B, position B, and descriptor information B. The feature points output by the siamese network A and the siamese network B are spatially aligned, and finally, the distance between the two points is minimized in the loss function to train the network. The feature points are differentiable through the T transformation and the loss function so that each siamese network can be trained and tested end-to-end.

3.2 Backbone

The backbone network takes an input image and generates intermediate feature map representations for each subtask. The first seven convolutional layers of the backbone network are symmetrically connected. Each layer consists of 32 convolution kernels of size 3×3 with a stride of 1 followed by a ReLU activation function. The Tanh activation function follows the last convolution layer. Three max-pooling layers separate the last four pairs of convolutional layers with a stride and kernel size of 2. After each pooling layer, the number of channels in subsequent convolutional layers doubles. The number of channels for 8 convolutional layers is 32-32-64-64-128-128-256-256. Each pooling layer samples twice the height and width of the feature map, while the entire trunk samples are eight times the height and width of the feature map. An entry in the final output corresponds to 8 × 8 regions in the input image. So for an input image of 480 × 640, the network will return (480/8) · (640/8) = 4800 entries Christiansen et al. (2019). Each entry is processed on each subtask in a fully convolutional way to output descriptors, scores, and locations, effectively creating 4800 points of interest Christiansen et al. (2019).

3.3 Low-light image enhancement branch

Underwater robots usually must deal with images with dark light and uneven illumination distribution of continuous video frames in the marine environment, Zero-DCE Li et al. (2021) proposes the idea of brightening the curve as shown in Eq. 1. This function is well designed to solve the problems of the constant brightness value range, monotonically increasing brightening curve, simple curve formula and network differentiability. However, this idea does not consider that the enhanced features between two adjacent frames should be as consistent as possible. Therefore, we draw on this idea to propose a new solution based on the siamese network to deal with the low-light enhancement problem of underwater constant frame images. Specifically as follows:

$$\begin{aligned} LE(I(x);\alpha) &= I(x) + \alpha I(x)(1 - I(x)), \\ LE_n(x) &= LE_{n-1}(x) + \alpha_n LE_{n-1}(x)(1 - LE_{n-1}(x)), \end{aligned} \tag{1}$$

where x is the pixel coordinate; $LE(I(x);\alpha)$ is the augmented image of the input image I(x); $\alpha \in [-1, 1]$ is a trainable curve parameter that adjusts the size of the LE curve. Each pixel is normalized to [0,1], and all operations are performed pixel-wise.

3.3.1 Temporal-spatial consistency loss

Inspired by the spatial consistency loss L_{spa} proposed in Zero-DCE [15], we further consider the temporal relationship between two adjacent frames and propose the temporal-spatial consistency loss L_{tspa} to extend the spatial consistency restriction from the image-level to the video level. Comparing with the L_{spa} defined in Zero-DCE, the proposed L_{tspa} takes into account the spatial consistency between a source image and the homography transformation of its adjacent frame.

Let S denote the siamese networks; I is the raw image. Then we can use the spatial homography transformation matrix T to represent the adjacent frame of the raw image as TI. Let us define $E_a = S(I)$ and $E_b = S(TI)$as the enhanced outputs from the siamese network S, respectively. Then we can define the temporal-spatial consistency loss as follows:

$$\begin{aligned} \mathcal{L}_{tspa} = \frac{1}{K}\sum_{i=1}^{K}(&\left|E_a^i - TE_b^i\right| \\ &+ \sum_{j\in\Omega(i)}(\left|E_a^i - E_a^j\right| + \left|TE_b^i - TE_b^j\right| - \left|TI^i - TI^j\right|))^2, \end{aligned} \tag{2}$$

where K is the number of pixels and i is the traversal of pixels, and $\Omega(i)$ is the 3×3 neighborhood of the i_{th} pixel.

10.3389/fmars.2023.1133881

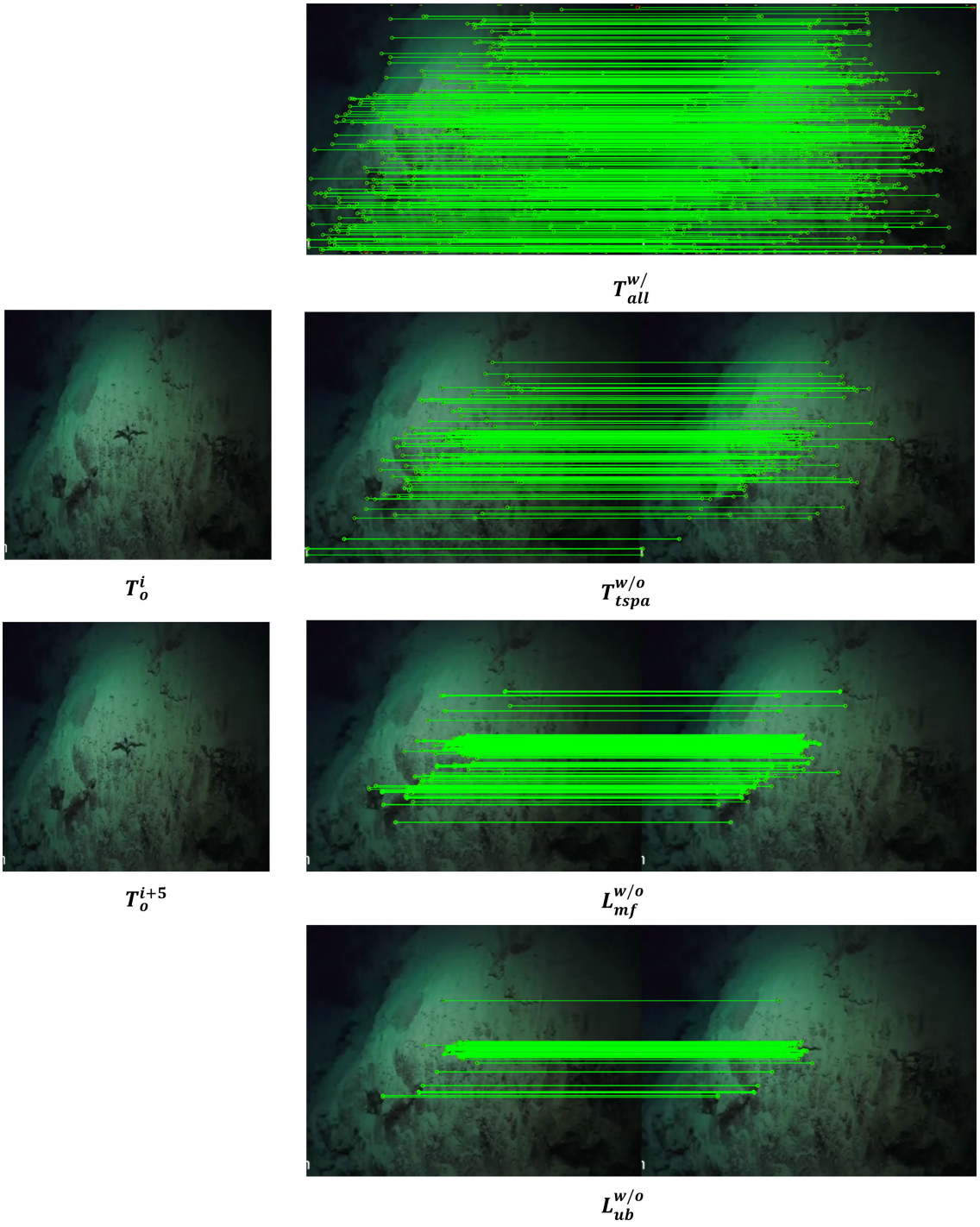


FIGURE 4
The ablation study of various loss functions. $T_{all}^{w/}$ represents the feature point matching result when using all loss functions; $T_{tspa}^{w/o}$ represents the feature point matching result without using L_{tspa}; $T_{mf}^{w/o}$ represents the feature point matching result without using L_{mf}; $T_{ub}^{w/o}$ represents the feature point matching result without using L_{ub}; T_o represents the original low-light image; i represents the image of the current moment; $(i + 5)$ represents the 5_{th} image after the current moment.

3.3.2 Mean frame loss

Our network adopts continuous video frame input for training. We propose a locally constrained loss function that stabilizes transitions between consecutive frames of enhanced images. The scene and pixel differences between consecutive frame images are minimal, and we adopt the idea of local optimization to control the drift between consecutive frame-enhanced images. The specific operations are as follows:

$$\mathcal{L}_{mf} = \frac{1}{M}\sum_{i=1}^{M}\sum_{j=i}^{j+n}(\left|E_{t_{j+1}}^{mean} - E_{t_j}^{mean}\right| + \left|E_{t_j}^{mean} - E_{t_i}^{mean}\right|)^2, \quad (3)$$

Here E_t^{mean} is the average pixel value of the output image of the siamese network at the current moment; M is the total number of images; n is the number of local images selected to participate in the optimization; this value is 4 in actual training.

3.3.3 Uneven brightness loss

In a deep marine environment, artificial illumination is a common light source. However, an artificial light source's power is always insufficient to illuminate the entire area, resulting in uneven illumination. To prevent some places from being too dark and to restrain overexposure, we make the brightness of each pixel closer to a specific intermediate value. We then propose a local uniform brightness loss function, which uses the following error function to express the constraint.

$$\mathcal{L}_{ub} = \sum_{s=1}^{N}|E_s - E_{median}|, E_{median}$$
$$= \begin{cases} \alpha_1 E_{median} & \text{if} \quad E_{median} \le 0.4 \\ \alpha_2 E_{median} & \text{if} \quad E_{median} \ge 0.8 \\ E_{median} & \text{otherwise} \end{cases}, \quad (4)$$

where E_s represents the average value of the local pixel area. During training, the image is divided according to the strategy that the local area is 25 pixels, and N represents the number of local pixel areas. E_{median} describes the median value of the pixel area of the entire image. To prevent the overall brightness of the enhanced image from being low or over exposed, we limit its weight. When the median pixel value is lower than or higher than the set threshold, we use weight parameters ha_1 and α_2 and its compensation to ensure that the generated image will not be overexposed or darkened and to maintain the generated image. The specific values in training are 1.75 and 0.7, respectively.

Meanwhile, to make the enhanced image maintain stable color and smooth illumination, we follow the color constant error loss and smooth illumination loss in Zero-DCE Li et al. (2021), as follows:

3.3.4 Color constancy loss

Zero-DCE Li et al. (2021), proposed color constancy loss corrects for potential color bias in the enhanced image and establishes the relationship between the three adjustment channels. The loss function is defined as follows:

$$\mathcal{L}_{col} = \sum_{\forall(p,q)\in\epsilon}(J_p - J_q)^2, \epsilon \in \{(R,G),(R,B),(G,B)\}, \quad (5)$$

where (p, q) traverses all pairwise combinations of the three RGB color channels, J_p represents the average luminance of color channel p, and (p, q) represents a pair of channels.

3.3.5 Illumination smoothness loss

To maintain the monotonic relationship between adjacent pixels, we follows the illumination smoothness loss defined in Zero-DCE Li et al. (2021). This requirement can be expressed as:

$$\mathcal{L}_{tv_A} = \frac{1}{M}\sum_{n=1}^{N}\sum_{c\in\xi}(|\nabla_x A_n^c| + |\nabla_y A_n^c|)^2, \xi = \{R,G,B\}, \quad (6)$$

N is the number of iterations, and $\nabla_x$and ∇_y are the horizontal and vertical gradient operators, respectively. For images, the horizontal and vertical gradients are the difference between the values of the adjacent pixels to the left and above.

3.4 Feature point extraction branch

To calculate the loss value of the network, we need to establish the relationship between the feature points. The same image passes through the siamese networks A and B and outputs two sets of matrices $A = [S_a, P_a, D_a]$, $B = [S_b, P_b, D_b]$, which respectively represent the feature point scores, feature point positions, and feature point descriptors of the two images output by the network. The position of the feature points detected in image A is transformed into image B through the matrix transformation T, and $\hat{A} = [\hat{S_a}, \hat{P_a}, \hat{D_a}]$ obtained. P_a and $\hat{P_a}$ called feature point pairs, where $\hat{P_a} = TP_a$, the distance between P_a and $\hat{P_a}$ is minimized. The smaller the distance between the two, the better the ability of the extraction network to extract feature points. However, not all $\hat{P_a}$ are involved in the calculation. This is because the siamese network is uncertain about the output of the same image after matrix transformation, and there will be occasional weak feature points. Therefore, according to the experience of reprojection error in SLAM, we define that after the homography matrix transformation T DeTone et al. (2018); Christiansen et al. (2019). The distance between the feature points and the position is within the neighborhood of 3×3 pixels, which means that the detected feature points are the same point in the input image. We sent the positions of such feature points to the loss function for calculation. The operation can effectively improve the stability and repeatability of network detection feature points. The Loss function is handled in the same way as UnSuperpoint Christiansen et al. (2019). We use $\mathcal{L}_{unsuperpoint}$ to describe it here.

Total loss.

$$L_{total} = \mathcal{L}_{tspa} + \mathcal{L}_{mf} + \mathcal{L}_{ub} + \mathcal{L}_{col} + W_{tv_A}\mathcal{L}_{tv_A} + \mathcal{L}_{unsuperpoint} \quad (7)$$

where weight W_{tv_A} is used to balance scales with different losses, which is a direct reference to the weight setting in Zero-DCE. The loss function L_{total} sums up the loss function of the image enhancement branch and the loss function of the feature point extraction branch. By minimizing the loss function L_{total}, the effect of the enhanced image can be achieved to generate in the direction favorable to feature point extraction, so that the network has the ability of feature point extraction in the underwater low-light environment.

4 Experiments

In this section, we compare the advantages of ULL-SLAM with the widespread feature point extraction based SLAM operating in a marine low-light environment. We choose ORB-SLAM2 Mur-Artal

and Tardós (2017), which has stable performance in the underwater test in our laboratory, as our baseline. ORB SLAM2 is also a visual SLAM framework that can be used for monocular, stereo, and RGB-D cameras based on the extraction of feature points (ORB). A new system —ULL-SLAM is constructed by replacing its physical sign point extraction module with our underwater low-light feature point extraction network. We also compared it to the original ORB-SLAM2 Mur-Artal and Tardós (2017), ORB-SLAM3 Campos et al. (2021), and Dual-SLAM Huang et al. (2020).

- Dual-SLAM Huang et al. (2020) extends ORB-SLAM2, saves the current mapping, and activates two new SLAM threads. One handles the incoming frame to create a new map, and the other targets link the new and old maps.
- ORB-SLAM3 Campos et al. (2021) Visual, visual-inertial, and multi-map SLAM using monocular, stereo, and RGB-D cameras, achieving state-of-the-art performance.

Since we adopt a deep learning-based method to extract feature points, we test the model's running speed (frame-per-second, FPS) on Jetson AGX Xavier, which is also widely equipped on ROV and AUV. Our ULL-SLAM can reach a speed of 40.6 FPS.

4.1 Implementation details and evaluation metrics

4.1.1 Dataset

4.1.1.1 Training dataset

The *URPC* dataset Liu et al. (2021) contains contains monocular video sequences collected by the ROV on a real aquaculture farm nearby Zhangzi Island, China. The ROV can travel in water depths of about 5 meters. The ROV captured a total of 190 seconds of video sequences at a 24Hz acquisition frequency. We obtain a total of 4,538 frames from the video. The collected video sequence scene changes significantly, the light is sufficient, but the water quality is cloudy. In order to ensure that the feature point extraction branch can extract more feature points, we add the image after image sharpening in the laboratory's previous work. The fusion of these two kinds of data not only ensures that the feature point extraction network can extract more feature points but also ensures the generalization ability and robustness of the model. The low-light image enhancement model based on zero-order learning cannot be trained typically with simple underwater images. However, acquiring underwater low-light data sets is difficult and expensive. Therefore, we adopt the idea of style transfer to transform the brightness of datasets and finally form images with different colors and brightness for training. Considering that there are no meaningful objects in the first 2000 consecutive images in the original sequence, we delete them and select only the last 2538 images, respectively, for brightness conversion. Among them, we used 1250 images for testing. In the training process, we select the open-source offline SFM Schönberger and Frahm (2016); Schönberger et al. (2016) library to generate a camera attitude track from 1250 continuous frame images to evaluate underwater SLAM performance.

4.1.1.2 Test datasets

The training data set *URPC* is an artificially generated low-light image. To test the performance of ULL-SLAM in a natural underwater environment, we select five video clips of natural underwater low-light scenes from the videos provided by Schmidt Ocean Alalykina and Polyakova (2022). These video clips are captured with an underwater vehicle to a depth of 400-500 meters in the Pacific Ocean. Each video clip is 2150, 3500, 4600, 5200, and 6000 frames, respectively. The rotation and ambiguity of the image in each piece of data are different. We generate the camera pose using SFM Schönberger and Frahm (2016); Schönberger et al. (2016). We also use SFM to provide ground truth to test the performance of the ULL-SLAM system in a natural underwater low-light environment.

4.1.2 Evaluation metric for SLAM

To measure SLAM performance, we choose 1) absolute trajectory error (ATE), 2) root mean square error (RMSE), and 3) initialization performance for evaluation. ATE directly computes the difference between the ground-truth trajectory of the camera pose and the SLAM-estimated trajectory. RMSE can describe the rotational and translational errors of two trajectories. The smaller the RMSE, the better the system trajectory fit. The initialization performance indicates the number of frames to perform underwater SLAM initialization. The lower the initialization frame, the better the SLAM performs and the more stable and continuous the output. We repeated ten underwater SLAM experiments to get the best results for all methods.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Y_i - f(x_i))^2} \tag{8}$$

where $f(x_i)$ represents the system's predicted trajectory, and Y_i represents the Groundtruth of the trajectory.

4.2 Low light enhanced visualization result

We verify the effect of the proposed loss function in this section and visualize the effect of each function separately by conducting ablation experiments during training. It is worth noting that the loss function we designed for continuous frames (Eq. 3) and overexposure (Eq. 4) mainly enables the network to have a good feature point extraction effect in the underwater low-light environment. The two networks are optimized end-to-end together rather than proposing a low-light image enhancement model. Therefore, we do not compare the performance of other low-light enhancement models on terrene in the same underwater scene. Figures 5, 6 show the comparison of the training dataset image and the real underwater test dataset image before and after the low-light enhancement network, respectively. Figure 7 verifies the ablation experiment of our proposed loss function on the low-light image enhancement effect. It should be noted that the ultimate purpose of our network is to focus on the effect of the network in feature point extraction, so Figure 4 shows the effect of our proposed loss function on feature point extraction.

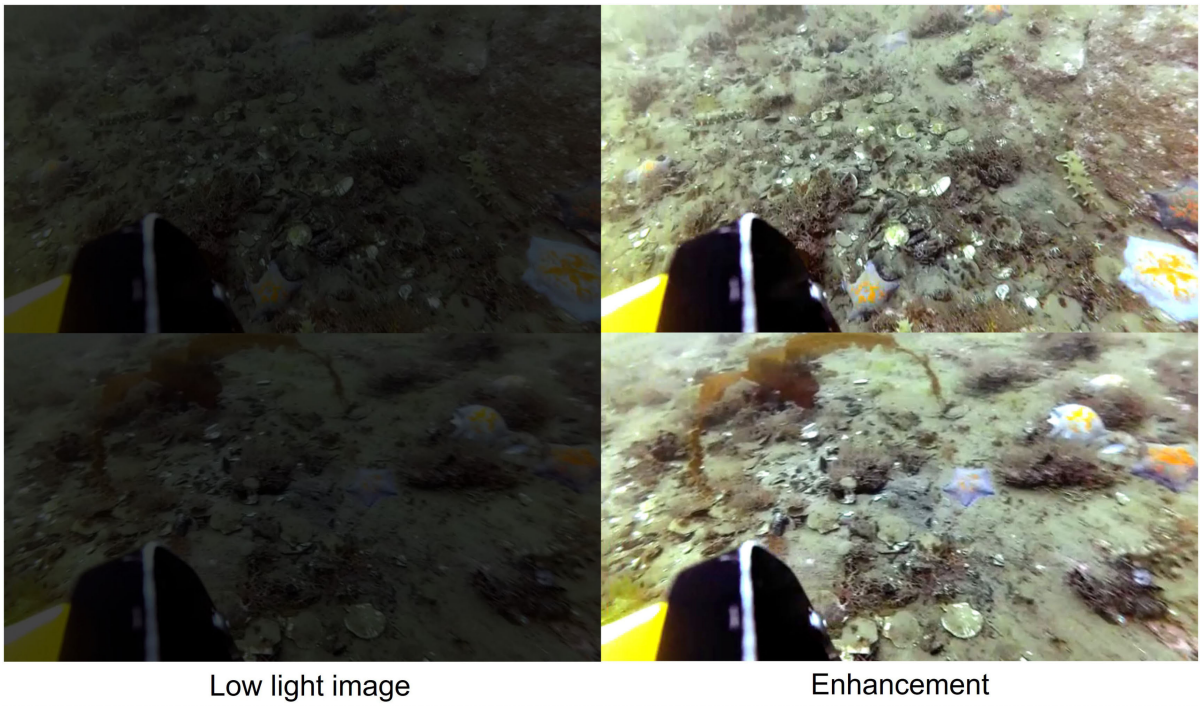


FIGURE 5
Comparison of low-light images before and after enhancement on URPC-dark dataset.

4.3 Feature point matching performance

To further reveal the superiority of the feature point extraction effect in ULL-SLAM compared with other methods, we show the matching pairs with ORB Rublee et al. (2011), SIFT Lowe (2004), and SURF Bay et al. (2008) under two consecutive frames in Figures 8, 9. We obtain ground-truth values from motion using a structure-of-motion-based COLMAP Schönberger and Frahm (2016); Schönberger et al. (2016) method. We conduct experiments using 2150 consecutive frames of underwater images with an image size of 640x480 and pre-calibrated in-camera references. Only matching pairs in the 3×3 pixel region are considered correct matched pairs.

To verify that the feature points detected by our system are valid interior points, we conduct the feature point matching test through the reprojection error of every 20 frames of images. Specifically, the feature points extracted from the current frame are reprojected onto the previous 20th frame image to compare the errors between the feature points. Then we select a 3×3 pixel region. When the error between the feature points is less than 3, the feature point is marked as number 0 and the inner point; then, the others are marked as the mismatched outer points and number 1. Finally, the feature-matching error rate of our proposed method is 0.9%, the error rate of ORB method is 6.7, the error rate of SIFT method is 5.1, and the error rate of SURF method is 3.5. The formula is as follows:

$$Pix = \left| p - KHp_{w_{interval=20}} \right| \begin{cases} 1 & otherwise \\ 0 & pix < 3 \end{cases} \tag{9}$$

where p represents the coordinates of the feature points of the current frame, K represents camera parameters, H represents the transformation matrix, and $p_{w_{interval=20}}$ represents the coordinates of the image feature point at the 20th frame interval from the current frame.

$$Error = \frac{1}{N} \sum_{i=1,10,20\ldots}^{N} \left| p_i - KHp_{wi} \right| \tag{10}$$

where Nrepresents the number of image pairs involved in reprojection.

To verify the ability of the system to extract feature points in a natural low-light underwater environment, we conducted a feature point detection test in the test dataset. According to the constraints of state estimation, the SLAM system outputs accurate positional estimation data only when a sufficient number of interior points are matched, and when the number of interior points is too small, it will cause the system to fail to complete the positional estimation. Therefore, we construct a test image pair at intervals of 20 and 30 frames for the test set video clips and perform feature point detection and matching tests in different feature point detectors. When the number of feature points detected between the two frames of the test image pair is greater than 50, we record the correct samples and calculate the proportion of the accurate sample numbers in all test pairs of the video clip. When the system is able to detect enough feature points at 20 or 30 frames between keyframes, it proves that the feature point matching capability of the network is good enough. The performance of the system is demonstrated by verifying the matching ability of the proposed network feature points. In this way, we use this method to compare the ability of

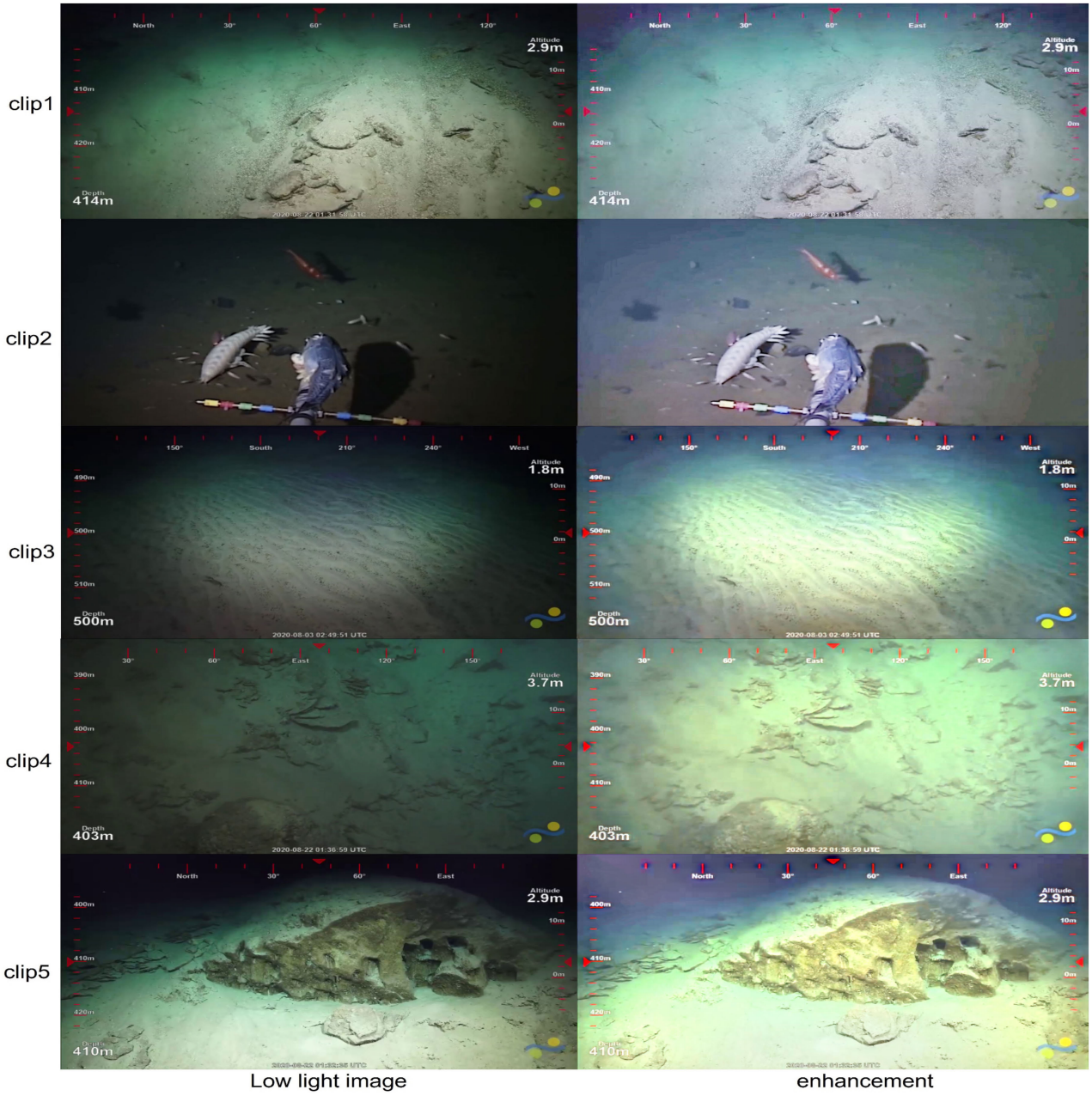


FIGURE 6
Comparison of low-light images before and after enhancement on real underwater dataset provided by Schmidt Ocean Alalykina and Polyakova (2022).

network feature point detection. The test results are shown in Tables 1, 2.

Similarly, we propose a SLAM system and pay more attention to the effectiveness of the extracted feature points on the SLAM system. There is no direct proportion between the number of matching feature points and the performance of SLAM. Therefore, in the comparison experiment, we only select the feature point extraction methods commonly used in the current SLAM system, such as (ORB). Other feature point extraction networks based on deep learning only focus on feature point extraction and have yet to be transplanted into the SLAM system, so we did not compare them.

4.4 Underwater SLAM results

We aim to validate the proposed network model in low-light feature points Extraction SLAM and the system's effectiveness. We adopt the ORB-SLAM2 of the stability of the effect in the early stage of the laboratory experiment as the basic SLAM framework. Our model replaces the ORB feature point extraction network in the original system, keeping the back-end optimization architecture with the original method unchanged, forming a new SLAM system – ULL-SLAM. Our model replaces the ORB feature point extraction network in the original system, keeping the back-end optimization architecture with the original method

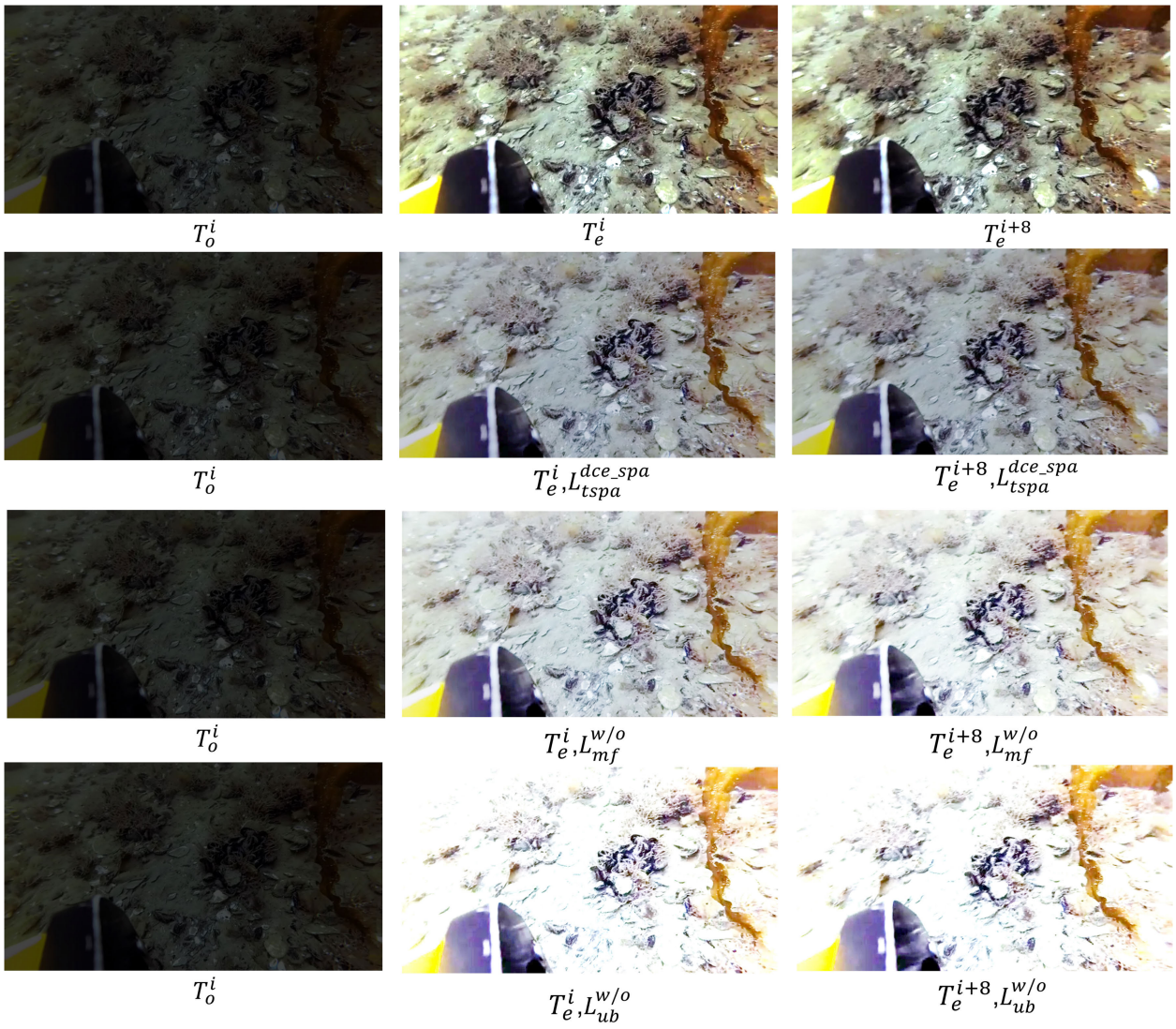


FIGURE 7
The ablation study of various loss functions. The first row of images represents the normal network output, T_o represents the original low-light image, i represents the image of the current moment. We select the i_th Frame and eighth (i+8) frames after the current moment to verify the effects of different functions, w/o represents the other functions unchanged, and the network output image after removing this function. When the loss function $\mathcal{L}_{ub}$ is removed, we can find overexposure occurs in the image after enhancement. When the loss function $\mathcal{L}_{mf}$ is removed, it can be seen that the image scene does not change significantly at the interval of 8 frames, but the enhancement effect has changed significantly.

unchanged, forming a new SLAM system – ULL-SLAM. It conducts comparative experiments with the original ORB-SLAM and the currently popular Dual-SLAM and ORB-SLAM3 on the URPC-dark dataset. The quantification results are shown in Table 3. From the results, it can be found that the quantization error of ULL-SLAM is significantly smaller than the other three, and the minor quantization error can make the estimated camera pose trajectory more stable, thereby considerably improving the initial performance. An excellent low-light feature point extraction network can make feature matching more reliable so that ULL-SLAM can obtain a more stable and accurate output.

In the five real underwater low-light scenes, we use Zero-DCE as the pre-processing of underwater low-light image enhancement tool. Then, we feed the enhanced images into ORB-SLAM2 for testing. As shown in Table 4, ORB-SLAM2 did not improve all the data sets. The results indicate that an image-level low-light enhancement network can hardly improve the feature point matching and SLAM's performance.

We compared the performance of ULL-SLAM and the other three SLAM systems in five real underwater low-light video clips on the test set provided by Schmidt Ocean. The visualization results of the test tracks of these four SLAM systems are shown in Figure 10. We can find that the SLAM trajectory obtained with ULL-SLAM is closest to the ground truth. Meanwhile, Table 5 shows the quantization error data of the four systems in the five video clips. The two experimental results confirm that the ULL-SLAM system

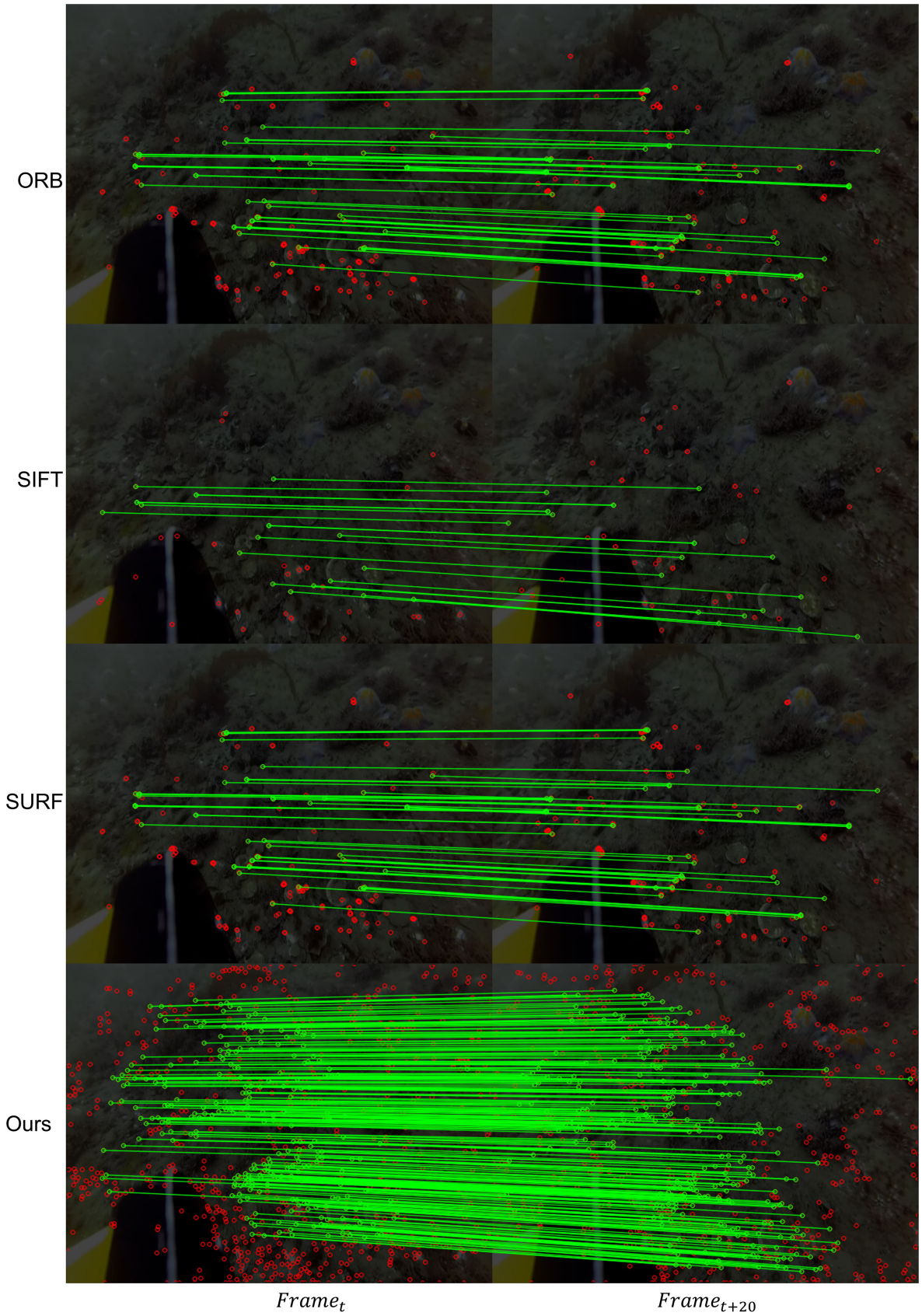


FIGURE 8
Comparison of extraction methods of different feature points. The image on the left is the current frame image, and the image on the right is the 20_{th} frame image behind the current frame image.

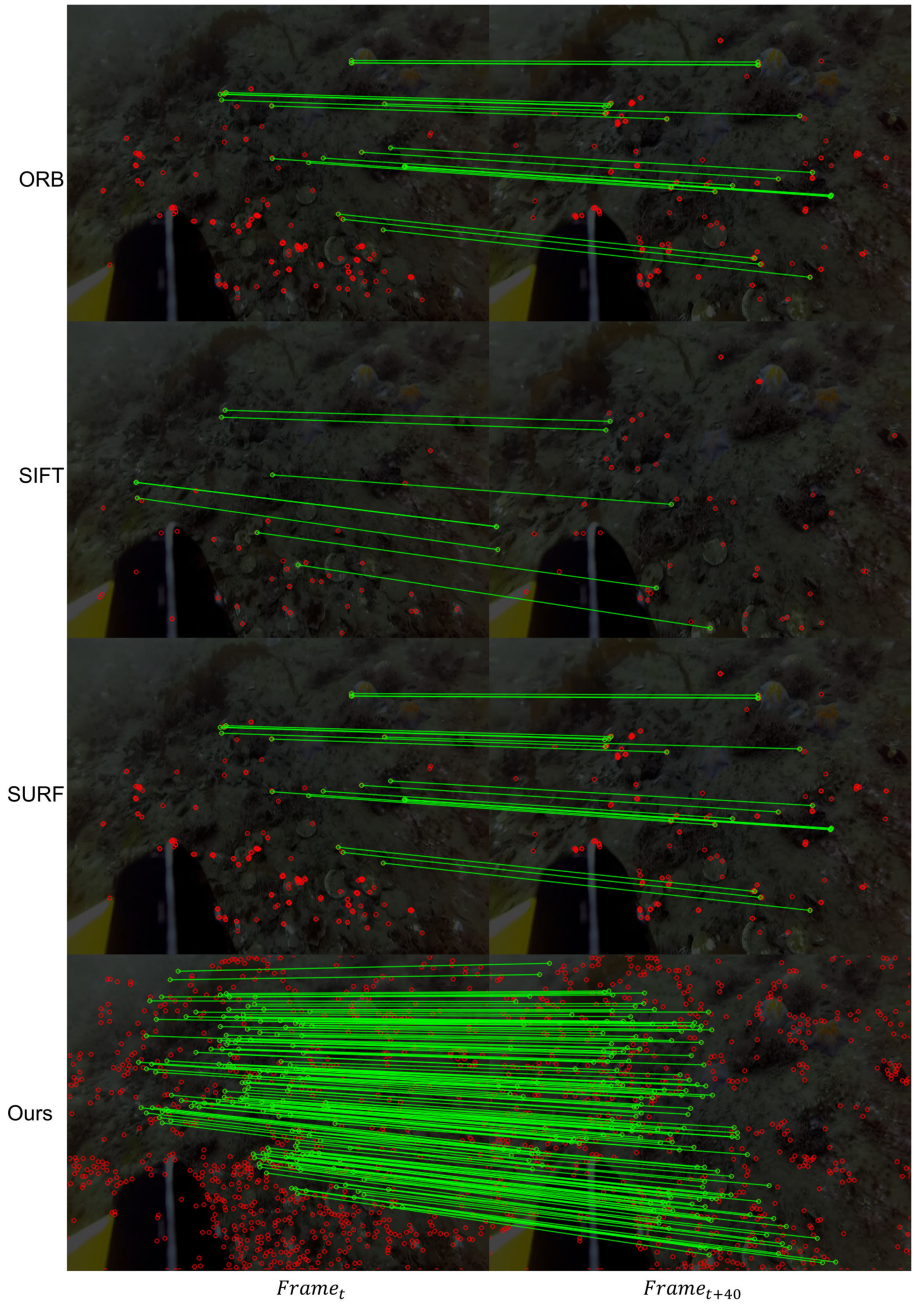


FIGURE 9
Comparison of extraction methods of different feature points. The image on the left is the current frame image, and the image on the right is the 40_{th} frame image behind the current frame image.

TABLE 1 Feature point detection effect of different feature point detectors in a real underwater environment.

Video clips	Method	> 50 ↑	Accuracy rate ↑
seg1	ORB	814	0.757
	SIFT	822	0.765
	SURF	869	0.808
	ULL-SLAM	912	0.848
seg2	ORB	1511	0.863

(Continued)

TABLE 1 Continued

Video clips	Method	> 50 ↑	Accuracy rate ↑
	SIFT	1505	0.860
	SURF	1542	0.881
	ULL-SLAM	1607	0.918
seg3	ORB	1467	0.638
	SIFT	1432	0.623
	SURF	1502	0.653
	ULL-SLAM	1624	0.706
seg4	ORB	2412	0.928
	SIFT	2391	0.919
	SURF	2421	0.931
	ULL-SLAM	2501	0.962
seg5	ORB	2288	0.762
	SIFT	2301	0.767
	SURF	2327	0.776
	ULL-SLAM	2433	0.811

Spaced 20 frame pairs of images.

TABLE 2 Feature point detection effect of different feature point detectors on the dataset provided by Schmidt Ocean.

Video clips	Method	>50 ↑	Accuracy rate ↑
seg1	ORB	772	0.718
	SIFT	784	0.729
	SURF	816	0.759
	ULL-SLAM	839	0.784
seg2	ORB	1449	0.828
	SIFT	1436	0.820
	SURF	1467	0.838
	ULL-SLAM	1521	0.869
seg3	ORB	1349	0.586
	SIFT	1327	0.577
	SURF	1413	0.614
	ULL-SLAM	1575	0.685
seg4	ORB	2305	0.886
	SIFT	2277	0.876
	SURF	2334	0.898
	ULL-SLAM	2419	0.930
seg5	ORB	2196	0.732
	SIFT	2225	0.741
	SURF	2276	0.759
	ULL-SLAM	2349	0.783

Spaced 30 frame pairs of images.

TABLE 3 Quantization errors of different SLAM systems on URPC-dark test dataset.

Method	ATE ↓	RMSE ↓	Initialization ↓
ORB-SLAM2	1.711	1.764	32
Dual-SLAM	1.693	1.722	23
ORB-SLAM3	1.686	1.707	26
ULL-SLAM	**1.292**	**1.316**	**3**

Bold text indicates that it performs best under the same evaluation index. For example, the bold text under the column ATE (absolute trajectory error) indicates that ULL-SLAM obtained the best performance in the ATE evaluation index, with a quantitative value of 1.292.The same goes for other bold letters.

can achieve the expected effect in the authentic underwater low-light environment, which verifies that our proposed scheme can be well applied in the underwater low-light environment.

4.5 Limitations and future work

The low-light image enhancement branch and feature point extraction branch share the same network and are optimized end-to-end, which can complement each other for mutual benefit and improve operational efficiency simultaneously. However, we did not consider a de-scattering module to remove forward and backward scattering noise for underwater exploration. We aim to build a universal underwater visual SLAM framework that is robust to various underwater conditions. We leave it as our subsequent work.

5 Conclusion

In this paper, we propose an underwater low-light feature point extraction network based on siamese networks and integrate it into the back-end framework of the SLAM system to form a new SLAM system—ULL-SLAM. To improve the

TABLE 4 Comparative experiments on the dataset provided by Schmidt Ocean.

Video clips	Method	ATE ↓	RMSE ↓	Initialization ↓
seg1	ORB-SLAM2	0.823	0.847	23
	Zero–DCE + ORB-SLAM2	0.809	0.821	19
	EnlightenGAN + ORB-SLAM2	0.779	0.792	16
	MBLLEN + ORB-SLAM2	0.807	0.822	20
seg2	ORB-SLAM2	0.611	0.643	10
	Zero–DCE + ORB-SLAM2	0.644	0.671	15
	EnlightenGAN + ORB-SLAM2	0.581	0.601	13
	MBLLEN + ORB-SLAM2	0.567	0.583	16
seg3	ORB-SLAM2	2.892	2.934	37
	Zero–DCE + ORB-SLAM2	2.979	3.073	40
	EnlightenGAN + ORB-SLAM2	3.017	3.225	47
	MBLLEN + ORB-SLAM2	2.709	2.811	38
seg4	ORB-SLAM2	0.391	0.404	4
	Zero–DCE + ORB-SLAM2	0.369	0.392	3
	EnlightenGAN + ORB-SLAM2	0.322	0.359	5
	MBLLEN + ORB-SLAM2	0.431	0.457	8
seg5	ORB-SLAM2	0.802	0.816	19
	Zero–DCE + ORB-SLAM2	0.792	0.801	15
	EnlightenGAN + ORB-SLAM2	0.676	0.692	14
	MBLLEN + ORB-SLAM2	0.845	0.861	20

Zero-DCE, EnlightenGAN and MBLLEN are used for preprocessing low-light images, feeding the enhanced image into the ORB-SLAM2.

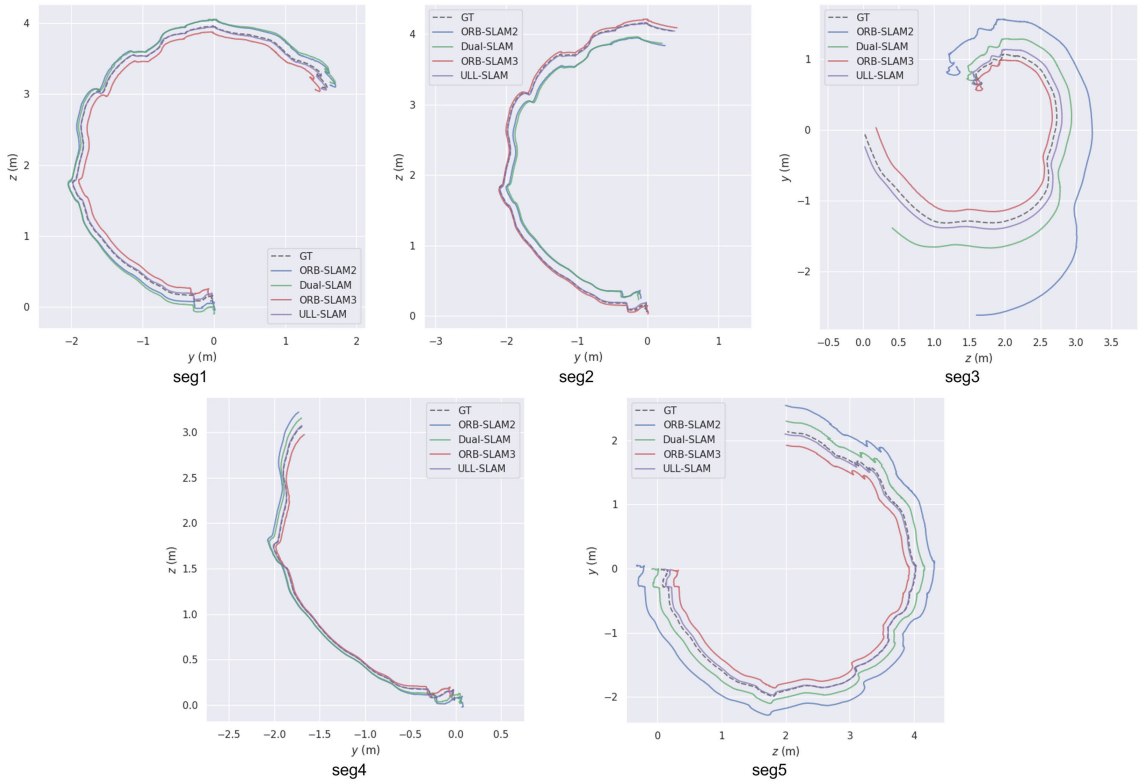


FIGURE 10
The performance of different SLAM systems on five real-life underwater low-light video segments.

TABLE 5 ULL-SLAM and three other SLAM systems performed in five segments of real underwater low-light environments in the test dataset provided by Schmidt Ocean.

Video clips	Method	ATE ↓	RMSE ↓	Initialization ↓
seg1	ORB-SLAM2	0.823	0.847	23
	Dual-SLAM	0.809	0.830	16
	ORB-SLAM3	0.786	0.802	18
	ULL-SLAM	0.592	0.624	4
seg2	ORB-SLAM2	0.611	0.643	10
	Dual-SLAM	0.595	0.619	6
	ORB-SLAM3	0.583	0.607	8
	ULL-SLAM	0.490	0.523	1
seg3	ORB-SLAM2	2.892	2.934	37
	Dual-SLAM	2.786	2.899	32
	ORB-SLAM3	2.795	2.836	26
	ULL-SLAM	2.601	2.625	9
seg4	ORB-SLAM2	0.391	0.404	4
	Dual-SLAM	0.387	0.395	3
	ORB-SLAM3	0.374	0.389	3
	ULL-SLAM	0.319	0.331	1

(Continued)

TABLE 5 Continued

Video clips	Method	ATE ↓	RMSE ↓	Initialization ↓
seg5	ORB-SLAM2	0.802	0.816	19
	Dual-SLAM	0.786	0.803	15
	ORB-SLAM3	0.778	0.791	14
	ULL-SLAM	0.589	0.606	2

Under the evaluation index of SLAM system, ULL-SLAM can achieve better results in real underwater low-light environments compared with other systems.

inference speed of the model to achieve real-time performance, we designed the low-light image enhancement branch and the feature point extraction branch with the same backbone. Moreover, the loss functions of the two branches are optimized together so that the low-light image enhancement branch can generate feature images beneficial to feature point detection. Thus the two are mutually beneficial. At the same time, the proposed network can be flexibly transplanted to the popular SLAM system based on feature point extraction to improve the system's performance. Experimental results show that this method makes the output trajectory of SLAM more continuous and stable in an underwater low-light environment and carries out more accurate state estimation.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

ZX is responsible for the design of the experiment and the implementation of the algorithm, ZW is responsible for drawing, and ZY and BZ are responsible for the idea and editing of the paper. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the Hainan Province Science and Technology Special Fund of China (Grant No. ZDYF2022SHFZ318), the Project of Sanya Yazhou Bay Science and Technology City (Grant No. SCKJ-JYRC-2022-102) and the National Natural Science Foundation of China (Grant No. 62171419).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alalykina, I. L., and Polyakova, N. E. (2022). New species of ophryotrocha (annelida: dorvilleidae) associated with deep-sea reducing habitats in the bering sea, northwest pacific. *Deep Sea Res. Part II: Top. Stud. Oceanog.* (Elsevier) 206, 105217.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Comput. Vision imag. understanding* 110, 346–359. doi: 10.1016/j.cviu.2007.09.014

Bresson, G., Alsayed, Z., Yu, L., and Glaser, S. (2017). Simultaneous localization and mapping: a survey of current trends in autonomous driving. *IEEE Trans. Intelligent Vehicles* 2, 194–220. doi: 10.1109/TIV.2017.2749181

Buscher, E., Mathews, D. L., Bryce, C., Bryce, K., Joseph, D., and Ban, N. C. (2020). Applying a low cost, mini remotely operated vehicle (rov) to assess an ecological baseline of an indigenous seascape in canada. *Front. Mar. Sci.* 7, 669. doi: 10.3389/fmars.2020.00669

Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., and Tardós, J. D. (2021). Orb-slam3: an accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* (IEEE). doi: 10.1109/TRO.2021.3075644

Carreras, M., Hernández, J. D., Vidal, E., Palomeras, N., Ribas, D., and Ridao, P. (2018). Sparus ii auv--a hovering vehicle for seabed inspection. *IEEE J. Oceanic Eng.* 43, 344–355. doi: 10.1109/JOE.2018.2792278

Christiansen, P. H., Kragh, M. F., Brodskiy, Y., and Karstoft, H. (2019). Unsuperpoint: end-to-end unsupervised interest point detector and descriptor. *arXiv preprint arXiv:1907.04011*.

DeTone, D., Malisiewicz, T., and Rabinovich, A. (2018)Superpoint: self-supervised interest point detection and description (Accessed Proceedings of the IEEE conference on computer vision and pattern recognition workshops).

Ferrera, M., Moras, J., Trouvé-Peloux, P., and Creuze, V. (2019). Real-time monocular visual odometry for turbid and dynamic underwater environments. *Sensors* 19, 687. doi: 10.3390/s19030687

García, S., López, M. E., Barea, R., Bergasa, L. M., Gómez, A., and Molinos, E. J. (2016)Indoor slam for micro aerial vehicles control using monocular camera and sensor fusion (IEEE) (Accessed 2016 international conference on autonomous robot systems and competitions (ICARSC)).

Hoegh-Guldberg, O., Mumby, P. J., Hooten, A. J., Steneck, R. S., Greenfield, P., Gomez, E., et al. (2007). Coral reefs under rapid climate change and ocean acidification. *Science* 318, 1737–1742. doi: 10.1126/science.1152509

Huang, H., Lin, W.-Y., Liu, S., Zhang, D., and Yeung, S.-K. (2020)Dual-slam: a framework for robust single camera navigation (IEEE) (Accessed 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)).

Huvenne, V. A., Robert, K., Marsh, L., Iacono, C. L., Le Bas, T., and Wynn, R. B. (2018). "Rovs and auvs," in *Submarine geomorphology* (Springer), 93–108.

Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., et al. (2021). Enlightengan: deep light enhancement without paired supervision. *IEEE Trans. Image Process.* 30, 2340–2349. doi: 10.1109/TIP.2021.3051462

Li, C., Guo, C., and Loy, C. C. (2021). Learning to enhance low-light image *via* zero-reference deep curve estimation. *arXiv preprint arXiv:2103.00860*.

Li, C., Guo, J., Porikli, F., and Pang, Y. (2018). Lightennet: a convolutional neural network for weakly illuminated image enhancement. *Pattern recognit. Lett.* 104, 15–22. doi: 10.1016/j.patrec.2018.01.010

Liu, C., Li, H., Wang, S., Zhu, M., Wang, D., Fan, X., et al. (2021)A dataset and benchmark of underwater object detection for robot picking (IEEE) (Accessed 2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)).

Lore, K. G., Akintayo, A., and Sarkar, S. (2017). Llnet: a deep autoencoder approach to natural low-light image enhancement. *Pattern Recognit.* 61, 650–662. doi: 10.1016/j.patcog.2016.06.008

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60, 91–110. doi: 10.1023/B:VISI.0000029664.99615.94

Lv, F., Lu, F., Wu, J., and Lim, C. (2018). "Mbllen: low-light image/video enhancement using cnns," in *BMVC*, vol. 220. , 4.

Mur-Artal, R., and Tardós, J. D. (2017). Orb-slam2: an open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* 33, 1255–1262. doi: 10.1109/TRO.2017.2705103

Qin, T., Li, P., and Shen, S. (2018). Vins-mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* 34, 1004–1020. doi: 10.1109/TRO.2018.2853729

Qin, T., and Shen, S. (2018)Online temporal calibration for monocular visual-inertial systems (IEEE) (Accessed 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)).

Rahman, S., Li, A. Q., and Rekleitis, I. (2018)Sonar visual inertial slam of underwater structures (IEEE) (Accessed 2018 IEEE International Conference on Robotics and Automation (ICRA)).

Rahman, S., Li, A. Q., and Rekleitis, I. (2019a)Contour based reconstruction of underwater structures using sonar, visual, inertial, and depth sensor (IEEE) (Accessed 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)).

Rahman, S., Li, A. Q., and Rekleitis, I. (2019b)Svin2: an underwater slam system using sonar, visual, inertial, and depth sensor (IEEE) (Accessed 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)).

Ren, W., Liu, S., Ma, L., Xu, Q., Xu, X., Cao, X., et al. (2019). Low-light image enhancement *via* a deep hybrid network. *IEEE Trans. Image Process.* 28, 4364–4375. doi: 10.1109/TIP.2019.2910412

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011)Orb: an efficient alternative to sift or surf (Ieee) (Accessed 2011 International conference on computer vision).

Schönberger, J. L., and Frahm, J.-M. (2016)Structure-from-motion revisited (Accessed Proceedings of the IEEE conference on computer vision and pattern recognition).

Schönberger, J. L., Zheng, E., Frahm, J.-M., and Pollefeys, M. (2016)Pixelwise view selection for unstructured multi-view stereo (Springer) (Accessed European conference on computer vision).

Yu, R., Liu, W., Zhang, Y., Qu, Z., Zhao, D., and Zhang, B. (2018). Deepexposure: learning to expose photos with asynchronously reinforced adversarial learning. *Adv. Neural Inf. Process. Syst.* 31.

Zhang, L., Zhang, L., Liu, X., Shen, Y., Zhang, S., and Zhao, S. (2019) Zero-shot restoration of back-lit images using deep internal learning (Accessed Proceedings of the 27th ACM International Conference on Multimedia).

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 10 May 2023
DOI 10.3389/fmars.2023.1151817

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Luis Gomez,
University of Las Palmas de Gran Canaria, Spain
Xiaoling Zhang,
University of Electronic Science and Technology of China, China

*CORRESPONDENCE
Dazhi Gao
dzgao@ouc.edu.cn

RECEIVED 26 January 2023
ACCEPTED 13 April 2023
PUBLISHED 10 May 2023

CITATION
Guo D, Gao D, Chen Z, Li Y, Zhao X, Song W and Li X (2023) Classification of inbound and outbound ships using convolutional neural networks. *Front. Mar. Sci.* 10:1151817. doi: 10.3389/fmars.2023.1151817

COPYRIGHT
© 2023 Guo, Gao, Chen, Li, Zhao, Song and Li. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Classification of inbound and outbound ships using convolutional neural networks

Doudou Guo[1], Dazhi Gao[1]*, Zhuo Chen[1], Yuzheng Li[1], Xiaojing Zhao[1], Wenhua Song[2] and Xiaolei Li[1]

[1]College of Marine Technology, Ocean University of China, Qingdao, China, [2]College of Physics and Optoelectronic Engineering, Ocean University of China, Qingdao, China

In general, a single scalar hydrophone cannot determine the orientation of an underwater acoustic target. However, through a study of sea trial experimental data, the authors found that the sound field interference structures of inbound and outbound ships differ owing to changes in the topography of the shallow continental shelf. Based on this difference, four different convolutional neural networks (CNNs), AlexNet, visual geometry group, residual network (ResNet), and dense convolutional network (DenseNet), are trained to classify inbound and outbound ships using only a single scalar hydrophone. Two datasets, a simulation and a sea trial, are used in the CNNs. Each dataset is divided into a training set and a test set according to the proportion of 40% to 60%. The simulation dataset is generated using underwater acoustic propagation software, with surface ships of different parameters (tonnage, speed, draft) modeled as various acoustic sources. The experimental dataset is obtained using submersible buoys placed near Qingdao Port, including 321 target ships. The ships in the dataset are labeled inbound or outbound using ship automatic identification system data. The results showed that the accuracy of the four CNNs based on the sea trial dataset in judging vessels' inbound and outbound situations is above 90%, among which the accuracy of DenseNet is as high as 99.2%. This study also explains the physical principle of classifying inbound and outbound ships by analyzing the low-frequency analysis and recording diagram of the broadband noise radiated by the ships. This method can monitor ships entering and leaving ports illegally and with abnormal courses in specific sea areas.

KEYWORDS

waveguide invariant, direction estimation, convolutional neural networks, horizontal slowly varying wedge waveguide, single hydrophone

1 Introduction

In target detection and recognition technologies, ocean targets are primarily classified into surface and underwater targets. Synthetic aperture radar (SAR) is one of the main methods used to identify and classify surface ships. Recently, many scholars have applied convolutional neural networks (CNNs) to SAR ship classification. Hog–ShipCLSNet, a novel deep-learning network with hog feature fusion for SAR ship classification, was

proposed by Zhang et al. (2021). Xu et al. (2022) proposed a lightweight deep-learning detector called lite-yolov5.

Underwater acoustic technology is one of the main methods of locating underwater targets. Passive location technology for underwater acoustic targets primarily locates the target by processing the acoustic signal that radiates from the target, which the hydrophone array receives. Because the system does not actively emit an acoustic signal, it exhibits good concealment. In the early stages, owing to the lack of sound field modeling theory, conventional underwater target positioning technology mainly used the time difference of arrival between each hydrophone array element. The most representative method was the three-sub-array positioning method (Carter, 1981). Positioning according to the change in the direction of arrival with the movement of the target, the main representative of which is target motion analysis (TMA) (Nardone et al., 1984). With the development of sound field modeling theory, some location methods have been developed to consider and utilize waveguide phenomena, among which the most typical methods are matched field estimation and sound-field interference fringes.

The three-subarray positioning method assumes that acoustic waves are cylindrical or spherical. This method estimates the distance and azimuth of the target using the difference in the wavefront's curvature and the relative time delay of each element. The calculated amount for the three-subarray positioning method was small. However, when the target is far away, the positioning error of the finite-aperture array is large because the wavefront's curvature changes slightly.

TMA methods include bearings-only and frequency-bearing TMA (Jauffret and Bar-Shalom, 1990; Maranda and Fawcett, 1991). Bearings-only TMA uses only target-bearing information but requires observation platform maneuvering. The frequency-bearing TMA does not require an observation platform to maneuver; it uses frequency and azimuth information as observations. The existing passive positioning method for the TMA requires maneuvering observations or multi-observation platforms, which require much computation and a complex processing system.

The received signal waveform distortion caused by the waveguide multipath dispersion characteristics was ignored by both the three-subarray positioning and TMA methods. In a shallow sea environment, where the boundary of the sea surface and bottom affects the acoustic propagation, the performance is seriously affected because the waveguide effect is not considered.

Matched field processing (MFP) is a generalized beamforming method that uses the spatial complexities of acoustic fields in an ocean waveguide to localize sources in range, depth, and azimuth or to infer the parameters of the waveguide itself. It has experimentally localized sources with accuracies exceeding the Rayleigh and Fresnel limits for depth and a range of two orders of magnitude, respectively. Nevertheless, there are some limitations to the MFP. The most important liability is sensitivity to mismatch. Because MFP exploits the environment, its model must be accurate, especially when seeking high performance (Baggeroer et al., 1993).

Because of their respective limitations, these three underwater acoustic target location methods have unavoidable defects when positioned in shallow-sea environments. To address this dilemma, many scholars have investigated target location methods based on sound field interference structures (Clay, 1987; Thode, 2000; Cockrell and Schmidt, 2010; Song and Cho, 2015; Cho et al., 2016; Song and Cho, 2017; Song et al., 2017; Chi et al., 2021; Li et al., 2022). Hence, the target location method based on the sound-field interference structure is more robust than that based on the matched field.

The single-hydrophone acoustic acquisition and processing system has a simple structure and low cost, making it convenient for installation on floating and submersible buoys, underwater gliders, unmanned underwater vehicles, and other small platforms. However, in conventional research, researchers have believed that the signal received by a scalar hydrophone lacks azimuthal information. Thus, the conventional single-hydrophone target location method can only be used for target ranging, not direction finding.

Unlike conventional research which believes that a single scalar hydrophone does not contain azimuth information, this study inferred that in the area where the topography of the sea floor changes (even if the change is small), the bending degree of the interference fringe in the range-frequency domain is different before and after the range at the closest point of approach (r_{CPA}), and the interference fringe is asymmetrical before and after the range at the closest point of approach (r_{CPA}). Based on this asymmetric feature, we used only a single scalar hydrophone to effectively distinguish between inbound and outbound vessels on a shallow continental shelf. In the concrete implementation, four network structures with good performance in image classification were introduced, namely AlexNet, visual geometry group (VGG), residual network (ResNet), and dense convolutional network (DenseNet), because images often describe the sound field interference structure (Krizhevsky et al., 2012; Simonyan and Zisserman, 2014; He et al., 2015; Huang et al., 2017). This ship classification algorithm can monitor ships entering and leaving ports illegally, supervise inbound and outbound ships, and monitor abnormal heading targets in the channel.

The remainder of this paper is organized as follows. Section 2 summarizes the experimental procedure and preprocessing of experimental data. Section 3 describes the ship classification algorithm, data simulating method, and training of the CNNs. The results of the trained deep learning models are discussed in Section 4. Section 5 introduces the definition of generalized waveguide invariants to analyze the physical factors responsible for the differences in the interference structure (Gao et al., 2022). Finally, Section 6 presents the conclusions of this study.

2 Experiment

2.1 Experiment procedure

The experimental data used here were collected from a submarine buoy deployed by the Ocean University of China. The experimental setup is shown in Figure 1A. The experimental area comprised shallow water with a depth of approximately 24 m and a wedge-shaped seabed with a slowly changing horizontal (a slope of 0.057°). The submarine buoy recorded the underwater noise from

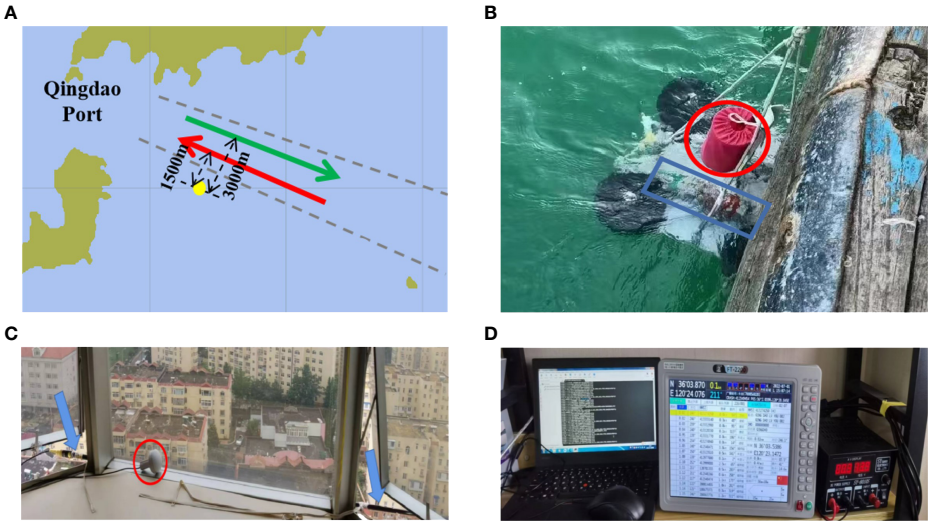


FIGURE 1
Sea trial system. **(A)** Submarine topographic map. The yellow circle marks the position of the submarine buoy. The middle part of the black dotted line is the channel. The red arrow is the direction of the inbound ship, which is about 1500 m away from the submarine buoy. The green arrow is the direction of the outbound ship, which is about 3000 m away from the submarine buoy. **(B)** Submersible buoys. The part in the red circle is the hydrophone part of the self-contained hydrophone, and the cylinder in the blue box is its data-sampling and processing system. **(C)** AIS signal receiving terminal. The part in the red circle is the GPS antenna, and the long black pole pointed by the blue arrow is the AIS signal antenna. **(D)** AIS data system.

321 inbound and outbound ships at Qingdao Port between June 15 and 22, 2022, the structure of which is shown in Figure 1B. The trajectories of these ships originated from the signals received by the automatic identification system (AIS) placed on the shore, as shown in Figures 1C, D.

2.2 Experimental data preprocessing

Data preprocessing is required to achieve better performance. The low-frequency analysis and recording (LOFAR) diagram is a basic time-frequency representation often used for localizing sources.

The short-time Fourier transform (STFT) can transform the raw signal into a LOFAR diagram using STFT. The time of the cloest point to the approach (t_{CPA}) was estimated based on the LOFAR diagram. After comparing t_{CPA} with the AIS data, we labeled the LOAR diagram as an inbound or outbound ship.

By processing the experimental data, we found that the structures of the LOFAR diagrams of inbound and outbound ships are different. LOFAR diagrams of Figures 2A, B are the LOFAR diagrams of the same ship's departure and arrival. In the experimental data, the interference fringes of the outbound ships generally bent down on the right side. In contrast, those of the inbound ships bent down on the left side. Owing to this difference,

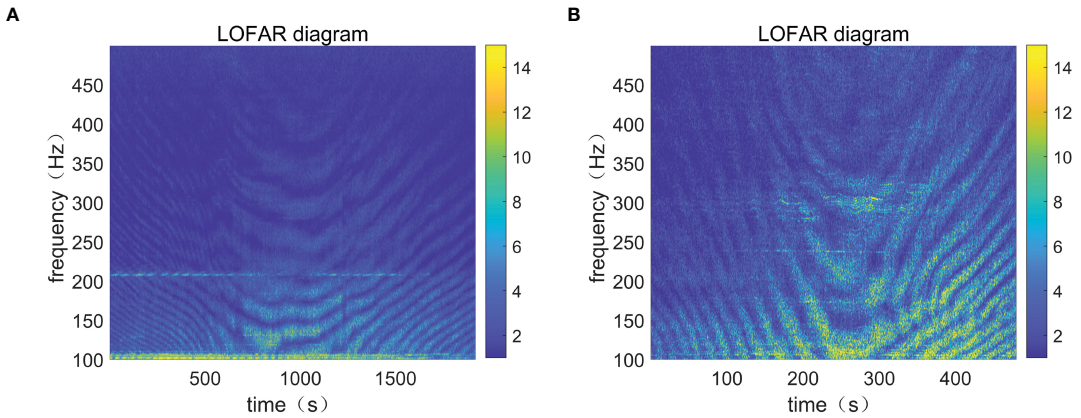


FIGURE 2
LOFAR diagrams of the same ship entering and leaving the port. **(A)** Inbound time-frequency diagram. **(B)** Outbound time-frequency diagram.

we used CNNs to extract features and classify inbound and outbound ships.

3 Method

This study proposed a method based on CNNs to classify inbound and outbound ships using a single scalar hydrophone on a shallow continental shelf. The flowchart is shown in Figure 3A. We used the STFT to transform the raw signal into a LOFAR diagram. The diagram was used as the input to the trained deep-learning models after edge detection to classify the ships.

Training deep-learning models require training datasets of various labeled samples. As the range at the closest point of approach, r_{CPA}, was relatively fixed in the experimental data, simulation data were also used during the training and validation steps. As shown in Figure 3B, both experimental and simulation data were used to train the models, which were tested with the experimental and simulation test sets, respectively.

3.1 Simulation data

The simulation was divided into three steps. First, building the ship radiated noise model and getting the ship radiated noise $s(\omega)$, and second, obtaining the channel transfer function $H(\omega)$ using the sound propagation calculation model Range-dependent acoustic model (RAM). Finally, the hydrophone reception signal is obtained by multiplying $H(\omega)$ and $s(\omega)$.

3.1.1 Ship noise simulation

Ship noise is mainly composed of a line spectrum and a continuous spectrum. Its mathematical model can usually be expressed as

$$S(t) = [1 + G(t)] \times S_x(t) + S_l(t) \quad (1)$$

The line spectrum component can be simulated by generating a series of sinusoidal signals, and its parameters can be set according to the following methods (He and Zhang, 2005).

(1) For line spectrum below 100 Hz, the fundamental frequency of shaft frequency line spectrum can be set as s, and the frequency of blade and harmonic line spectrum is mns Where s is the propeller speed; the unit is *turn/s*; n is the number of propeller blades, and m is the harmonic number.

(2) The line spectrum with a frequency of 100–1000 Hz has no significant relationship with the ship's speed but varies with the type of ship. K frequencies can be set without loss of generality.

The construction of the continuous spectral data was completed in three steps. First, we constructed the ship noise source level for different tonnages and speeds according to the empirical equation summarized by Ross (1976), as shown in Figure 4A. Next, we constructed an finite impulse response (FIR) filter with a specific frequency response using the LMS-adaptive algorithm, as shown in Figure 4B. Finally, a continuous spectrum of the radiated noise of the ship was obtained by inputting Gaussian white noise through the filter.

After adding a line spectrum to the continuous spectrum, the power spectrum of the radiated noise of the ship was obtained, as shown in Figure 4C.

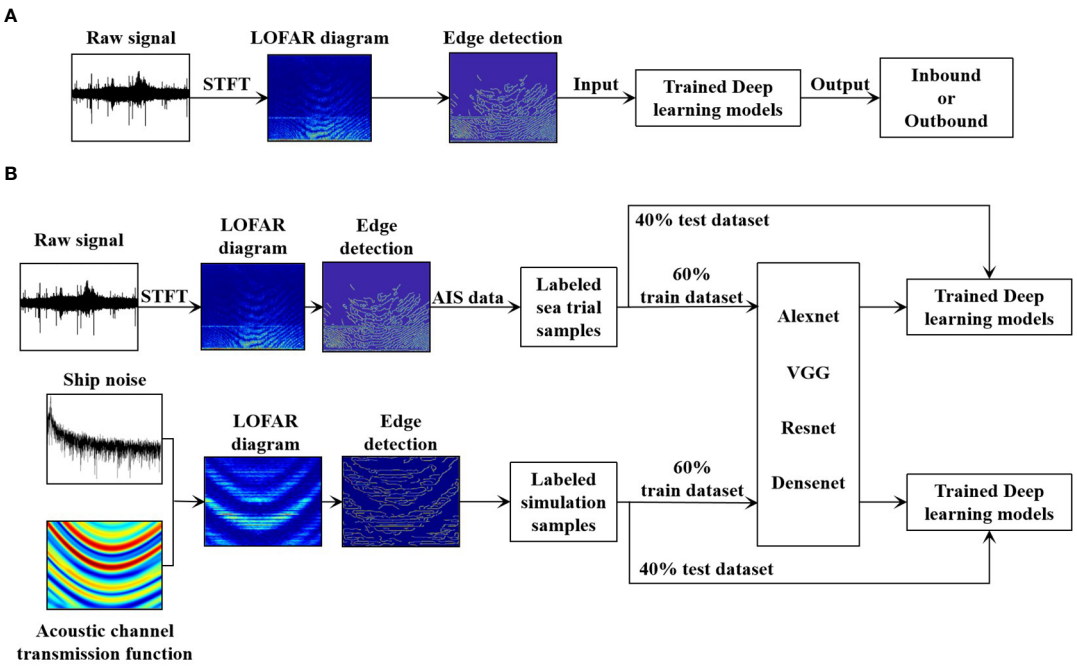


FIGURE 3
Ships classification algorithm flow chart. **(A)** Overall flow chart of ships classification algorithm. **(B)** Deep-learning models training flow chart.

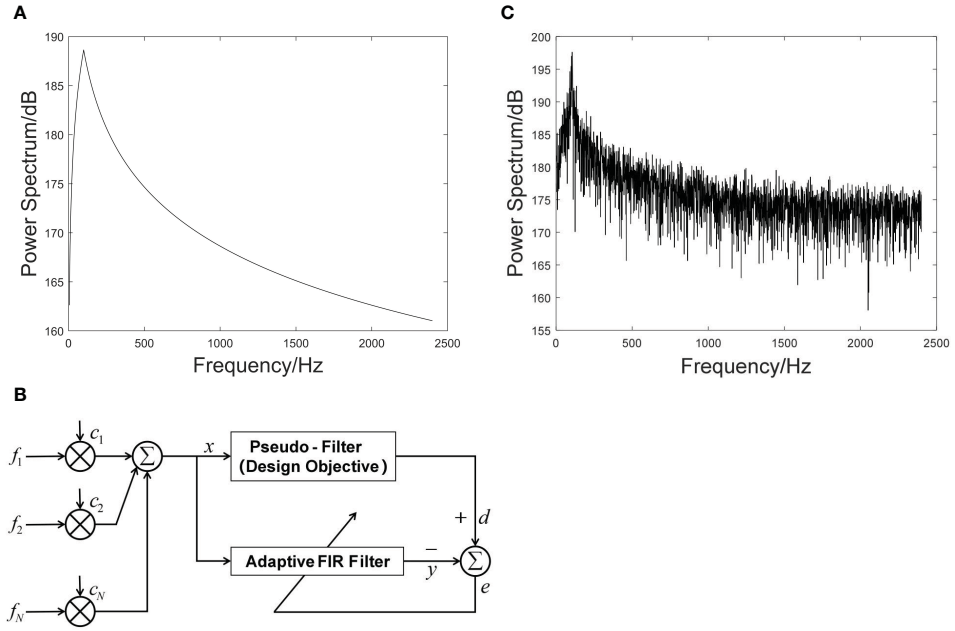


FIGURE 4
The ship's radiated noise. **(A)** The ship noise source level. **(B)** Structure of FIR filter with specific frequency. **(C)** Power spectrum of ship's radiated noise.

3.1.2 Transfer function simulation

In this section, the underwater acoustic propagation software RAM is used to simulate the transfer function $H(\omega)$ of the channel (Collins, 1993).

The main environmental parameters for the simulation were as follows. The experimental sea area was off the coast of Qingdao, and the seabed terrain was a typical horizontal, slowly varying wedge seabed. Therefore, a wedge-shaped seafloor was used for the simulation. The sound velocity of the seabed is set as 1620 m·s^{-1}, the seabed density is 1.76 g·cm^{-3}, and the seabed attenuation is 0.3 dB·λ^{-1}. The sound velocity profile was obtained from the measured CTD data in the offshore waters of Qingdao on June 30, 2022. The acoustic source emission band was 200 – 400 Hz; the receiver depth was 26 m; the time of the closest point of approach (t_{CPA})was 150 s; the range at the closest point of approach (r_{CPA})was set to 1000 m; the sound source depth (d) was 5 m, and the motion speed (v) was 10 m/s. The 3D structure chart is shown in Figure 5A.

The spectrum received by the hydrophone is calculated every 2 s. After splicing, the time-frequency diagram, as shown in Figure 5B, is obtained. The transfer function $H(\omega)$ of the channel under different conditions is obtained by changing the parameters such as d, v, and r_{CPA}.

The spectrum of the radiated noise of the ship was multiplied by the transfer function spectrum, and the LOFAR diagram was obtained after splicing. As shown in Figure 6, the signal-to-noise ratio is set at 10 dB.

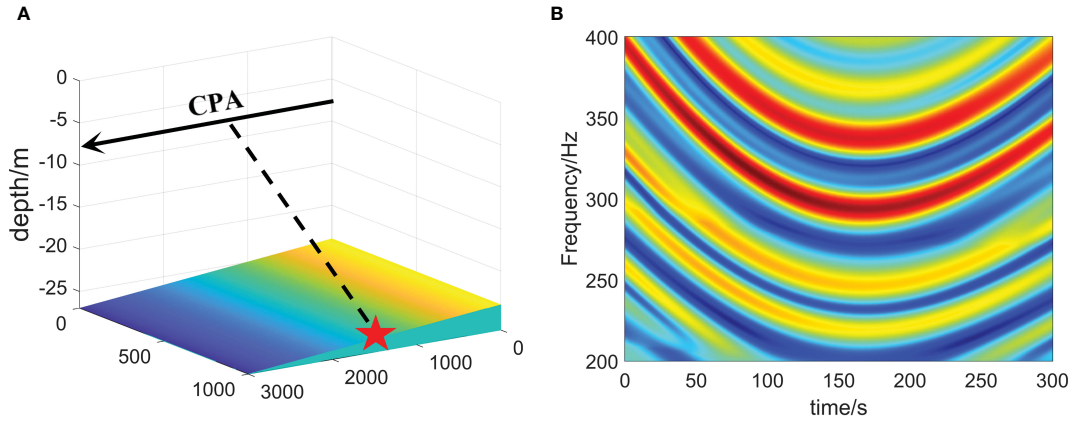


FIGURE 5
Simulation of transfer function. **(A)** 3D water depth distribution. **(B)** Time-frequency diagram of waveguide transfer function.

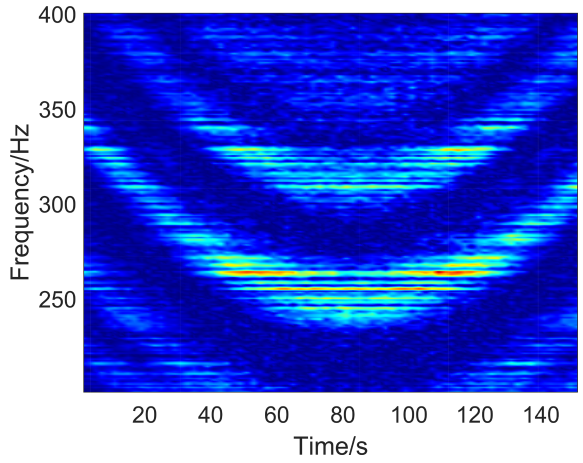


FIGURE 6
Time-frequency diagram of the received signal.

3.2 Network architecture

3.2.1 AlexNet

In 2012, Krizhevsky et al. proposed AlexNet, which realized a TOP5 error rate of 15.4% (The TOP5 error rate is the probability that, given an image, its label is not in the top five outcomes that the model considers most likely), and realized a deep convolutional neural network structure in a large-scale image dataset for the first time. AlexNet includes eight layers of transformations, including five convolution layers, two fully connected hidden layers, and one fully connected output layer (Krizhevsky et al., 2012), as shown in Figure 7. The network uses a rectified linear unit (ReLU) as a nonlinear mapping function, which makes the model converge more rapidly. The dropout mechanism was used to effectively reduce the overfitting problem to a certain extent, and the GPU replaced the CPU for calculations, significantly improving the training speed of the network.

3.2.2 VGG

Simonyan and Zisserman (2014) studied the depth of CNNs based on AlexNet, proved that increasing the depth of the network can affect its performance to a certain degree, and proposed the idea of building a depth model by reusing simple basic blocks. The network structure of the VGG is shown in Figure 8. The first part comprises convolution and convergence layers, and the second comprises a fully connected layer. The original VGG network has five convolution blocks, of which the first two blocks each have one convolution layer, and the last three blocks each contain two convolution layers. Because the network uses eight convolution layers and three fully connected layers, it is usually called VGG-11.

Compared to AlexNet, VGG uses a smaller convolution core and a deeper network structure. However, the increase in the network depth is limited. Many network layers leads to network degradation.

3.2.3 ResNet

Based on VGG, He et al. (2015) effectively solved the problem of decreasing the accuracy of the training set with the deepening of the network through the design of residual blocks.

The basic structure of the residual block is shown on the right side of Figure 9. The residual block changes the learning target to the difference between target values $H(X)$ and x, called the residual. Residual mapping is often easier to optimize. Through the design of the residual block, some neural network layers can be artificially created to skip the connection of neurons in the next layer, thus weakening the strong connections between each layer.

3.2.4 DenseNet

In 2017, Huang et al. proposed DenseNet based on ResNet. However, unlike ResNet, DenseNet proposed a more radical dense connection mechanism where all layers are interconnected (2017). Specifically, each layer accepts all the layers in front of it as its additional input, as shown in Figure 10, which can achieve feature reuse and improve efficiency.

3.3 Training CNNs

3.3.1 Input data preprocessing

As shown in Figure 11, before inputting in the CNNs, all of the LOFAR diagrams were resampled to 256×256 for the

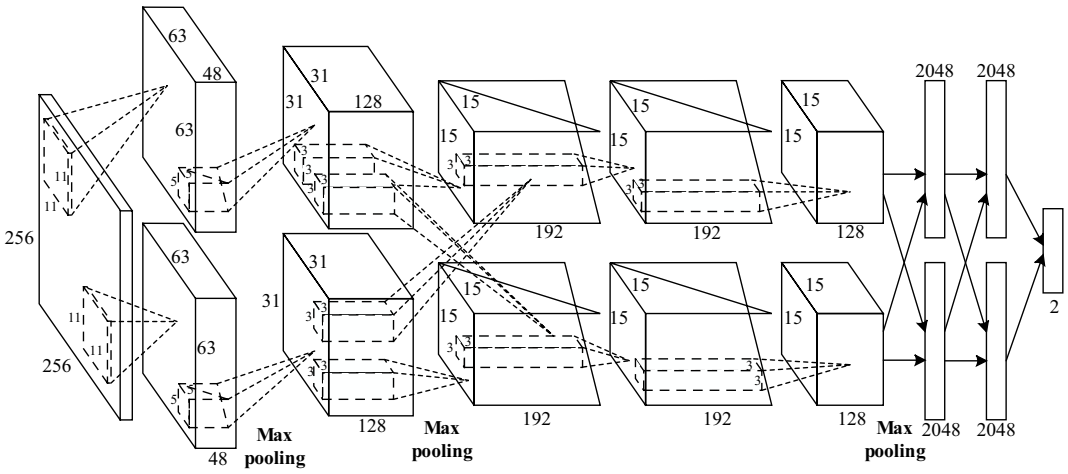


FIGURE 7
Structure of AlexNet.

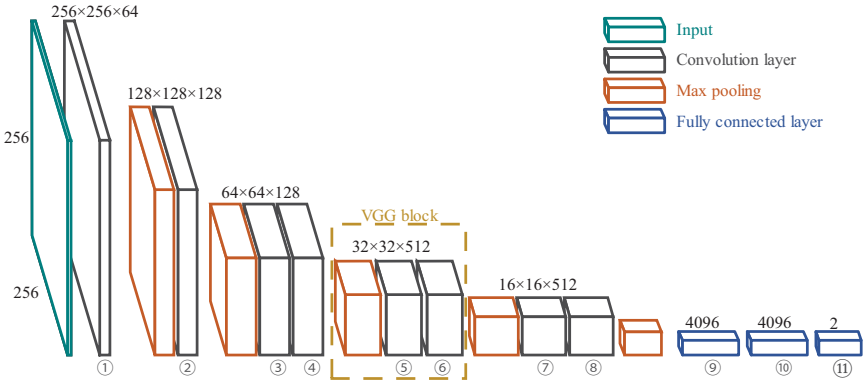


FIGURE 8
Network structure of VGG.

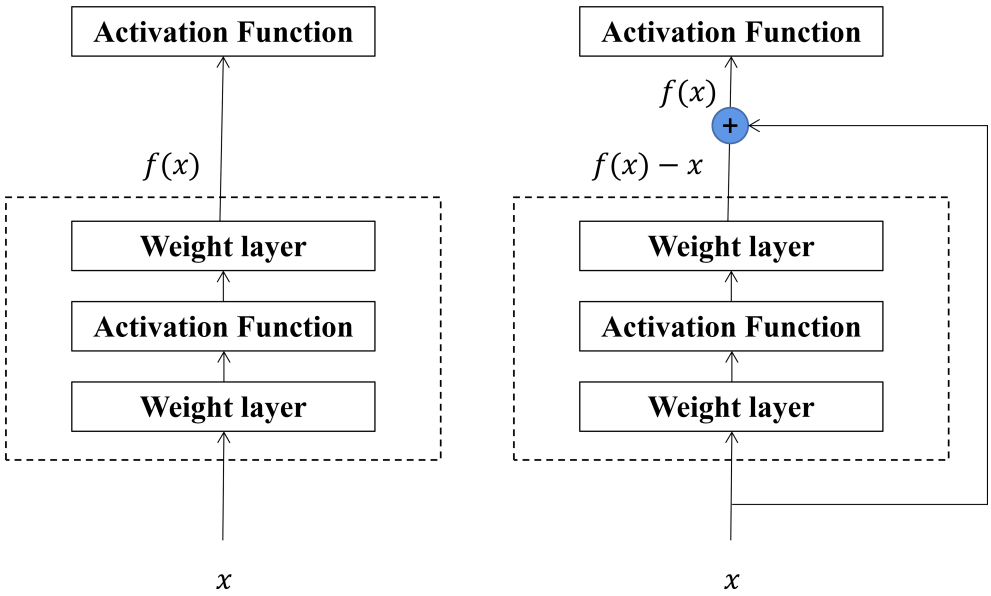


FIGURE 9
Structure of residual block.

reduction of the computation, smoothed by mean filtering to reduce salt-and-pepper noise, and the Canny operator detected edges for better performance. The experimental and simulation data were split into training and test sets at the same ratio. The ratio of the training set to the test set is 40%:60%.

3.3.2 Network training

The implementation of the neural networks mentioned in Section 3.2 was done in Python 3 using the open-source Pytorch (Paszke et al., 2019). The network was trained for 100 and 20 epochs on the sea trial and simulation datasets, respectively. The batch size was set to 32. An Intel Core i7-9700 3.00 GHz CPU trained the networks. The final trained model could complete the classification of 128 samples in 8.53 s.

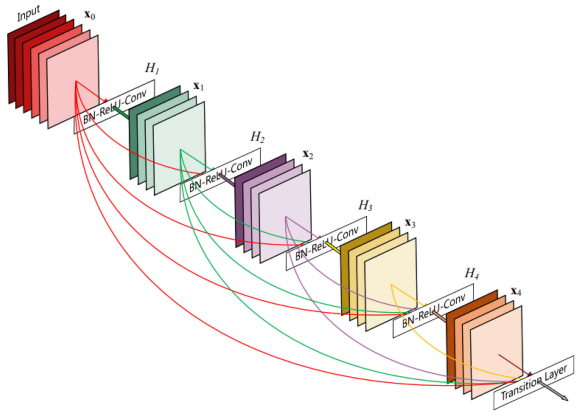


FIGURE 10
Dense connection structure.

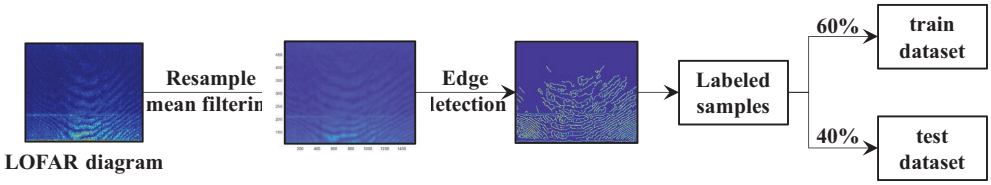


FIGURE 11
Input data preprocessing flow chart.

4 Results

4.1 Accuracy and train loss

The results for the simulation training dataset are presented in Figure 12. From the perspective of training loss, ResNet and DenseNet declined rapidly, whereas AlexNet and VGG declined relatively slowly. From the training set's perspective, the four networks' training accuracy reached 100%, but that of AlexNet and VGG fluctuated significantly. For the test dataset, the final test accuracies of AlexNet, VGG, ResNet, and DenseNet were 99.49%, 100%, 100%, and 100%, respectively. AlexNet and VGG exhibited larger fluctuations.

The results for the experimental dataset are illustrated in Figure 13. Compared to the simulation dataset, the experimental dataset fluctuated greatly, which may be due to the influence of marine environmental noise (such as the calls of marine organisms) in individual samples. From the perspective of the test dataset, the final test accuracies of AlexNet, VGG, ResNet, and DenseNet were 90.63%, 95.51%, 96.63%, and 99.22%, respectively. AlexNet and VGG fluctuated less, but their final test-set accuracies were lower. ResNet fluctuated more; however, its final test set accuracy was higher. DenseNet fluctuated less but had the highest final test set accuracy.

Overall, ResNet and DenseNet performed better than AlexNet and VGG on both the simulation and experimental datasets, possibly because they used a residual block design with a deeper network structure. In the experimental dataset, the stability of DenseNet was better than that of ResNet, and the final test set accuracy of DenseNet was higher, which may be because DenseNet adopts a denser connection mechanism.

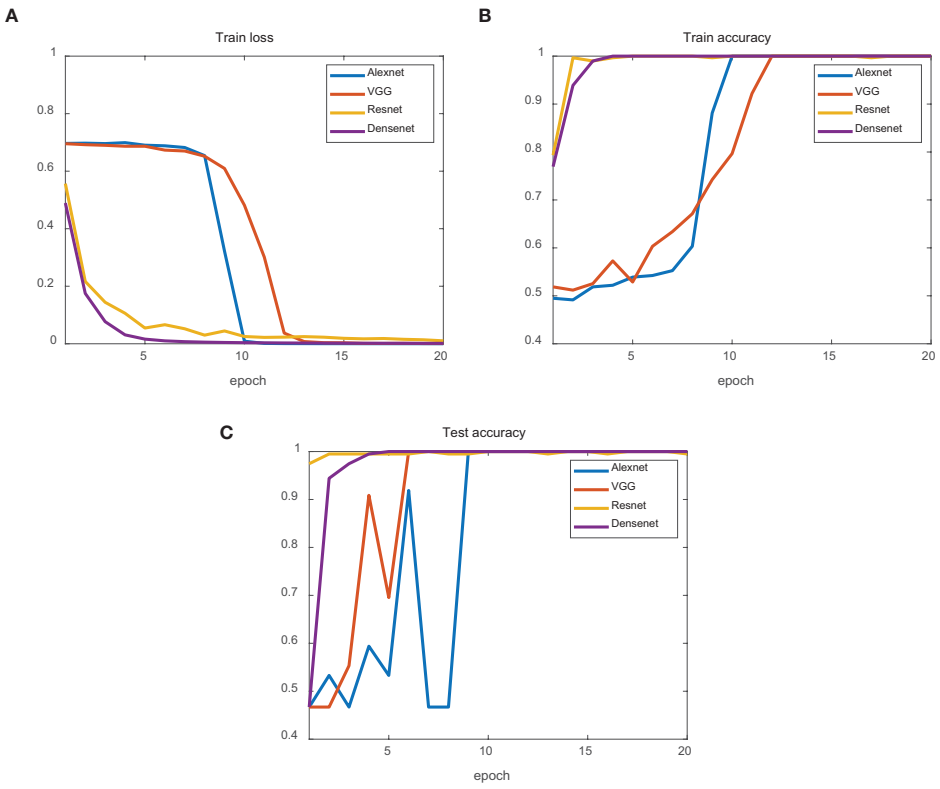


FIGURE 12
Simulation data classification results. **(A)** Train loss of four networks. **(B)** Train accuracy of four networks. **(C)** Test accuracy of four networks. AlexNet is the blue line. VGG is the red line. ResNet is the yellow line. DenseNet is the purple line.

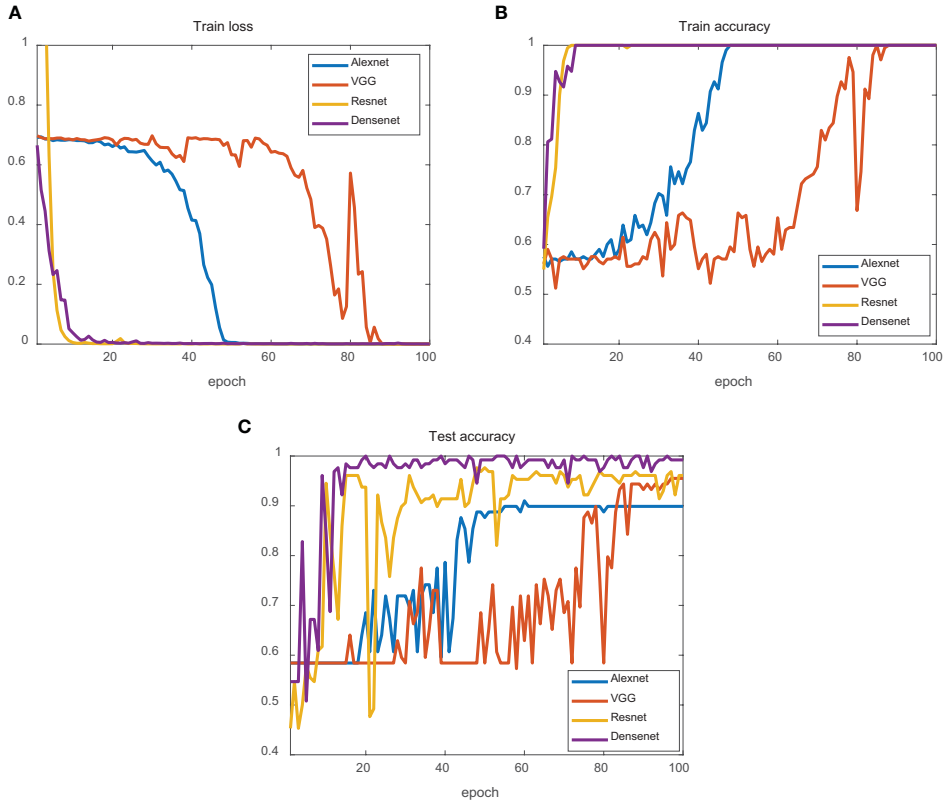


FIGURE 13
Experimental data classification results. **(A)** Train loss of four networks. **(B)** Train accuracy of four networks. **(C)** Test accuracy of four networks. AlexNet is the blue line. VGG is the red line. ResNet is the yellow line. DenseNet is the purple line.

4.2 Confusion matrixes

Table 1 presents the confusion matrices of the four networks trained using the experimental datasets. Each column represents a prediction, and each row represents the true label of the data. Among the 128 inbound and outbound ships, AlexNet mistakenly judged three inbound ships as outbound and nine outbound ships as inbound. The VGG mistakenly judged one inbound ship as outbound and five outbound ships as inbound. ResNet recognized 70 outbound ships and 54 diagrams from 58 test diagrams of inbound ships. DenseNet outperformed the other three CNNs and recognized all 58 inbound ships and 68 diagrams from 70 testing diagrams of outbound ships. DenseNet offers a reliable method for classifying inbound and outbound ships.

5 Analysis of physical principles

This study introduced the concept of a generalized waveguide invariant to analyze the reasons for the differences in the interference fringe structures of inbound and outbound ships. Based on the conventional definition of waveguide invariance, the generalized waveguide invariant considers the effect of azimuth change on the waveguide invariant β and derives a new definition equation.

Assuming that the sound source moves along a straight line and its track does not pass through the receiver, the movement of the sound source will not only cause a change in the sound propagation path distance with time but also cause a change in the azimuth angle with time.

In Figure 14, the orange circle represents the receiver's position, the blue rectangle represents the sound source, v represents the sound source's moving speed, v_τ and v_n the radial and tangential velocities, respectively, and θ represents the azimuth.

In this case, the waveguide-invariant β was related to the distance and azimuth of the sound propagation path. Based on the definition equation $\boldsymbol{\beta} = \frac{r}{\omega}\frac{d\omega}{dr}$ of the conventional waveguide invariant, the definition equation of the generalized waveguide invariant is derived as shown in Equation 2.

$$\beta = \frac{\mathbf{r}}{\boldsymbol{\omega}}\frac{\mathbf{d}\boldsymbol{\omega}}{\mathbf{dr}} = -\frac{\mathbf{s}_{\mathbf{p,mn}}(\boldsymbol{\omega},\boldsymbol{\theta},\mathbf{r}) + \frac{\partial}{\partial\boldsymbol{\theta}}\left\{\frac{1}{\mathbf{r}}\int_0^{\mathbf{r}}\mathbf{s}_{\mathbf{p,mn}}(\boldsymbol{\omega},\boldsymbol{\theta},\mathbf{r}')\mathbf{dr}'\right\}\mathbf{cot}\boldsymbol{\theta}}{\frac{1}{\mathbf{r}}\int_0^{\mathbf{r}}\mathbf{s}_{\mathbf{g,mn}}(\boldsymbol{\omega},\boldsymbol{\theta},\mathbf{r}')\mathbf{dr}'} \quad (2)$$

In Equation 2, the first term of the molecule contributes to the change in distance along the sound propagation path, and the second term corresponds to the change in azimuth. When the sound source is close to the nearest point and azimuth $\theta \to 0$, there is a singularity in the waveguide invariants' values. When the distance between the sound source and the receiver is far enough, the azimuth change θ is very weak. β is mainly

TABLE 1 Confusion matrixes of four networks trained by experimental datasets.

AlexNet		Inbound	Outbound	ResNet		Inbound	Outbound
	Inbound	55	3		Inbound	54	4
	Outbound	9	61		Outbound	0	70
VGG		Inbound	Outbound	DenseNet		Inbound	Outbound
	Inbound	57	1		Inbound	58	0
	Outbound	5	65		Outbound	2	68

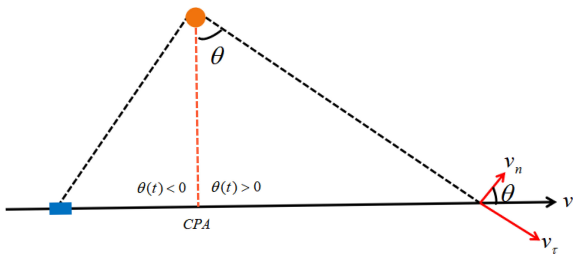


FIGURE 14
Schematic of sound source moving along a straight line without passing the receiver.

determined by the first term of Equation 2, and the influence of azimuth on waveguide invariants can be ignored. The first term in Equation 2 is a classical waveguide invariant expression. After adding the second term, the waveguide invariant β is related to the distance and the azimuth variation term between the sound source and the receiver.

According to the theory of generalized waveguide invariants, the value of waveguide invariants changes abruptly before and after the target ship passes the nearest point, which causes asymmetric interference fringes on the time-frequency diagram. Furthermore, the directions of the inbound and outbound ships are opposite; therefore, their interference fringe structures show different characteristics: one is high on the left and low on the right, and the other is high on the right and low on the left.

6 Conclusion

Here, we first found that the spectrum interference structure of the acoustic signal received by a single hydrophone is asymmetric in sea trial experimental data. Then we used this feature to classify inbound and outbound ships using a single hydrophone through CNNs and explained the physical principle through generalized waveguide invariants.

This method overcame the idea that single-scalar hydrophones can only be used for ranging, and not direction-finding. This algorithm classified the direction of the target ship using only a single-scalar hydrophone. However, at this stage, it was only possible to determine approximately whether a ship was inbound or outbound, and a more detailed course judgment could not be completed. When the seabed topography was known for specific sea areas, this method could achieve more detailed course discrimination and complete the positioning of the target ship or underwater target. This algorithm could achieve more comprehensive sea area monitoring by combining ls-ssdd-v1.0 and official-ssdd with SAR ship classification and identification. This issue should be addressed in future studies.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

Conceptualization, DZG. Methodology, DZG and DDG. Software, DDG and ZC. Experiment, YL, XZ, and DDG. Writing, review, and editing, DZG, DDG, XZ, WS, and XL. All authors contributed to the article and approved the submitted version.

Funding

The word was supposed by the National Natural Science Foundation of China (Grant Nos. 12274385, 11874331, 12004359, 52001296).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Baggeroer, A. B., Kupennan, W. A., and Mikhalevsky, P. N. (1993). An overview of matched field methods in ocean acoustics. *IEEE J. Oceanic Eng.* 18 (4), 401–424. doi: 10.1109/48.262292

Carter, G. C. (1981). Time delay estimation for passive sonar signal processing. *IEEE Trans. Acoustics Speech Signal Process.* 29 (3), 463–470. doi: 10.1109/TASSP.1981.1163560

Chi, J., Gao, D. Z., Zhang, X. G., Zhang, X. Y., and Wang, Z. Z. (2021). Motion parameter estimation of multitonal sources with a single hydrophone[J]. *JASA Express Lett.* 1 (1), 016006. doi: 10.1121/100003368

Cho, C., Song, H. C., and Hodgkiss, W. S. (2016). Robust source-range estimation using the array/waveguide invariant and a vertical array. *J. Acoustical Soc. America* 139 (1), 63~69. doi: 10.1121/1.4939121

Clay, C. S. (1987). Optimum time domain signal transmission and source location in a waveguide. *J. Acoustical Soc. America* 81 (3), 660–664. doi: 10.1121/1.394834

Cockrell, K. L., and Schmidt, H. (2010). Robust passive range estimation using the waveguide invariant. *J. Acoustical Soc. America* 127 (5), 2780. doi: 10.1121/1.3337223

Collins, M. D. (1993). A split-step pade solution for the parabolic equation method. *Acoust Soc. Am(S0001-4966)* 93 (4), 1736–1742. doi: 10.1121/1.406739

Gao, D. Z., Kang, D. X., Song, W. H., Li, X. L., and Li, Y. Z. (2022). Generalized waveguide invariant. *Tech. Acoustics* 41 (03), 403–411. doi: 10.16300/j.cnki.1000-3630.2022.03.014

He, Z. Y., and Zhang, Y. P. (2005). Modeling and simulation research of ship-radiated noise. *Audio Eng.* 12), 52–55. doi: 10.16311/j.audioe.2005.12.014

He, K., Zhang, X. Y., Ren, S. Q., and Sun, J. (2015). Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. doi: 10.1109/CVPR.2016.90

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. *IEEE Conf. Comput. Vision Pattern Recognition (CVPR)*. doi: 10.1109/CVPR.2017.243

Jauffret, C., and Bar-Shalom, Y. (1990). Track formation with bearing and frequency measurements in clutter. *IEEE Trans. Aerospace Electronic Syst.* 26 (6), 999–1010. doi: 10.1109/7.62252

Krizhevsky, A., Sutskever, I., and Hinton, G. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM* 60 (6), 84–90. doi: 10.1145/3065386

Li, Q. L., Song, W. H., Gao, D. Z., Chi, J., and Gao, D. Y. (2022). Passive localization of shallow sea target using interferogram. *Acta Acustica* 47 (05), 625–633. doi: 10.15949/j.cnki.03710025.2022.05.012

Maranda, B. H., and Fawcett, J. A. (1991). Detection and localization of weak targets by space-time integration. *IEEE J. Oceanic Eng.* 16 (2), 189–194. doi: 10.1109/48.84135

Nardone, S. C., Lindgren, A. G., and Gong, K. F. (1984). Fundamental properties and performance of conventional bearings-only target motion analysis. *Automatic Control IEEE Trans.* 29 (9), 775~787. doi: 10.1109/TAC.1984.1103664

Paszke, A., Gross, S., Massa, F., Lerer, A., and Chintala, S. (2019). Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst.* 32, 1912.01703. doi: 10.48550/arXiv.1912.01703

Ross, D. (1976). *Mechanics of underwater noise* (New York: Pergamon Press).

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for Large-scale image recognition. *CoRR*, abs/1409.1556. doi: 10.48550/arXiv.1409.1556

Song, H. C., and Cho, C. (2015). The relation between the waveguide invariant and array invariant. *J. Acoustical Soc. America* 138 (2), 899~903. doi: 10.1121/1.4927090

Song, H. C., and Cho, C. (2017). Array invariant-based source localization in shallow water using a sparse vertical array. *J. Acoustical Soc. America* 141 (1), 183~188. doi: 10.1121/1.4973812

Song, H. C., Cho, C., Byun, G., and Kim, J. S. (2017). Cascade of blind deconvolution and array invariant for robust source-range estimation. *J. Acoustical Soc. America* 142 (4), 3270~3273. doi: 10.1121/1.4983303

Thode, A. M. (2000). Source ranging with minimal environmental information using the virtual receiver and waveguide invariant concepts. *J. Acoustical Soc. America* 107 (5), 2867. doi: 10.1121/1.1289409

Xu, X. W., Zhang, X. L., and Zhang, T. W. (2022). Lite-YOLOv5: a lightweight deep learning detector for on-board ship detection in Large-scene sentinel-1 SAR images. *Remote Sens.* 14 (4), 1018–1018. doi: 10.3390/rs14041018

Zhang, T., Zhang, X., Ke, X., Liu, C., Xu, X., Zhan, X., et al. (2021). HOG-ShipCLSNet: a novel deep learning network with HOG feature fusion for SAR ship classification. *IEEE Trans. Geosci. Remote Sens.* 60, 1–22. doi: 10.1109/TGRS.2021.3082759

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 18 May 2023
DOI 10.3389/fmars.2023.1181095

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Wei-Jen Huang,
National Sun Yat-sen University, Taiwan
Abhra Chanda,
Jadavpur University, India

*CORRESPONDENCE
Chunli Liu
chunliliu@sdu.edu.cn

RECEIVED 07 March 2023
ACCEPTED 02 May 2023
PUBLISHED 18 May 2023

CITATION
Li W, Liu C, Zhai W, Liu H and Ma W (2023) Remote sensing and machine learning method to support sea surface pCO_2 estimation in the Yellow Sea. *Front. Mar. Sci.* 10:1181095. doi: 10.3389/fmars.2023.1181095

COPYRIGHT
© 2023 Li, Liu, Zhai, Liu and Ma. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Remote sensing and machine learning method to support sea surface pCO_2 estimation in the Yellow Sea

Wei Li[1], Chunli Liu[1]*, Weidong Zhai[2], Huizeng Liu[3] and Wenjuan Ma[1]

[1]Marine College, Shandong University, Weihai, China, [2]Frontier Research Center, Southern Marine Science and Engineering Guangdong Laboratory, Zhuhai, China, [3]Institute for Advanced Study, Shenzhen University, Shenzhen, China

With global climate changing, the carbon dioxide (CO_2) absorption rates increased in marginal seas. Due to the limited availability of *in-situ* spatial and temporal distribution data, the current status of the sea surface carbon dioxide partial pressure (pCO_2) in the Yellow Sea is unclear. Therefore, a pCO_2 model based on a random forest algorithm has been developed, which was trained and tested using 14 cruise data sets from 2011 to 2019, and remote sensing satellite sea surface temperature, chlorophyll concentration, diffuse attenuation of downwelling irradiance, and in-situ salinity were used as the input variables. The seasonal and interannual variations of modeled pCO_2 were discussed from January 2003 and December 2021 in the Yellow Sea. The results showed that the model developed for this study performed well, with a root mean square difference (RMSD) of 43 μatm and a coefficient of determination (R^2) of 0.67. Moreover, modeled pCO_2 increased at a rate of 0.36 μatm $year^{-1}$ (R^2 = 0.27, $p < 0.05$) in the YS, which is much slower than the rate of atmospheric pCO_2 (pCO_2^{air}) rise. The reason behind it needs further investigation. Compared with pCO_2 from other datasets, the pCO_2 derived from the RF model exhibited greater consistency with the in-situ pCO_2 (RMSD = 55 μatm). In general, the RF model has significant improvement over the previous models and the global data sets.

KEYWORDS

machine learning, random forest, remote sensing, the Yellow Sea, pCO_2

1 Introduction

The rapid growth of fossil fuel usage and industry has increased the atmospheric carbon dioxide (CO_2) concentration by approximately 40% since the Industrial Revolution (Landschützer et al., 2014; Friedlingstein et al., 2022). Global oceans absorb 30% of the CO_2 released by industry and human activities and they are a significant sink for

atmospheric CO_2. Coastal seas cover 7% of the oceanic surface area but the sea-air exchange carbon fluxes (FCO_2) comprise approximately 25–50% of the global oceans (Laruelle et al., 2018), and thus they play important roles in absorbing atmospheric CO_2 (Dai et al., 2022). Due to the effects of the complex physical environment and biological activities, great errors occur in estimations of FCO_2 in coastal seas (Landschützer et al., 2018; Mignot et al., 2022). Therefore, estimating sea surface carbon dioxide partial pressure (pCO_2) accurately for coastal seas is critical for precisely estimating the global FCO_2 (Laruelle et al., 2018).

In general, pCO_2 is regulated by thermodynamic effects, biogeochemical effects, mixing effects, and air–sea exchange effects (Liu et al., 2019; Ye et al., 2022). Some environmental variables can characterize these four effects. In particular, the sea surface temperature (SST, °C) directly reflects thermodynamic effects, while the chlorophyll concentration (Chl, mg m^{-3}) and diffuse attenuation of downwelling irradiance (Kd, m^{-1}) can indicate biogeochemical effects on the surface pCO_2. In addition, the SST, salinity (SSS, psu), and mixed layer depth (MLD, m) are closely related to mixing effects, and the wind speed can characterize the sea–air exchange process (Gu et al., 2021).

Due to their unique advantage in terms of high spatiotemporal resolution, satellite approaches are efficient for observing pCO_2. In previous studies, both semi-analytical (Hales et al., 2012; Bai et al., 2015; Chen et al., 2017) and empirical approaches (Lohrenz et al., 2010; Tao et al., 2012; Qin et al., 2014; Chen et al., 2016; Chen et al., 2019; Fu et al., 2020) were used to estimate the sea surface pCO_2. Many studies have used satellite data to estimate the sea surface pCO_2, but recent studies also examined and compared the capability of semi-analytical and empirical algorithms for estimating the coastal pCO_2 (Chen et al., 2017; Chen et al., 2019). However, the high spatiotemporal variability and diversity of pCO_2, the inaccuracy of satellite data, and limited availability of *in-situ* pCO_2 data from coastal seas make it challenging to establish a model of pCO_2. Several efforts have been made to construct various algorithms or models, but the satellite-derived pCO_2 in coastal seas generally has higher uncertainty than that for open seas, and the root mean square difference (RMSD) can be as high as 90 µatm (Chen et al., 2019).

The Yellow Sea (YS) is an important coastal sea in the west Pacific Ocean. The pCO_2 in the YS has considerable seasonal variations and an unbalanced spatial distribution (Wang and Zhai, 2021). For example, extremely high pCO_2 values have been observed during the summer in the center of the YS, whereas extremely low pCO_2 values have been observed in the southwestern YS (Qu et al., 2014; Zhai, 2018). Since the 1980s, many studies have investigated carbonate, pCO_2, and FCO_2 in the YS (Xue et al., 2011; Qu et al., 2014; Zhai et al., 2014; Zhai, 2018; Choi et al., 2019; Deng et al., 2021). However, accurately quantifying pCO_2 and FCO_2 in the YS remains a challenge. In particular, Wang and Zhai (2021) indicated that the YS is a carbon sink and FCO_2 is about −0.5 ± 1.9 mol m^{-2} $year^{-1}$, whereas Qu et al. (2014) suggested that the YS is a carbon source. In addition, the physical and biological conditions in coastal seas have changed due to rapid climate change. For example, SST and Chl have increased (Liu et al., 2021; Lu et al., 2021). These variations will have influenced the changes in the sea surface pCO_2.

Indeed, recent studies showed that the CO_2 absorption rates increased in some coastal seas (Li and Zhai, 2019; Xiong et al., 2020). To the best of our knowledge, no previous studies have quantified the long-term trend in the carbon absorption capacity of the YS due to the lack of *in-situ* pCO_2 data over the entire YS. Thus, in order to accurately quantify the pCO_2 in the YS and understand the response of the pCO_2 to global climate change, we developed an inversion model of pCO_2 in the YS in the present study. Two previous remote sensing studies investigated the pCO_2 in the YS (Tao et al., 2012; Qin et al., 2014), and both used *in-situ* SST and Chl data to establish multiple polynomial regression (MPR) models. This modeling method is simple but the errors are large. Therefore, in the present study, we aimed: (1) to develop machine learning models for accurately deriving pCO_2 from satellite remote sensing data; and (2) to analyze the long-term trend in the pCO_2 during 2003–2021 in the YS.

2 Materials and methods

2.1 Study area

The YS is a semi-enclosed shelf shallow sea (29.5°N–40.5°N, 118.5°E–126.5°E) located west of the Liaodong Peninsula and east of the Korean Peninsula (Figure 1). The mean water depth is 44 m (Liu et al., 2009). The areas and depths of the North Yellow Sea (NYS) and South Yellow Sea (SYS) are 70 × 10^3 km^2 and 38 m, and

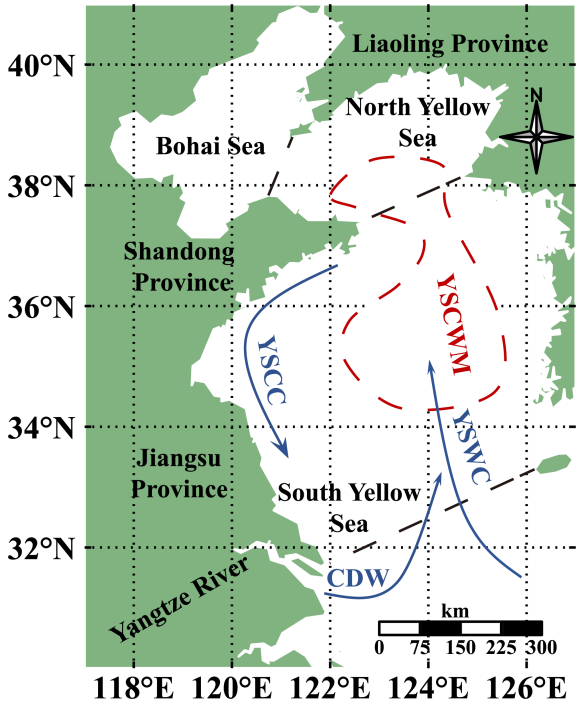


FIGURE 1
Chart of the study region. The three black dashed lines represent the boundaries between the North Yellow Sea (NYS) and Bohai Sea, the NYS and South Yellow Sea (SYS), and the SYS and East China Sea (ECS).

300 × 10^3 km^2 and 44 m, respectively. The climate and ocean circulations exhibit strong seasonality due to the effect of the East Asian Monsoon (Ding et al., 2018). In the winter, the YS is mainly influenced by the Yellow Sea Warm Current (YSWC) and the Yellow Sea Coastal Current. The Yellow Sea Warm Current invades the YS from south to north, and brings warm ocean water to the YS, which makes some regions into carbon sources in the YS (Xue et al., 2011). In the summer, the central YS is occupied by the Yellow Sea Cold Water Mass (YSCWM) and there is a strong thermocline above the YSCWM. In addition, the northeastern extension of the Changjiang Dilution Water (CDW) carries a considerable amount of nutrients to the west of the SYS, and this region sustains high phytoplankton production, thereby leading to lower pCO_2 values (Qu et al., 2014). Overall, the YS current is an important factor that affects pCO_2. A previous study showed that the coastal currents in the YS have strengthened in recent years (Liu S, et al., 2023), which may affect the interannual variation in the pCO_2 in the YS.

The YS is surrounded by rapidly developing economic regions, and the rapid development of mariculture has caused severe environmental problems, such as phytoplankton blooms and changes in ocean acidification. Therefore, the carbon cycle process in the YS is managed by both the coastal hydrodynamics and human activities (Choi et al., 2019).

2.2 Data sets

We collected fugacity of CO_2 (fCO_2) data from 14 cruises conducted between 2011 and 2019, which homogenously covered the entire annual cycle (Table 1). Data were derived from four cruises conducted in 2019 by Yu et al. (2022), and data collected from 10 other cruises by Wang and Zhai (2021).

fCO_2 was conversed into pCO_2 using the following formula (1):

$$fCO_2 = pCO_2 \cdot \exp\left(p \cdot \frac{B + 2\sigma}{RT}\right) \quad (1)$$

where p is the total pressure (Pa), R is a gas constant (8.314 J K^{-1} mol^{-1}), T is the absolute temperature of the sea surface (K), and B and σ are rectification coefficients, which are calculated with formulas (2) and (3).

$$B = (-1636.75 + 12.0408 \times T - 3.27957 \times 10^{-2}T^2 + 3.16528 \times 10^{-5}T^3) \times 10^{-6} \quad (2)$$

$$\sigma = (57.7 - 0.118T) \times 10^{-6} \quad (3)$$

The inverse model of pCO_2 in the YS was established with Chl, SST, SSS, and Kd as input variables. In addition, Julday (Jday, or day of year) was selected as an input to highlight the periodical changes in pCO_2 (Lefevre et al., 2005; Signorini et al., 2013). Chl and Kd, SST, and SSS were used to represent biochemical, thermodynamic, and mixing effects on the sea surface pCO_2, respectively. Level 3 8-days and monthly SST (°C), Chl (mg m^{-3}), and Kd (m^{-1}) data sets were obtained from Moderate Resolution Imaging Spectroradiometer (MODIS)-Aqua for January 2003 and December 2021 (https://oceancolor.gsfc.nasa.gov/) at a spatial resolution of 4 km. SSS data observed directly by ocean color sensor satellites are not available, so *in-situ* SSS data were used to develop the model in this study. The HYbrid Coordinate Ocean Model (HYCOM) SSS data set (monthly products with a 4-km resolution) was selected to derive maps of the sea surface pCO_2 (available from: https://www.hycom.org/). In addition, the gridded atmospheric pCO_2 (pCO_2^{air}) data set (daily, with a spatial resolution of 2° × 2.5°) provided by Rödenbeck et al. (2013) was used (available from: http://www.bgc-jena.mpg.de/SOCOM/).

Due to the influence of cloud cover, sensor technology, atmospheric correction algorithms, and other factors, satellite remote sensing data have a high missing rate in time and space. Therefore, satellite data were interpolated using Data Interpolating Empirical Orthogonal Functions (DINEOF) to obtain more matching pairs. A pixel located at 122°E and 33.2°N was selected to verify the rationality of the reconstructed data. The reconstructions agreed with the original data and complemented the missing data well (Figure 2).

Satellite data were matched with *in-situ* data according to (Le et al., 2019). Briefly, a time window of ± 8 days was applied between the *in-situ* and satellite-derived data. In addition, in order to filter sensor and algorithm noise, the median of a 3 × 3-pixel box was focused on every sample point. If the coefficient of variation for the effective pixels in the 3 × 3-pixel box was ≤ 0.4, the extracted data were used to develop the model together with the *in-situ* data. Finally, we obtained 638 matched pairs from 14 cruises (Figure 3).

2.3 Model training and testing, and model selection

The 638 matched pairs were split into training and test data sets in a stratified random manner, where they accounted for 80% and 20% of the pairs, respectively. Histograms showing the

TABLE 1 Comparison of two empirical modeling approaches. .

Approach	RMSD (μatm)	R^2	MAE (μatm)	MAPE
PSO-SVR	43	0.63	35	9%
	54	0.44	40	11%
RF	**34**	**0.82**	**24**	**6%**
	43	**0.67**	**32**	**8%**

10.3389/fmars.2023.1181095

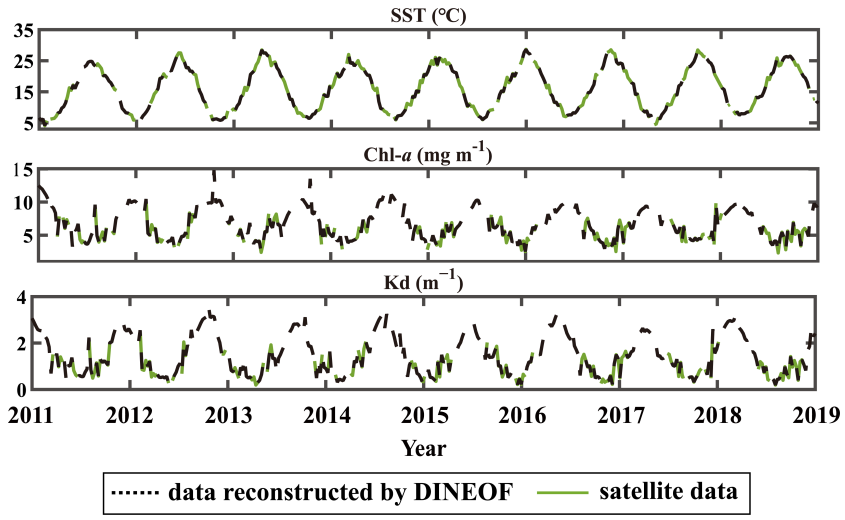


FIGURE 2
Comparison of reconstructed and original data.

distributions of the sample points in the training and test data sets are presented in Figure 4. Evaluation indicators comprising the RMSD, coefficient of determination (R^2), mean absolute error (MAE), and mean absolute percentage error (MAPE) were employed to quantify the reliability of the pCO_2 model.

Two machine learning algorithms comprising Random Forest (RF) and particle swarm optimization-support vector regression (PSO-SVR) were used to develop sea surface pCO_2 models because of their high generalizability for nonlinear relationships (Mountrakis et al., 2011). The inversion model was established using identical data sets. The algorithm was determined as formula (4).

$$pCO_2 = f(\text{input variables})$$
$$= f(\text{SST, Kd, SSS, Chl, } \cos(2\pi(\text{Julday} - \gamma)/365) \quad (4)$$

The value of γ was optimized iteratively (0 to 365) until the RMSD reached a minimum value.

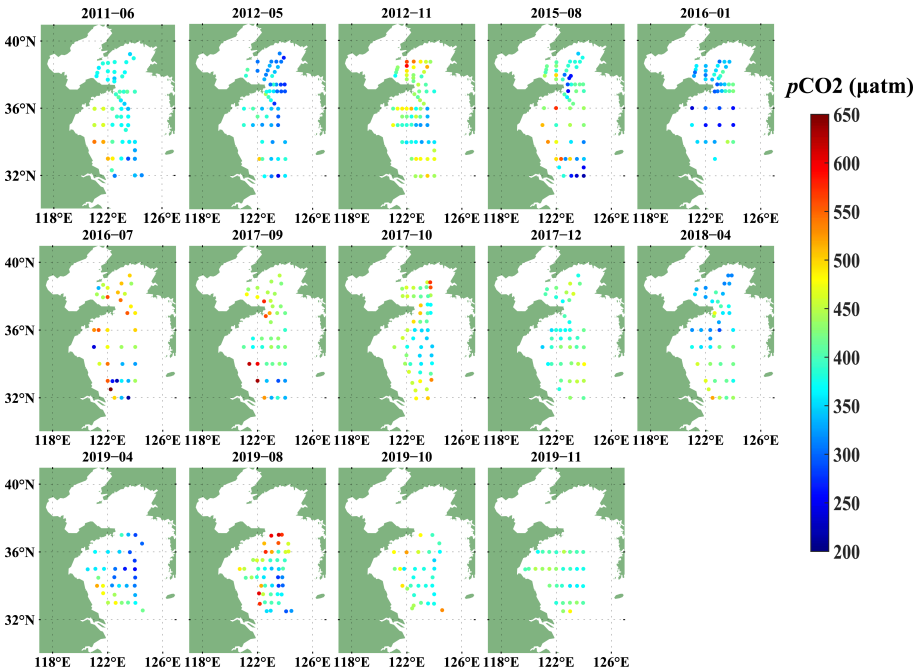


FIGURE 3
Spatial distribution of 638 matched pairs.

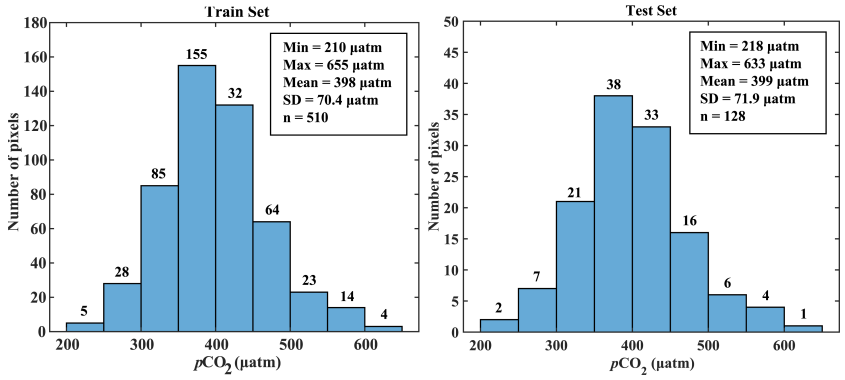


FIGURE 4
Histograms showing the distributions of the sample points in the training and test data sets.

2.4 Random forest

The RF consists of multiple decision trees, where the structure of a single decision tree is based on a group of training data (Breiman, 2001). In RF, a bootstrap strategy is used to conduct resampling from the original data sets to produce multiple subgroups. The structure regression trees are then obtained for every subgroup, and the final output is the mean of the outputs of all regression trees.

RF model development (Figure 5) requires the determination of three customized parameters: the number of randomly selected variables for constructing the tree (mtry), the minimum number of terminal nodes for each tree (node size), and the number of trees (ntree) (Sun et al., 2016).

The node size was set to 5 because this is a common value for regression models (Sun et al., 2016). The grid search method was used to determine the RF parameters ntree and mtry (Figure 6). The optimal values were determined with the minimal RMSD, and 4 and 200 were selected as the best mtry and ntree values, respectively, for the RF model.

2.5 Model sensitivity to input variables

Sensitivity analysis was conducted to assess the sensitivity of the model to the inherent uncertainties in SST, SSS, Chl, and Kd. The original pCO_2 (using the original inputs) was compared with the new pCO_2 (using inputs with extra added uncertainties) derived from the same RF model to identify the model's sensitivity to the uncertainty in these inputs. Only one input variable was changed in each analysis and the remaining variables were kept the same. Statistical parameters comprising the mean bias (MB), mean ratio (MR), RMSD, and R^2 were applied to quantify the sensitivity.

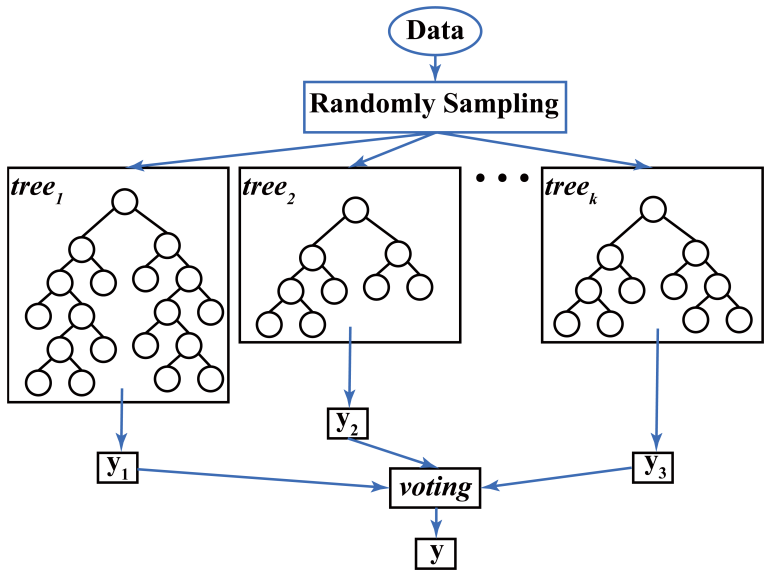


FIGURE 5
General Random Forest model development process.

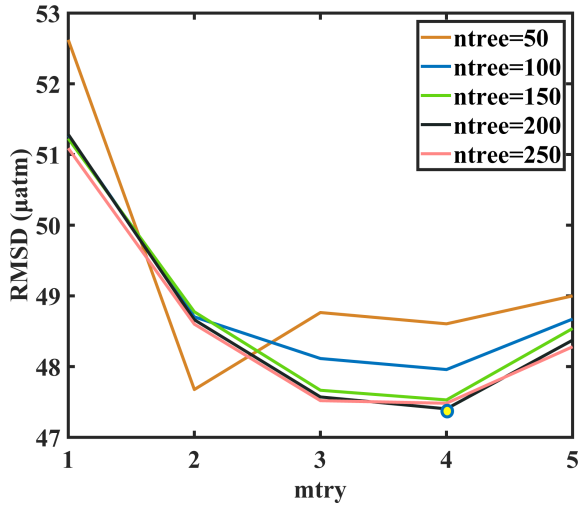


FIGURE 6
Influence of mtry and ntree on RMSD.

The uncertainties of environmental variables were determined by referring to published studies. In particular, the uncertainty of remote sensing SST is ≤ 1°C (Hao et al., 2017), the uncertainty of HYCOM SSS is about 0.5 when SSS is more than 32, the uncertainty of HYCOM SSS is about 3 when SSS is less than 32 (Jang et al., 2022), and the uncertainties of Chl and Kd are 32% and 48%, respectively (Cui et al., 2014). Thus, we used ± 1°C, ± 1, ± 30%, and ± 45% as the uncertainties of SST, SSS, Chl, and Kd, respectively.

3 Results

3.1 Model performance

Table 1 shows that RF outperformed PSO-SVR. The R^2 and RMSD values were 0.82 and 34 µatm, and 0.67 and 43 µatm for the model training and test data sets, respectively.

The sea surface pCO_2 predicted by the RF model was slightly underestimated when the sea surface pCO_2 was larger than 500 µatm, and slightly overestimated when pCO_2 was smaller than 300 µatm (Figure 7). The pCO_2 values estimated by the model varied in the range of 250–550 µatm, with some larger than 550 µatm and lower than 250 µatm. A histogram showing the residuals (modeled pCO_2 minus field pCO_2) is presented in Figure 7, which demonstrates that 82.45% of the residuals were within the interval of ± 50, i.e., the observed 50 µatm pCO_2 standard deviation.

3.2 Model sensitivity

Statistically, when a bias of +1°C was applied to the SST input, the RF model overestimated the sea surface pCO_2 slightly (RMSD = 10 µatm, R^2 = 0.96, MB = 3 µatm), and when a bias of −1°C was applied to the SST input, the RF model underestimated the sea surface pCO_2 slightly (R^2 = 0.96, RMSD = 10 µatm, MB = −2 µatm) (Figure 8). These results suggest that pCO_2 increased with SST, and vice versa, which is consistent with the relationship between temperature and pCO_2 in thermodynamics.

Compared with the SST, the RF pCO_2 model was more sensitive to the uncertainties in SSS. Moreover, the RF model was more sensitive to lower SSS values, where a change of −1 in SSS resulted in a substantial decrease in the predicted pCO_2. In particular, with input +1 uncertainty in SSS, the RF pCO_2 model tended to overestimate the sea surface pCO_2 (R^2 = 0.83, RMSD = 20 µatm, and MB = 5 µatm) and with input −1 uncertainty in SSS, the RF pCO_2 model tended to greatly underestimate the sea surface pCO_2 (R^2 = 0.73, RMSD = 30 µatm, and MB = −16 µatm).

Similar to SST, the RF pCO_2 model exhibited minor sensitivity to Chl. When all data were used in the calculations with +30% uncertainties added, the RF model slightly overestimated pCO_2 (R^2 = 0.96, RMSD = 10 µatm, and MB = 2 µatm). With input −30% uncertainties in Chl, the RF model slightly underestimated pCO_2 (R^2 = 0.95, RMSD = 11 µatm, and MB = −3 µatm). Similarly, the RF pCO_2 model was insensitive to Kd. With +45% and −45% uncertainties added in Kd, the new pCO_2 was not very different from the original pCO_2. In particular, with a bias of +45% uncertainty added to Kd, the RF slightly overestimated the surface pCO_2 (R^2 = 0.93, RMSD = 16 µatm, and MB = 9 µatm), and with a bias of −45% uncertainty added, the RF pCO_2 model slightly underestimated the pCO_2 (R^2 = 0.89, RMSD = 18 µatm, and MB= −8 µatm).

The sensitivity of the RF model was different according to the uncertainty in each environment variable, but the differences introduced by each variable were generally within the range of the uncertainty of the model itself.

3.3 Seasonal and interannual variations in pCO_2 in the YS

The RF model was applied to monthly MODIS and HYCOM data for the period between January 2003 and December 2021 to generate monthly climatological maps and determine the annual trend in pCO_2 in the YS (Figure 9).

Spatially, due to the effects of the hydrology environment and terrestrial organic matter, the pCO_2 values tended to decrease from the nearshore to central areas, and the highest pCO_2 values were observed in the SYS. Seasonally, there were apparent variations in pCO_2 throughout the YS (Figure 9). Statistically, the average sea surface pCO_2 values were 377 ± 7 µatm, 430 ± 6 µatm, 426 ± 11 µatm, and 378 ± 10 µatm in the spring, summer, autumn, and winter, respectively. In addition to these seasonal patterns, more complex variations were found in the spring and autumn (Figure S1). In most years, pCO_2 decreased in March because of phytoplankton blooms, and increased in September or November because of the collapsing seasonal stratification.

The annual mean sea surface pCO_2 values were extracted to explore the interannual variation. The results showed that the surface pCO_2 values in the YS increased between 2003 and 2021 at a rate of 0.36 µatm $year^{-1}$ (R^2 = 0.27, p< 0.05, N = 19) (Figure 10).

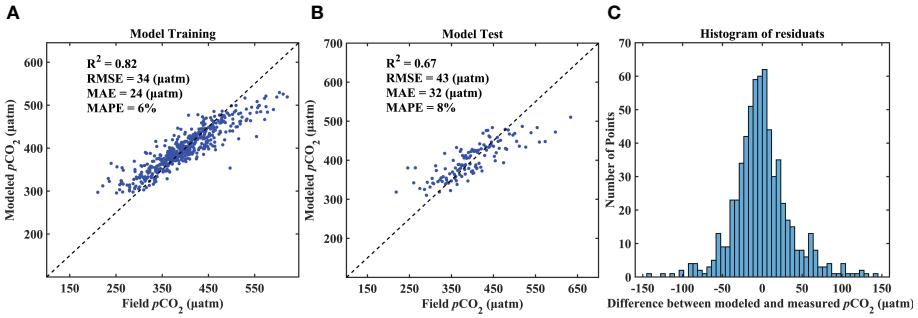


FIGURE 7
Performance evaluation for RF using **(A)** training and **(B)** test data sets; and **(C)** histogram of residuals.

According to the model sensitivity analysis results in section 3.2, when a bias of +1°C was applied to the SST input, the RF model overestimated pCO_2 by 10 µatm. The annual rate of change in the SST determined by the remote sensing products was 0.039°C year^{-1} (Figure S2). Therefore, increasing the SST approximately led to an increase in the pCO_2 at a rate of 0.39 µatm year^{-1} in the YS. The pCO_2 in the YS has increased in the past 19 years, but its rate of increase was lower than that for pCO_2^{air} (with a rate of 2.31 µatm year^{-1}; R^2 = 0.99, p< 0.01, N = 19) in the same period (Figure S3). Therefore, the ΔpCO_2 (sea surface pCO_2– pCO_2^{air}) exhibited a remarkable decreasing trend with a rate of −1.95 µatm year^{-1} (R^2 = 0.92, p< 0.01, N = 19).

Moreover, the spatial trends in pCO_2 were obtained by calculating the trend for each grid in pCO_2 (Figure 10B). In general, pCO_2 increased in most regions of the YS, with a range from 0 to 2.78 µatm year^{-1} from 2003 to 2021. Decreasing trends were also found in some regions. For example, pCO_2 decreased in the NYS and the runoff area of the Changjiang River. These results indicate that the NYS and runoff area of the Changjiang River have more substantial carbon absorption capacities. Both pCO_2 and Chl tended to decrease in the runoff area of the Changjiang River (Figures 10B, S4). Therefore, the decrease in the transportation of terrestrial organic matter might be the main reason for the decrease in pCO_2 in this area, which might alleviate the seasonal hypoxia phenomenon.

4 Discussion

4.1 Evaluation based on comparisons with field observations of sea surface pCO_2

Two algorithms were tested to establish models for estimating pCO_2. The best RMSD and R^2 values for the model were 43 µatm and 0.67 in the YS, respectively (Figure 7). The accuracy of four data sets were evaluated by comparing with field observations of sea surface pCO_2. The resolutions, names of the four data sets, and comparisons of the results are shown in Table 2.

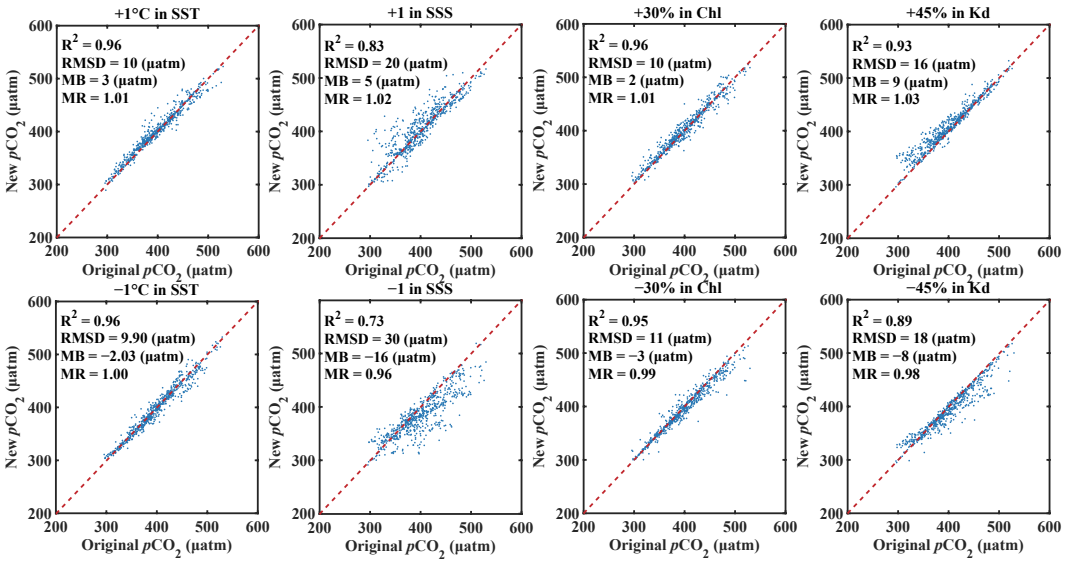


FIGURE 8
Sensitivity of RF model to the uncertainties in SST, SSS, Chl, and Kd.

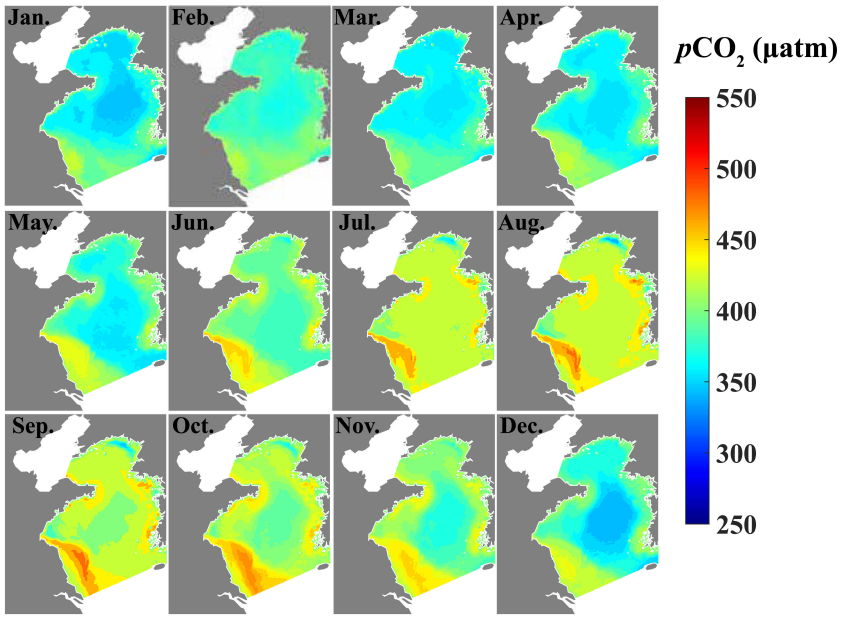


FIGURE 9
Monthly climatological maps of pCO_2 in the YS from January 2003 to December 2021.

Figure 11 shows scatter diagrams to compare the results. The pCO_2 derived from the RF model exhibited greater consistency (RMSD = 55 μatm) with the *in-situ* pCO_2 than CSIR-ML6 (RMSD = 71 μatm), MPI-SOMFNN (RMSD = 82 μatm), and $SatCO_2$ (RMSD = 119 μatm). The significant underestimation of the field pCO_2 by $SatCO_2$ was predictable because the algorithm was originally developed for the ECS and it may not be applicable to the YS. Significant differences between the global pCO_2 products and *in-situ* data in coastal seas were expected (Landschützer et al., 2020). Moreover, CSIR and ML6 were not effective at matching the pCO_2 in the YS, as shown by the number of scatter points in Figure 11. The comparison of four products showed that the RF model was the optimal method for estimating pCO_2 in the YS because the root mean square difference was less than those with the other three products (CSIR-ML6, MPI-SOMFNN, and $SatCO_2$). Understanding the variations in pCO_2 can provide greater insights into the response of the carbon absorption capacity to climate change in the YS. Erroneous estimates may be obtained in coastal seas if global pCO_2 products are used, which might affect quantification of the longer-term trends in global carbon budgets.

4.2 Satellite estimation of pCO_2 in coastal seas

Due to its unique advantage in terms of high spatiotemporal resolution, satellite remote sensing is an effective method for observing the sea surface pCO_2. Table 2 lists some inversion models for pCO_2 in coastal seas. The maximum RMSD for these models was 45.19 μatm. Tao et al. (2012) and Qin et al. (2014)

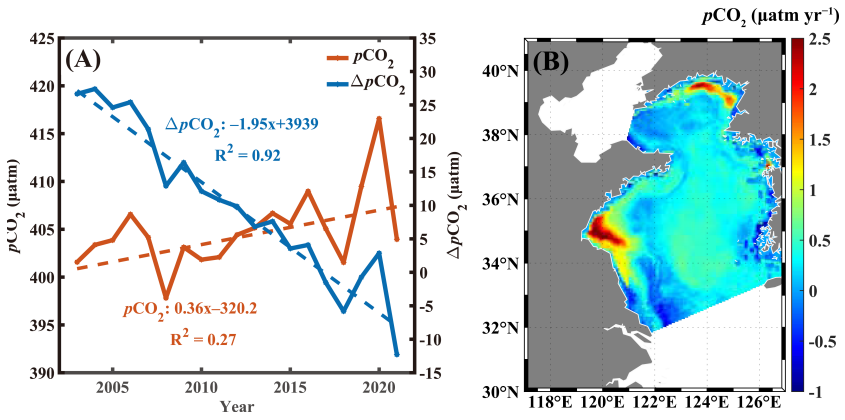


FIGURE 10
(A) Long-term trends in regional average pCO_2 and ΔpCO_2 ($pCO_2 - pCO_2^{air}$); and **(B)** spatial trends in pCO_2 during 2003–2021.

TABLE 2 Published models based on remote sensing of sea surface pCO_2 and global pCO_2 products.

Reference	Model or data set	Study area	Spatial resolution/Model inputs	RMSD (H atm)
Gregor et al. (2019)	CSIR-ML6	Yellow Sea	1° x1°	71
Landschützer et al. (2016)	MPI-SOMFNN	Yellow Sea	1° x1°	82
Bai et al. (2015)	$SatCO_2$	Yellow Sea	1.6 km	119
this study	RF	Yellow Sea	4 km	55
Parard et al. (2014)	SOM	Baltic Sea	SST. Chl. CDOM, NPP, MLD. Jday	35
Tao et al. (2012)	MPR	Yellow Sea and Bohai Sea	SST. Chl	31.74
Qin et al. (2014)	MPR	Yellow Sea	SST. Chl	16.68–21.46
Chen et al. (2016)	MNR	West Florida Shelf	SST. Kd. Chl. Iday	<11.79
Liu J, et al. (2023)	MNR	East China Sea	SST. SSS, Chl. Jday, LAT. LON	3.73-45.19

SOM, Self Organizing Map; MNR, Multi-variate Nonlinear Regression; NPP, Net Primary Production; CDOM, Colored Dissolved Organic Matter; LAT, Latitude; LON, Longitude.

established pCO_2 estimation models based on MPR using the *in-situ* SST and Chl, and the RMSD values for the two models were 15.82 −31.7 and 16.68–21.46, respectively, and both were less than 43. The error was small for the two models, mainly because the *in-situ* data used for modeling were mostly located in the YS center, with few data located in the nearshore area. The MPR-based inversion model was developed using the same training data sets employed in the present study, and the error was much larger than 43 µatm. Overall, the error was acceptable for the RF model developed in this study. The RMSD of the model for estimating the surface pCO_2 in the YS was higher than that in other marginal seas due to the following three reasons. (1) The uncertainty of satellite data and field pCO_2. In the YS, the error of satellite remote sensing Kd and Chl data can reach 48%, and 32%, respectively (Cui et al., 2014). Moreover, the pCO_2 data used in this study were converted from fCO_2, and fCO_2 was estimated using the dissolved inorganic carbon and total alkalinity. The uncertainty in the pCO_2 obtained by using this method is ± 5%, which is larger compared with ± 1% using directly measured pCO_2 data (Wang and Zhai, 2021). (2) The hydrological complexity of the YS environment leads to a wide range of sea

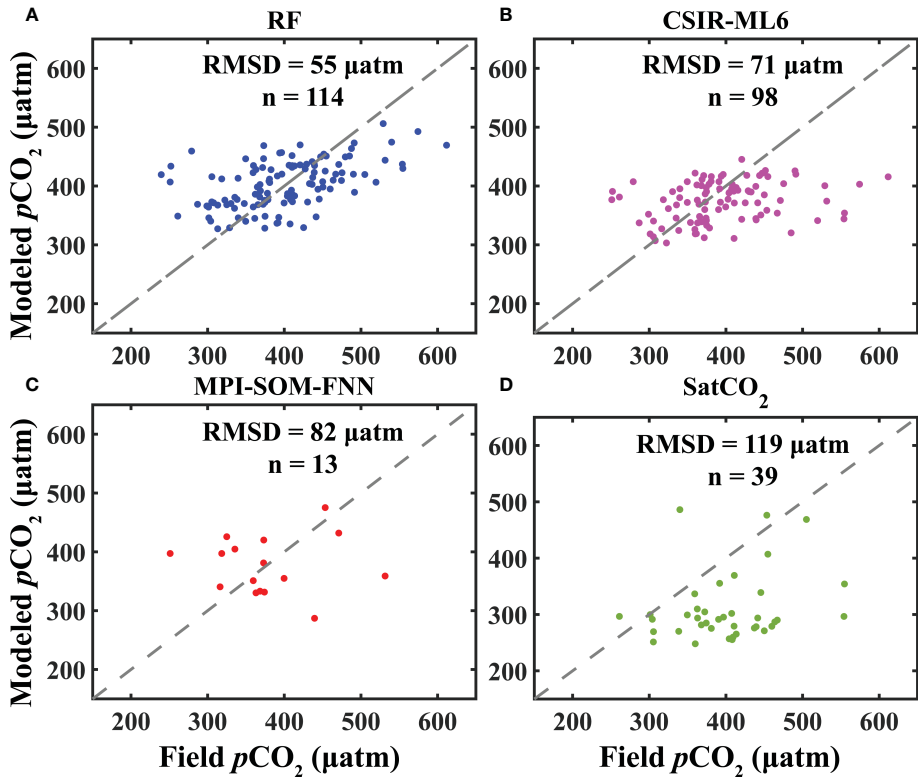


FIGURE 11
Scatter plots of pCO_2 obtained from **(A)** RF model, **(B)** CSIR-ML6, and **(C)** MPI-SOMFNN; and **(D)** $SatCO_2$ against the field pCO_2 in the test set.

surface pCO_2 changes. In particular, the magnitude of the change in pCO_2 in the YS is 450 µatm (Figure 3), but only about 350 µatm in the Gulf of Mexico (Fu et al., 2020) and the Gulf of Maine (Signorini et al., 2013). The performance of the model constructed for the YS was similar to that of a model for the Baltic Sea (RMSD = 47.48 µatm, R^2 = 0.63) (Zhang et al., 2021), where pCO_2 ranged from 100 −600 µatm. (3) Importantly, the RF model needed to include all of the processes from 2011 to 2019. These three reasons explain why estimating pCO_2 is very difficult in the YS compared with other marginal seas, and thus the error is large.

4.3 Advantages and limitations of RF model

The comparisons of the models based on the two algorithms showed that the RF algorithm was advantageous for inverting the sea surface pCO_2 in the YS (Table 1; Figure 11), and the uncertainty was less than 50 µatm. However, the RF model still has some problems.

First, in the eastern YS, the seasonal variation in the pCO_2 obtained from the RF model differed compared with the *in-situ* pCO_2. Choi et al. (2019) found that pCO_2 tended to increase from May to February in the Southeastern YS. However, the maximum pCO_2 obtained by RF inversion was in August (Figure 9). Wang and Zhai (2021) divided the YS region west of 124°E into four regions and analyzed the seasonal variations in the pCO_2. They found that the maximum values in the four regions occurred in July, September, or October, with none in February. Due to the effect of hydrodynamics and other factors, the seasonal patterns in the pCO_2 differ greatly in the eastern YS and western YS. Therefore, the differences in the seasonal variations in pCO_2 may be explained by only using *in-situ* data for the area located west of 124°E for modeling, and thus the model was unable to fully identify the pCO_2 control process.

Second, using the RF model to compute the interannual trends in the pCO_2 could introduce uncertainties. The homogenously collected cruise data covered the whole annual period (Table 3). The variation in pCO_2 was influenced by physical and biogeochemical processes in the sea, and the increase in atmospheric CO_2 (Xue et al., 2016). However, the parameters (SST, Chl, Kd, and SSS) used in this study could only characterize the physical and biogeochemical processes in the sea. If changes in pCO_2 caused by increases in the atmospheric CO_2 could not be captured implicitly by one or more of the four parameters (SST, SSS, Chl, and Kd), uncertainties would be introduced when computing the interannual trend in the pCO_2 (Chen et al., 2019). The long-term trend of SST in the YS was influenced by regional climate change (Park et al., 2015), that is to say, the change of SST included the change of atmospheric CO_2 internally and implicitly, therefore, the increase in the SST appeared to can capture the effects of increasing atmospheric CO_2 on the pCO_2, the interannual trend was still credible to some extent.

Third, in the present study, RF performed poorly at simulating data from both ends of the data sets (underestimation for high values and overestimation for low values) (Figure 7), which may be explained as follows. First, due to the features of the algorithm itself, RF averages the results for all regression trees. The underestimation of extreme values and overestimation of small values appears to be a common problem for RF regression models (Čeh et al., 2018; Zimmerman et al., 2018; Wolfensberger et al., 2021). Second, the training data sets contained very few extreme pCO_2 values and they were underrepresented in the RF model, thereby leading to a more mean-biased output from the RF model.

In general, the problems with the RF model described above were caused by the unbalanced distributions of the modeling data sets. The number of extreme pCO_2 values (>550 µatm or<250 µatm) was relatively small in the field measurements (only 4.7%) but it did not seem to affect the interannual variation in the pCO_2. However, extreme pCO_2 is an influential component of the carbon cycle and it has significant impacts on the health of marine ecosystems. Therefore, it is very necessary to accurately estimate the extreme pCO_2. The crucial limitation of RF model is that its ability to estimate new pCO_2 is limited by the range of the training data set. That mean it can not estimate the pCO_2 beyond the range of the training data set (no extrapolation). Therefore, a better RF model may be developed by using a data set with

TABLE 3 Cruises and statistics for SST, SSS, and sea surface pCO_2 measurements used for model training and test (mean ± standard deviation).

Season	Time	SST (°C)	SSS	pCO_2 (µatm)	Number of observations
Spring	2012–05 2018–04 2019–04	10.4 ± 2.9	32.1 ± 0.8	361 ± 58	133
Summer	2011–06 2015–08 2016–07 2019–08	23.0 ± 3.7	31.1 ± 1.1	410 ± 88	204
Autumn	2012–11 2017–09 2017–10 2019–10 2019–11	19.3 ± 3.7	31.5 ± 0.5	425 ± 58	231
Winter	2016–01 2017–12	8.6 ± 3.1	32.2 ± 0.3	373 ± 51	92
average/Total samples	—	17.2 ± 6.6	31.6 ± 0.9	400 ± 73	660

a wider range of variation, which can improve the reproducibility of the RF model for extreme values. Therefore, we suggest that the modeling data set need to include all pCO_2 values that can be matched to the satellite data, some extreme values in the *in-situ* data sets should not be arbitrarily deleted (excluding the low and high values caused by measurement errors).

5 Conclusions

In this study, we constructed a RF model of the YS with SST, SSS, Chl, Kd, and Julday as the inputs. The RF model performed well at estimating pCO_2, with an RMSD of 43 µatm and R^2 of 0.67. The RF model was applied to satellite data from between 2003 and 2021 to obtain a 19-year time sequence of pCO_2 in the YS. Spatially, except for the eastern YS, the spatial pCO_2 distributions derived by the RF model matched with the *in-situ* data. According to the interannual changes, the sea surface pCO_2 increased in most regions of the YS, but there were differences among the regions, with decreased trends in the pCO_2 in the NYS and the runoff area of the Changjiang River, which appears to contrast with the background global warming and increasing atmospheric CO_2 concentration. The present study is the first to using machine learning methods to estimate the pCO_2, and also the first to determine the long-term trend in the pCO_2 in the YS. Future research should focus on obtaining balanced *in-situ* pCO_2 data and coupling the RF model with a mechanistic model to develop more accurate pCO_2 models. In addition, the reasons for the increasing trend in the pCO_2 in the YS should be explored.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author.

Author contributions

WL: Methodology, Software, Writing-original draft. CL: Conceptualization, Resources, Writing-review & editing. WZ: Investigation, Writing-review & editing. HL: Software, Writing-review & editing. WM: Formal analysis. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the following research grants: the National Natural Science Foundation of China-Shandong joint fund (U1806203), Shandong Provincial Natural Science Foundation (ZR2020MD098), Shandong Universities Interdisciplinary Research and Innovation Team of Young Scholars (2020QNQT20).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1181095/full#supplementary-material

References

Bai, Y., Cai, W. J., He, X. Q., Zhai, W. D., Pan, D. L., Dai, M. H., et al. (2015). A mechanistic semi-analytical method for remotely sensing sea surface pCO_2 in river-dominated coastal oceans: a case study from the East China Sea. *J. Geophys. Res. Ocean.* 120, 2331–2349. doi: 10.1002/2014JC010632

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/A:1010933404324

Čeh, M., Kilibarda, M., Lisec, A., and Bajat, B. (2018). Estimating the performance of random forest versus multiple regression for predicting prices of the apartments. *ISPRS. Int. J. Geo-Inf.* 7, 168. doi: 10.3390/ijgi7050168

Chen, S. L., Hu, C. M., Barnes, B. B., Wanninkhof, R., Cai, W. J., Barbero, L., et al. (2019). A machine learning approach to estimate surface ocean pCO_2 from satellite measurements. *Remote Sens. Environ.* 228, 203–226. doi: 10.1016/j.rse.2019.04.019

Chen, S. L., Hu, C. M., Byrne, R. H., Robbins, L. L., and Yang, B. (2016). Remote estimation of surface pCO_2 on the West Florida shelf. *Cont. Shelf. Res.* 128, 10–25. doi: 10.1016/j.csr.2016.09.004

Chen, S. L., Hu, C. M., Cai, W. J., and Yang, B. (2017). Estimating surface pCO_2 in the northern gulf of Mexico: which remote sensing model to use? *Cont. Shelf. Res.* 151, 94–110. doi: 10.1016/j.csr.2017.10.013

Choi, Y., Kim, D., Cho, S., and Kim, T. W. (2019). Southeastern yellow Sea as a sink for atmospheric carbon dioxide. *Mar. pollut. Bull.* 149, 110550. doi: 10.1016/j.marpolbul.2019.110550

Cui, T. W., Zhang, J., Tang, J. W., Sathyendranath, S., Groom, S., Ma, Y., et al. (2014). Assessment of satellite ocean color products of MERIS, MODIS and SeaWiFS along the East China coast (in the yellow Sea and East China Sea). *ISPRS-J. Photogramm. Remote Sens.* 87, 137–151. doi: 10.1016/j.isprsjprs.2013.10.013

Dai, M. H., Su, J. Z., Zhao, Y. Y., Hofmann, E. E., Cao, Z. M., Cai, W. J., et al. (2022). Carbon fluxes in the coastal ocean: synthesis, boundary processes, and future trends. *Annu. Rev. Earth Planet. Sci.* 50, 593–626. doi: 10.1146/annurev-earth-032320-090746

Deng, X., Zhang, G. L., Xin, M., Liu, C. Y., and Cai, W. J. (2021). Carbonate chemistry variability in the southern yellow Sea and East China Sea during spring of 2017 and summer of 2018. *Sci. Total. Environ.* 779, 146376. doi: 10.1016/j.scitotenv.2021.146376

Ding, Y., Bao, X. W., Yao, Z. G., Song, D. H., Song, J., Gao, J., et al. (2018). Effect of coastal-trapped waves on the synoptic variations of the yellow Sea warm current during winter. *Cont. Shelf. Res.* 167, 14–31. doi: 10.1016/j.csr.2018.08.003

Friedlingstein, P., Jones, M. W., O'Sullivan, M., Andrew, R. M., Bakker, D. C. E., Hauck, J., et al. (2022). Global carbon budget 2021. *Earth Syst. Sci. Data* 14, 1917–2005. doi: 10.5194/essd-14-1917-2022

Fu, Z. Y., Hu, L. S., Chen, Z. D., Zhang, F., Shi, Z., Hu, B. F., et al. (2020). Estimating spatial and temporal variation in ocean surface pCO_2 in the gulf of Mexico using remote sensing and machine learning techniques. *Sci. Total. Environ.* 745, 140965. doi: 10.1016/j.scitotenv.2020.140965

Gregor, L., Lebehot, A. D., Kok, S., and Scheel Monteiro, P. M. (2019). A comparative assessment of the uncertainties of global surface ocean CO_2 estimates using a machine-learning ensemble (CSIR-ML6 version 2019a) – have we hit the wall? *Geosci. Model. Dev.* 12, 5113–5136. doi: 10.5194/gmd-12-5113-2019

Gu, Y. Y., Katul, G. G., and Cassar, N. (2021). The intensifying role of high wind speeds on air-sea carbon dioxide exchange. *Geophys. Res. Lett.* 48, e2020GL090713. doi: 10.1029/2020GL090713

Hales, B., Strutton, P. G., Saraceno, M., Letelier, R., Takahashi, T., Feely, R., et al. (2012). Satellite-based prediction of pCO_2 in coastal waters of the eastern north pacific. *Prog. Oceanogr.* 103, 1–15. doi: 10.1016/j.pocean.2012.03.001

Hao, Y. L., Cui, T. W., Singh, V. P., Zhang, J., Yu, R. H., and Zhang, Z. L. (2017). Validation of MODIS Sea surface temperature product in the coastal waters of the yellow Sea. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 10, 1667–1680. doi: 10.1109/JSTARS.2017.2651951

Jang, E., Kim, Y. J., Im, J., Park, Y.-G., and Sung, T. (2022). Global sea surface salinity via the synergistic use of SMAP satellite and HYCOM data based on machine learning. *Remote Sens. Environ.* 273, 112980. doi: 10.1016/j.rse.2022.112980

Landschützer, P., Gruber, N., and Bakker, D. C. (2016). Decadal variations and trends of the global ocean carbon sink. *Glob. Biogeochem. Cycle* 30, 1396–1417. doi: 10.1002/2015GB005359

Landschützer, P., Gruber, N., Bakker, D. C. E., and Schuster, U. (2014). Recent variability of the global ocean carbon sink. *Glob. Biogeochem. Cycle* 28, 927–949. doi: 10.1002/2014GB004853

Landschützer, P., Gruber, N., Bakker, D. C. E., Stemmler, I., and Six, K. D. (2018). Strengthening seasonal marine CO_2 variations due to increasing atmospheric CO_2 nature climate change. *Nat. Clim. Chang.* 8, 146–150. doi: 10.1038/s41558-017-0057-x

Landschützer, P., Laruelle, G. G., Roobaert, A., and Regnier, P. (2020). A uniform pCO_2 climatology combining open and coastal oceans. *Earth Syst. Sci. Data* 12, 2537–2553. doi: 10.5194/essd-12-2537-2020

Laruelle, G. G., Cai, W. J., Hu, X. P., Gruber, N., Mackenzie, F. T., and Regnier, P. (2018). Continental shelves as a variable but increasing global sink for atmospheric carbon dioxide. *Nat. Commun.* 9, 1–11. doi: 10.1038/s41467-017-02738-z

Le, C. F., Gao, Y. Y., Cai, W. J., Lehrter, J. C., Bai, Y., and Jiang, Z. P. (2019). Estimating summer sea surface pCO_2 on a river-dominated continental shelf using a satellite-based semi-mechanistic model. *Remote Sens. Environ.* 225, 115–126. doi: 10.1016/j.rse.2019.02.023

Lefevre, N., Watson, A. J., and Watson, A. R. (2005). A comparison of multiple regression and neural network techniques for mapping *in situ* pCO_2 data. *Tellus. Ser. B-Chem. Phys. Meteorol.* 57, 375–384. doi: 10.3402/tellusb.v57i5.16565

Li, C. L., and Zhai, W. D. (2019). Decomposing monthly declines in subsurface-water pH and aragonite saturation state from spring to autumn in the north yellow Sea. *Cont. Shelf. Res.* 185, 37–50. doi: 10.1016/j.csr.2018.11.003

Liu, J., Bellerby, R. G., Zhu, Q., and Ge, J. Z. (2023). Estimation of sea surface pCO_2 and air–sea CO_2 flux in the East China Sea using *in-situ* and satellite data over the period 2000–2016. *Cont. Shelf. Res.* 254, 104879. doi: 10.1016/j.csr.2022.104879

Liu, Q., Dong, X., Chen, J. S., Guo, X. H., Zhang, Z. R., Xu, Y., et al. (2019). Diurnal to interannual variability of sea surface pCO_2 and its controls in a turbid tidal-driven nearshore system in the vicinity of the East China Sea based on buoy observations. *Mar. Chem.* 216, 103690. doi: 10.1016/j.marchem.2019.103690

Liu, S. C., Luo, Z. P., Wang, Y. W., Rao, Q. R., Zhang, X. S., Yu, B., et al. (2023). Interannual variation in winter thermal front to the east of the Shandong peninsula in the yellow Sea. *J. Sea. Res.* 193, 102370. doi: 10.1016/j.seares.2023.102370

Liu, Z. Y., Wei, H., Lozovatsky, I., and Fernando, H. (2009). Late summer stratification, internal waves, and turbulence in the yellow Sea. *J. Mar. Syst.* 77, 459–472. doi: 10.1016/j.jmarsys.2008.11.001

Liu, J. L., Xia, J., Zhuang, M. M., Zhang, J. H., Sun, Y. Q., Tong, Y. C., et al. (2021). Golden seaweed tides accumulated in pyropia aquaculture areas are becoming a normal phenomenon in the yellow Sea of China. *Sci. Total. Environ.* 774, 145726. doi: 10.1016/j.scitotenv.2021.145726

Lohrenz, S. E., Cai, W. J., Chen, F. Z., Chen, X. G., and Tuel, M. (2010). Seasonal variability in air-sea fluxes of CO_2 in a river-influenced coastal margin. *J. Geophys. Res. Ocean.* 115 (C10). doi: 10.1029/2009jc005608

Lu, X. L., Liu, C. L., Niu, Y., and Yu, S. X. (2021). Long-term and regional variability of phytoplankton biomass and its physical oceanographic parameters in the yellow Sea, China. *Estuar. Coast. Shelf. Sci.* 260, 107497. doi: 10.1016/j.ecss.2021.107497

Mignot, A., von Schuckmann, K., Landschützer, P., Gasparin, F., van Gennip, S., Perruche, C., et al. (2022). Decrease in air-sea CO_2 fluxes caused by persistent marine heatwaves. *Nat. Commun.* 13, 4300. doi: 10.1038/s41467-022-31983-0

Mountrakis, G., Im, J., and Ogole, C. (2011). Support vector machines in remote sensing: a review. *ISPRS-J. Photogramm. Remote Sens.* 66, 247–259. doi: 10.1016/j.isprsjprs.2010.11.001

Parard, G., Charantonis, A., and Rutgerson, A. (2014). Remote sensing algorithm for sea surface CO_2 in the Baltic Sea. *Biogeosci. Discuss.* 11, 12255–12294. doi: 10.5194/bgd-11-12255-2014

Park, K.-A., Lee, E.-Y., Chang, E., and Hong, S. (2015). Spatial and temporal variability of sea surface temperature and warming trends in the yellow Sea. *J. Mar. Syst.* 143, 24–38. doi: 10.1016/j.jmarsys.2014.10.013

Qin, B. Y., Tao, Z., Li, Z. W., and Yang, X. F. (2014). Seasonal changes and controlling factors of sea surface pCO_2 in the yellow Sea. *IOP. Conf. Ser.: Earth Environ. Sci.* 17, 012025. doi: 10.1088/1755–1315/17/1/012025

Qu, B. X., Song, J. M., Yuan, H. M., Li, X. G., and Li, N. (2014). Air-sea CO_2 exchange process in the southern yellow Sea in April of 2011, and June, July, October of 2012. *Cont. Shelf. Res.* 80, 8–19. doi: 10.1016/j.csr.2014.02.001

Rödenbeck, C., Keeling, R. F., Bakker, D. C., Metzl, N., Olsen, A., Sabine, C., et al. (2013). Global surface-ocean pCO_2 and sea–air CO_2 flux variability from an observation-driven ocean mixed-layer scheme. *Ocean. Sci.* 9, 193–216. doi: 10.5194/os-9-193-2013

Signorini, S. R., Mannino, A., Najjar, R. G.Jr., Friedrichs, M. A. M., Cai, W. J., Salisbury, J., et al. (2013). Surface ocean pCO_2 seasonality and sea-air CO_2 flux estimates for the north American east coast. *J. Geophys. Res. Ocean.* 118, 5439–5460. doi: 10.1002/jgrc.20369

Sun, H. W., Gui, D. W., Yan, B. W., Liu, Y., Liao, W. H., Zhu, Y., et al. (2016). Assessing the potential of random forest method for estimating solar radiation using air pollution index. *Energy Conv. Manage.* 119, 121–129. doi: 10.1016/j.enconman.2016.04.051

Tao, Z., Qin, B. Y., Li, Z. W., and Yang, X. F. (2012). Satellite observations of the partial pressure of carbon dioxide in the surface water of the huanghai Sea and the bohai Sea. *Acta Oceanol. Sin.* 31, 67–73. doi: 10.1007/s13131-012-0207-y

Wang, S. Y., and Zhai, W. D. (2021). Regional differences in seasonal variation of air–sea CO_2 exchange in the yellow Sea. *Cont. Shelf. Res.* 218, 104393. doi: 10.1016/j.csr.2021.104393

Wolfensberger, D., Gabella, M., Boscacci, M., Germann, U., and Berne, A. (2021). RainForest: a random forest algorithm for quantitative precipitation estimation over Switzerland. *Atmospheric. Measurement. Techniques.* 14, 3169–3193. doi: 10.5194/amt-14-3169-2021

Xiong, T. Q., Wei, Q. S., Zhai, W. D., Li, C. L., Wang, S. Y., Zhang, Y. X., et al. (2020). Comparing subsurface seasonal deoxygenation and acidification in the yellow Sea and northern East China Sea along the north-to-South latitude gradient. *Front. Mar. Sci.* 7, 686. doi: 10.3389/fmars.2020.00686

Xue, L., Cai, W. J., Hu, X. P., Sabine, C., Jones, S., Sutton, A. J., et al. (2016). Sea Surface carbon dioxide at the Georgia time series site, (2006–2007): air–sea flux and controlling processes. *Prog. Oceanogr.* 140, 14–26. doi: 10.1016/j.pocean.2015.09.008

Xue, L., Zhang, L., Cai, W.-J., and Jiang, L.-Q. (2011). Air-sea CO2 fluxes in the southern yellow Sea: an examination of the continental shelf pump hypothesis. *Cont. Shelf. Res.* 31, 1904–1914. doi: 10.1016/j.csr.2011.09.002

Ye, H. J., Tang, S. L., and Morozov, E. (2022). Variability in Sea surface pCO_2 and controlling factors in the bay of Bengal based on buoy observations at 15°N, 90°E. *J. Geophys. Res. Ocean.* 127, e2022JC018477. doi: 10.1029/2022JC018477

Yu, S. Q., Xiong, T. Q., and Zhai, W. D. (2022). Quasi-synchronous accumulation of apparent oxygen utilization and inorganic carbon in the south yellow Sea cold water mass from spring to autumn: the acidification effect and roles of community metabolic processes, water mixing and spring thermal state. *Front. Mar. Sci.* 9, 858871. doi: 10.3389/fmars.2022.858871

Zhai, W. D. (2018). Exploring seasonal acidification in the yellow Sea. *Sci. China Earth Sci.* 61, 647–658. doi: 10.1007/s11430-017-9151-4

Zhai, W. D., Zheng, N., Huo, C., Xu, Y., Zhao, H. D., Li, Y. W., et al. (2014). Subsurface pH and carbonate saturation state of aragonite on the Chinese side of the north yellow Sea: seasonal variations and controls. *Biogeosciences* 11, 1103–1123. doi: 10.5194/bg-11-1103-2014

Zhang, S. P., Rutgersson, A., Philipson, P., and Wallin, M. B. (2021). Remote sensing supported Sea surface pCO_2 estimation and variable analysis in the Baltic Sea. *Remote Sens.* 13, 259. doi: 10.3390/rs13020259

Zimmerman, N., Presto, A. A., Kumar, S. P., Gu, J., Hauryliuk, A., Robinson, E. S., et al. (2018). A machine learning calibration model using random forests to improve sensor performance for lower-cost air quality monitoring. *Atmos. Meas. Tech.* 11, 291–313. doi: 10.5194/amt-11-291-2018

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 18 May 2023
DOI 10.3389/fmars.2023.1150593

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Ning Wang,
Dalian Maritime University, China
Guanying Huo,
Beihang University, China

*CORRESPONDENCE
Hao Zhang
zhanghao@qut.edu.cn

RECEIVED 24 January 2023
ACCEPTED 26 April 2023
PUBLISHED 18 May 2023

CITATION
Zhang H, Gong L, Li X, Liu F and Yin J (2023) An underwater imaging method of enhancement via multi-scale weighted fusion. *Front. Mar. Sci.* 10:1150593. doi: 10.3389/fmars.2023.1150593

COPYRIGHT
© 2023 Zhang, Gong, Li, Liu and Yin. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

An underwater imaging method of enhancement via multi-scale weighted fusion

Hao Zhang[1]*, Longxiang Gong[2], Xiangchun Li[3], Fei Liu[4] and Jiawei Yin[3]

[1]School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China, [2]Civil Aviation Logistics Technology Co., Ltd, Chengdu, China, [3]Institute of Oceanographic Instrumentation, Qilu University of Technology (Shandong Academy of Sciences), Qingdao, China, [4]School of Optoelectronic Engineering, Xidian University, Xi'an, China

Blurring and color distortion are significant issues in underwater optical imaging, caused by light absorption and scattering impacts in the water medium. This hinders our ability to accurately perceive underwater imagery. Initially, we merge two images and enhance both the brightness and contrast of the secondary images. We also adjust their weights to ensure minimal effects on the image fusion process, particularly on edges, colors, and contrast. To avoid sharp weighting transitions leading to ghost images of low-frequency components, we then propose and use a multi-scale fusion method when reconstructing the images. This method effectively reduces scattering and blurring impacts of water, fixes color distortion, and improves underwater image contrast. The experimental results demonstrate that the image fusion method proposed in this paper effectively improves the fidelity of underwater images in terms of sharpness and color, outperforming the latest underwater imaging methods by comparison in PSNR, Gradient, Entropy, Chroma, AG, UCIQE and UIQM. Moreover, this method positively impacts our visual perception and enhances the quality of the underwater imagery presented.

KEYWORDS

underwater optical imaging, multi-scale weight, image enhancement, image fusion, homomorphic filtering

1 Introduction

The ocean holds vast resources and is considered a new continent to be exploited by mankind. However, rapid population growth, depletion of land resources, and natural environment deterioration has increased the importance of both exploiting and protecting marine resources. In this context, ocean information acquisition, transmission, and processing theory and technology play a critical role in the rational exploitation and utilization of ocean resources. Underwater images are a key source of ocean data and a useful visualization tool for identifying the ocean.

However, compared to the air medium, the attenuation coefficient of light beam propagation in the water medium is much larger, leading to poor underwater imaging quality (Yang et al., 2019). The scattering of light by water and suspended particles also reduces image contrast, resulting in blurred images and poor visibility. Additionally, the attenuation characteristics in water vary with wavelengths of light, with red light being the most attenuated, losing its energy after a distance of 4-5 meters. This makes underwater images more likely to have a bluish or greenish appearance.

These factors collectively limit the quality of underwater imaging, posing significant practical and scientific challenges in the application of underwater images in marine military, marine environmental protection, and marine engineering. Therefore, it is essential to develop effective techniques and technologies to improve underwater imaging quality and overcome these limitations.

The motivation for designing the multi-scale fusion mechanism in underwater image enhancement is to deal with the unique challenges that arise when imaging underwater environments. In particular, underwater images often suffer from severe noise, low contrast, and color distortion, which can reduce visibility and make it difficult to distinguish between different objects in the scene.

One approach to addressing these challenges is to use image enhancement techniques that adjust the brightness, contrast, and color balance of the underwater images. However, standard image enhancement techniques may not be effective in underwater environments due to the complex nature of the underwater light field and the scattering and absorption of light by water and suspended particles.

To overcome these challenges, researchers have developed multi-scale fusion mechanisms that combine information from different scales in the image to improve the overall image quality. This approach involves breaking down the image into different scales and processing each scale separately before fusing the results back together.

By using this multi-scale approach, the low-level features of the image can be enhanced at the pixel level, while the high-level features, such as edges and boundaries, can be preserved to maintain the overall structure of the image. This allows for better visibility and the ability to distinguish between different objects in the scene, making it easier to interpret underwater images for scientific, commercial, and military applications.

With the advancement and maturation of image processing and computer vision technologies (Sahu et al., 2014), many scientists are paying more attention to using these technologies as post-processing steps to enhance the visual quality of underwater images to meet the needs of both human visual characteristics and machine recognition (Guo et al., 2017). Jiang et al. make efforts in both subjective and objective aspects to fully understand the true performance of underwater image enhancement algorithms (Jiang et al., 2022). Image enhancement is a widely-used technique that can be used to improve the quality of underwater photographs by primarily increasing image contrast and correcting color distortion (Wang et al., 2019).

Typical enhancement methods used in this field include histogram equalization (HE) (Hummel, 1977; Pisano et al., 1998), generalized unsharp mask (GUM), and fusion using a monochromatic model (Ancuti et al., 2012). Iqbal et al. developed an Integrated Color Model (ICM) algorithm based on the integrated color model (Iqbal et al., 2007) and an Unsupervised Colour Correction Method (UCM) for underwater image enhancement (Iqbal et al., 2010). Abdul Ghani et al. employed the Rayleigh distribution function to redistribute the input image (Abdul Ghani and Mat Isa, 2015). Huang et al. proposed the RGHS model to enhance image information entropy (Huang et al., 2018). To solve blurriness and color degradation issues, Zhou et al. developed a restoration method based on backscatter pixel prior and color cast removal from the physical point of view of underwater image degradation (Zhou et al., 2022). Peng et al. proposed a depth estimation method for underwater scenes based on image blurriness and light absorption (IBLA), which can be used in the image formation model (IFM) to restore and enhance underwater images (Peng and Cosman, 2017).

For underwater image restoration, a common approach is to analyze the effective degradation model of the underwater imaging mechanism and determine the model parameters based on prior knowledge (Chang et al., 2018). For image defogging, the Dark Channel Prior (DCP) has attracted attention due to the similarity between outdoor and underwater images (Ancuti et al., 2020). Drews-Jr provided a method of Underwater Dark Channel Prior (UDCP) (Drews-Jr et al., 2013) that only considers the G and B channels to produce underwater DCP without taking into account the red channel.

Deep learning-methods have gradually become a research hot spot/highlight as the progress of artificial intelligence technology, such as visual recognition and detection of aquatic animals (Li et al., 2023).Chen et al. constructed a real-time adaptive underwater image restoration method, called GAN-based restoration scheme (GAN-RS) (Chen et al., 2019). Yu et al. developed an underwater image restoration network using an underwater image dataset to simulate the relevant imaging model (Yu et al., 2019). Sun et al. also developed a framework for underwater image enhancement that employs a Markov Decision Process (MDP) for reinforcement learning (Sun et al., 2022). Wang et al. proposed a one-stage CNN detector-based benthonic organism detection (OSCD-BOD) scheme to outperform typical approaches (Chen et al., 2021). Then they summarized on Architectures and algorithms in deep learning techniques for marine object recognition (Wang et al., 2022), especially in organisms (Wang et al., 2023). Li et al. proposed the first comparative learning framework for underwater image enhancement problem beyond training with single reference, namely Underwater Image Enhancement via Comparative Learning (CLUIE-Net), to learn from multiple candidates of enhancement reference (Li et al., 2022). To address the challenges of degraded underwater images, Zhou et al. propose a novel cross domain enhancement network (CVE-Net) that uses high-efficiency feature alignment to utilize neighboring features better (Zhou et al., 2023b). They addressed that most existing deep learning methods utilize a single input end-to-end network structure leading to a single form and content of the extracted features. And they presented a multi-feature underwater image enhancement method via embedded fusion mechanism (MFEF) (Zhou et al., 2023a). To

boost the performance of data-driven approaches, Qi et al. proposed a novel underwater image enhancement network, called Semantic Attention Guided Underwater Image Enhancement (SGUIE-Net), in which we introduce semantic information as high-level guidance across different images that share common semantic regions (Qi et al., 2022).

We summarize our main contributions as follows:

(1) We propose a fusion frame for underwater image enhancement. This frame supplies a basis of different in different scenes for underwater images.

(2) We proposed a multi-scale weighted method, which employed the white balance method to obtain initial enhancement images as references in real conditions, and then filter the compensated images.

(3) We applied Contrast-Limited Adaptive Histogram Equalization (CLAHE) (Pisano et al., 1998) to the L channel in the model in this frame, to reduce time costs and improve efficiency compared to global image equalization.

The following parts are organized as below: Section 2 illustrates the presented method's structure, including color compensation, initial enhancement, image equalization, contrast enhancement, and image fusion. Next, Section 3 explains our experiment and compares the results to those of other methods. Finally, we summarize our method and references, and then discuss its prospects for the future in Section 4.

2 Materials and methods

From the standpoint of image fusion, two different technologies are used on the underwater degraded image to obtain a new image with color brightness and contrast enhancement. The weight maps of the two images are determined and a high-quality image is obtained by weighted fusion.

The original degraded underwater image, based on the Jaffe-McGlamery model, can be expressed based on the follow formula:

$$\boldsymbol{I}(\boldsymbol{x}) = \boldsymbol{J}(\boldsymbol{x})\boldsymbol{e}^{-\eta d(x)} + \boldsymbol{B}_{\infty}(\boldsymbol{X})(\boldsymbol{1} - \boldsymbol{e}^{-\eta d(x)}) \tag{1}$$

where $I(x)$ represents the original image taken underwater, $J(x)$ represents transmissivity, $d(x)$ means observer and object's distance, η gives attenuation coefficient, $B_{\infty}(x)$ refer to color vector.

2.1 Color compensation and initial enhancement

2.1.1 Color channel compensation

Several studies on underwater images have shown that green light attenuates less than red and blue light when propagating underwater, and, as a result, the water body, as well as most captured underwater, is typically blue-green in appearance. Red and blue dual-channel colour compensation is used to solve colour cast (Ancuti et al., 2020), and the images I_r^c and I_b^c after color compensation is obtained as:

$$\boldsymbol{I}_r^c(\boldsymbol{x}) = \boldsymbol{I}_r(\boldsymbol{x}) + \boldsymbol{\alpha} \cdot (\bar{\boldsymbol{I}}_g - \bar{\boldsymbol{I}}_r) \cdot (1 - \boldsymbol{I}_r(\boldsymbol{x})) \cdot \boldsymbol{I}_g(\boldsymbol{x}) \tag{2}$$

$$\boldsymbol{I}_b^c(\boldsymbol{x}) = \boldsymbol{I}_b(\boldsymbol{x}) + \boldsymbol{\alpha} \cdot (\bar{\boldsymbol{I}}_g - \bar{\boldsymbol{I}}_b) \cdot (1 - \boldsymbol{I}_b(\boldsymbol{x})) \cdot \boldsymbol{I}_g(\boldsymbol{x}) \tag{3}$$

where I_r, I_g and I_b represent the red, green and blue colour channels of the initial image $\boldsymbol{I}$, each channel being in the range (0,1), after normalization by the upper limits of their dynamic ranges; and $\bar{\boldsymbol{I}}_r$, $\bar{\boldsymbol{I}}_g$ and $\bar{\boldsymbol{I}}_b$ denoting the average of those channels over the whole image.α is the compensation parameter, and the test shows that α=1 is suitable for a variety of lighting conditions and acquisition settings.

For underwater scenes with limited distortion, in the grayscale world white-balance algorithm achieves good visual performance. In this paper, in the grayscale world this method was applied to calculate the white balance image and obtain the final colour-corrected result by compensating for the loss in both red and blue channel.

2.1.2 Homomorphic filtering

Due to the light limitation of underwater imaging system, the illumination on imaging target is uneven, which deteriorates imaging quality. Homomorphic filtering method was applied for image compensated (Jiao and Xu, 2010). At the end, the image is decomposed into direct irradiation, reflection component, which are then logarithmically transformed as follows:

$$\boldsymbol{lnf}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{lnf}_i(\boldsymbol{x},\boldsymbol{y}) + \boldsymbol{lnf}_r(\boldsymbol{x},\boldsymbol{y}) \tag{4}$$

where $f_i(x,y)$ refer to illumination component, $f_r(x,y)$ represents reflection component, corresponding to high-frequency information. And equation (5) is performed with Fourier transform:

$$\boldsymbol{F}(\boldsymbol{u},\boldsymbol{v}) = \boldsymbol{F}_i(\boldsymbol{u},\boldsymbol{v}) + \boldsymbol{F}_r(\boldsymbol{u},\boldsymbol{v}) \tag{5}$$

In the frequency domain, the different frequency parts of the underwater image are processed based on a Gaussian filter $H(u,v)$, with its transfer function given as below:

$$\boldsymbol{H}(\boldsymbol{u},\boldsymbol{v}) = (\boldsymbol{\gamma}_H - \boldsymbol{\gamma}_L)[1 - \boldsymbol{e}^{\frac{cD^2(u,v)}{D_0^2}}] + \boldsymbol{\gamma}_L \tag{6}$$

where γ_H is the enhanced part in high frequency, γ_L is the reduced part in low frequency, and $D(u,v)$ is referred to the distance of the frequency in the midpoint and (u_0, v_0). D_0 is the value of D when $(u,v) = (0,0)$. The brightness range is compressed to make them average and improve image contrast. The homomorphic filter can appropriately separate the different components. Then, multiply $F(u,v)$ by $H(u,v)$ as follows:

$$\boldsymbol{C}(\boldsymbol{u},\boldsymbol{v}) = \boldsymbol{H}(\boldsymbol{u},\boldsymbol{v})\boldsymbol{F}(\boldsymbol{u},\boldsymbol{v}) \tag{7}$$

The output of homomorphic filter is further performed by the inverse Fourier transform and exponential transformation. The final result $g(x,y)$ is finally given as follows:

$$g(x,y) = \exp\left(c(x,y)\right) = exp(g_i(x,y), g_r(x,y)) \quad (8)$$

where $c(x,y)$ represents the result obtained by inverse Fourier transform, $g_i(x,y)$ is direct illumination component and $g_r(x,y)$ is reflection component. Figure 1 shows the preliminary enhanced image, which was generated using color compensation and homomorphic filtering. When comparing the two-colour images before and after processing, it is clear that colours in three channels of the image are more balanced by using the method described in this paper.

2.2 Image equalization and contrast enhancement

2.2.1 Gamma correction

After color compensation, the white balance algorithm is used to process the original/preliminary enhanced image. The goal of this step is to improve image quality by reducing color shifts because of excessive illumination. However, because the underwater image is often brighter after color compensation and homomorphic filtering, we convert the preliminary enhanced image into HSV space to enlarge the contrast of bright and dark areas. Set the gamma as follow:

$$s = \alpha I_e^{\gamma} \quad (9)$$

Figure 2 depicts the gamma correction curve. When γ<1, the dynamic range of low gray values increases, the image's overall gray value increases. When γ>1, the dynamic range of low gray value shrinks while the high gray value expands. Selecting a value greater than 1 can correct the global contrast in high-brightness underwater images, such as γ=2.2 in our case.

2.2.2 Contrast limited adaptive histogram equalization

The gray values of most underwater images are low and, therefore, their histogram distribution tends to be narrow.

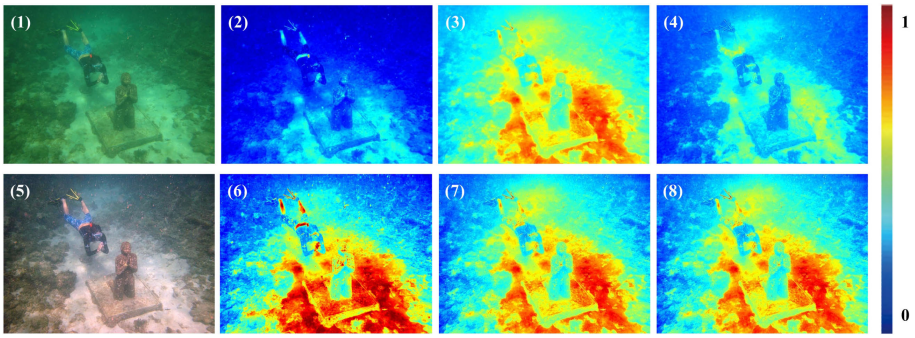


FIGURE 1
Comparison original images with preliminary enhancement images. (1) Original; (2) Channel R; (3) Channel G; (4) Channel B; (5) Colour compensation and homomorphic filtering; (6) Channel R after preliminary enhancement; (7) Channel G after preliminary enhancement; (8) Channel B after preliminary enhancement.

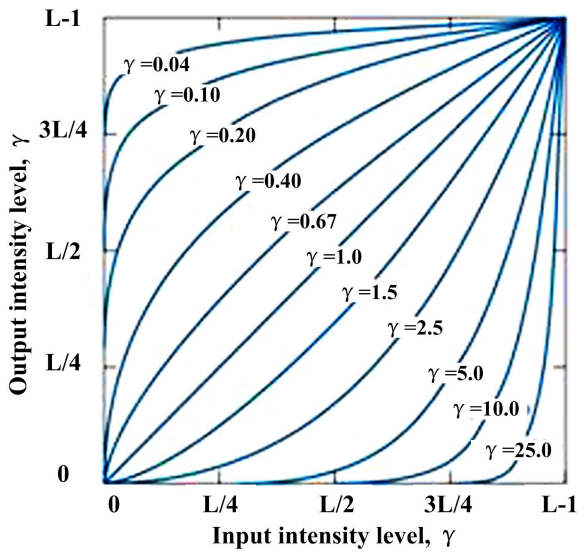


FIGURE 2
Gamma correction curve.

CLAHE can be used to modify the histogram distribution of an underwater image so that correcting colour bias and improving image contrast to some extent (Pisano et al., 1998). In this paper, histogram equalization was performed in LAB space, i.e. the contrast of L component was enhanced separately in LAB space. L stands for brightness in LAB model, while A and B stand for colour. By enhancing the L channel separately, you can avoid impact colour component of image.

The image is divided into several sub-blocks using local histogram equalization, and into limited non-coincidence sub-blocks in this paper. The pixel points in grayscale are then calculated using bilinear interpolation technology to solve the block effect in image reconstruction.

2.3 Image fusion by weight

2.3.1 Define the weight of fusion

After obtaining two fusion input images, we calculate the special weight map of these inputs to reflect high contrast, regions with edge texture change of images. Brightness, local contrast, and saturation of the image are primarily considered in the selection of the weight map in this paper.

To maintain consistency in local image contrast, the brightness weight W_k^E is applied to assess the exposure degree of its pixels, giving higher weights to well situated pixels in brightness. Because the mean natural brightness of an image pixel is typically close to 0.5, the mean experimental brightness is set to 0.5, and its standard deviation is set to 0.25 (Ancuti and Ancuti, 2013), and the brightness weight W_k^E of image $\boldsymbol{I}$:

$$\mathbf{W}_k^E(\mathbf{x},\mathbf{y}) = \mathbf{exp}\left\{-\frac{[\mathbf{L_k}(\mathbf{x},\mathbf{y})-0.5]^2}{2\boldsymbol{\sigma}^2}\right\} \tag{10}$$

The normalized image in grayscale is represented by $L_k(x,y)$.

The average value of its neighboring pixels is represented by the local contrast weight W_k^C. By using local contrast weighting, we can draw attention to the transition area between the light and dark parts. $W_k^C$is the input image's brightness weight, given as follows:

$$\mathbf{W_k^C}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{I_k} - \boldsymbol{I}_{\boldsymbol{\omega hc}}^{\boldsymbol{k}} \tag{11}$$

where I^k refers to the brightness channel of image and $I_{\omega_{hc}}^k$ represents its low-pass part. The low-pass filter uses a separable binomial kernel of 5×5 (1/16 (Pisano et al., 1998; Wang et al., 2019; Wang et al., 2019; Yang et al., 2019; Yang et al., 2019)) with a ω_{hc}=π/2.75. The binomial kernel is very similar to the Gaussian kernel and, as a result, easy to calculate.

We define an input image's saturation weight W_K^{Sat}, which makes the fusion image evenly saturated by adjusting the highly saturated area of the input image, expressed as below:

$$\boldsymbol{W_K^{Sat}}(\boldsymbol{x},\boldsymbol{y}) = \sqrt{[\boldsymbol{R_k}(\boldsymbol{x},\boldsymbol{y})-\boldsymbol{L_k}(\boldsymbol{x},\boldsymbol{y})]^2+[\boldsymbol{G_k}(\boldsymbol{x},\boldsymbol{y})-\boldsymbol{L_k}(\boldsymbol{x},\boldsymbol{y})]^2+[\boldsymbol{B_k}(\boldsymbol{x},\boldsymbol{y})-\boldsymbol{L_k}(\boldsymbol{x},\boldsymbol{y})]^2} \tag{12}$$

where $R_k(x,y)$, $G_k(x,y)$ and $B_k(x,y)$ represent red, green and blue channel respectively. Calculate the normalized weight $\bar{W}_k(x,y)$ by applying normalization to the brightness, local contrast, and saturation weights as:

$$\bar{\boldsymbol{W}}_{\boldsymbol{k}}(\boldsymbol{x},\boldsymbol{y}) = \frac{\boldsymbol{W_k}(\boldsymbol{x},\boldsymbol{y})}{\sum_{\boldsymbol{k}}\boldsymbol{W_k}(\boldsymbol{x},\boldsymbol{y})} \tag{13}$$

$$\boldsymbol{W_k}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{W_k^E}+\boldsymbol{W_k^C}+\boldsymbol{W_K^{Sat}} \tag{14}$$

2.3.2 Multi-scale fusion

The typical intuitive method in this field is to add two weighted images, but which will lead to significant halos. Therefore, the experiment in this paper uses multi-scale fusion technology (Ancuti and Ancuti, 2013), which is developed from the classic multi-scale fusion. The following is a description of fusion computing:

$$\boldsymbol{F_l}(\boldsymbol{x},\boldsymbol{y}) = \sum_{\boldsymbol{k}}\boldsymbol{L_l}[\boldsymbol{I_k}(\boldsymbol{x},\boldsymbol{y})]\boldsymbol{G_l}[\bar{\boldsymbol{W}}_{\boldsymbol{k}}(\boldsymbol{x},\boldsymbol{y})] \tag{15}$$

where l is pyramid decomposition layers number, k is fused images number, $\bar{W}_k(x,y)$ is the normalized weight, $L_l[I_k(x,y)]$ is the Laplacian pyramid decomposition, and $F_l(x,y)$ is the l layer of image pyramid, l=5 and k=2 in this experiment.

$$I_{result} = \sum_{l} Up[F_l(x,y)] \tag{16}$$

where I_{result} denotes final output image; $Up[F_l(x,y)]$ denotes up-sampling.

2.4 Our methodology

This paper proposes a multi-scale fusion-based underwater image enhancement algorithm. Figure 3 depicts the algorithm flow. Colour compensation and colour cast correction are performed on the underwater image in shallow water, and the compensated image is then subjected to complete the preliminary enhancement. To get two fused input images, the enhanced image is subjected to gamma correction equalization. Finally, to achieve the goals of attenuation compensation and contrast and definition improvement, a multi-scale fusion algorithm was used.

The method established in this study outperforms existing underwater image enhancement methods in subjective visual effects and objective evaluation indicators in an experimental comparison of various types of underwater images in shallow water.

3 Experimental results and discussion

We compare the enhancement method set in this article with existing professional algorithms by processing multiple underwater images and summarizing their image visual effects and objective image quality evaluation to verify the effectiveness of i in this section. With reference to our previous work, our experiments with each previous algorithm (Pisano et al., 1998; Iqbal et al., 2010; Drews-Jr et al., 2013; Abdul Ghani and Mat Isa, 2015; Panetta et al., 2015; Yang and Sowmya, 2015; Peng and Cosman, 2017; Huang

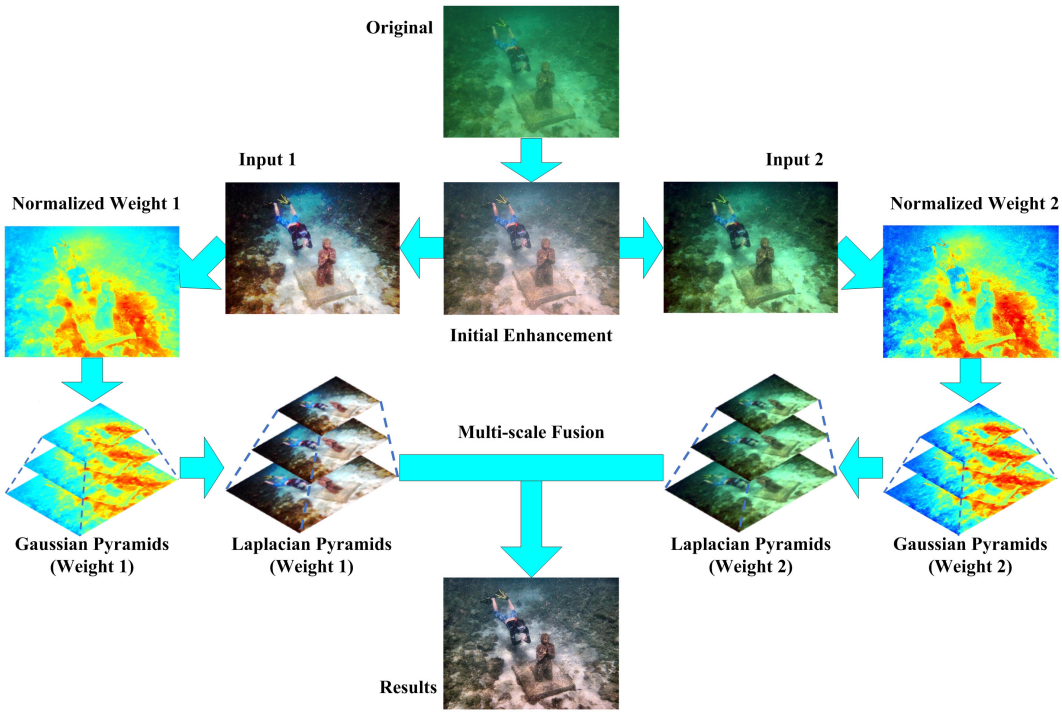


FIGURE 3
The procedure of our method.

et al., 2018; Chen et al., 2019) in the experiment are set in 3 parts for more details.

3.1 Image enhancement visual effect comparison (dataset 1)

Experiment results are shown as Figure 4. Imaging results in Rayleigh (Abdul Ghani and Mat Isa, 2015) show varying degrees of colour restoration, especially overcompensation in the red channel loss of several local details. Images from the algorithms demonstrate poor quality with excessive compensation and partial colour cases (Huang et al., 2018; Chen et al., 2019). The image contrast has improved significantly (Pisano et al., 1998; Huang et al., 2018), but the colour restoration effect is still poor. In terms of image enhancement and colour restoration, UDCP has not outperformed the competition (Drews-Jr et al., 2013). However, the experimental method described in this paper has demonstrated best performance on a variety of

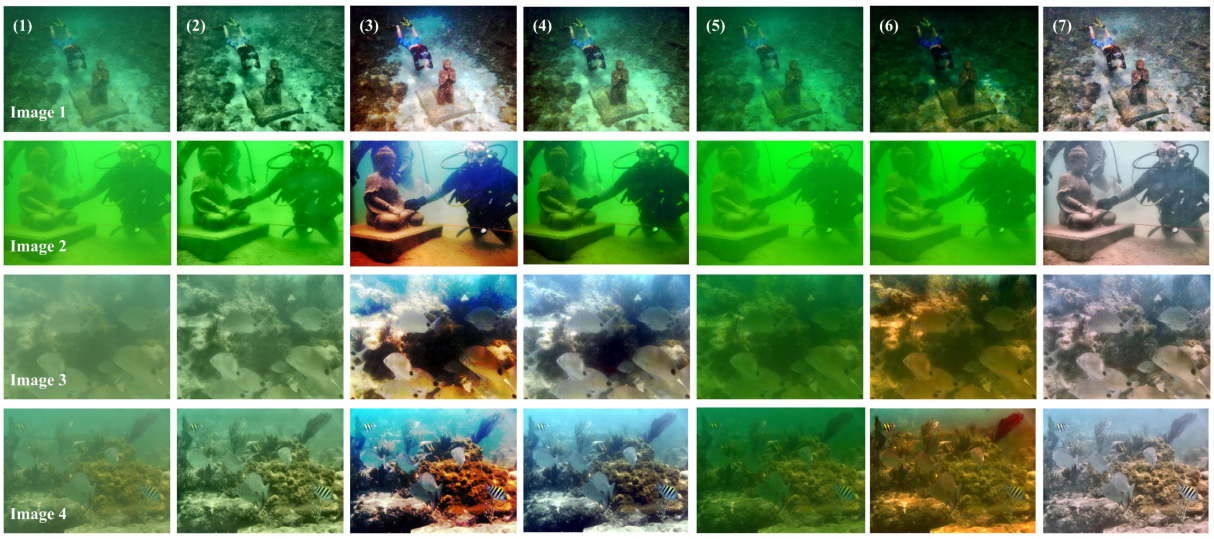


FIGURE 4
The results comparison of underwater images from different methods. (1) The original; (2) CLAHE; (3) Rayleigh; (4) RGHS; (5) UDCP; (6) GAN-RS; (7) Our method.

underwater images, allowing it to more effectively correct colour bias, improve contrast, and preserve image details.

To evaluate the results of various algorithms, this article uses several traditional image quality objective evaluation indicators: peak signal-to-noise ratio (PNSR), average gradient (Average Gradient), tone (Chroma). A higher PNSR value indicates that the algorithm introduces a small amount of noise and retains more valuable image information. The mean gradient value reflects the small detail contrast feature in image, and the information entropy indicates the mean amount of information contained in that. The tone is a summary of the color of the entire image.

The above quality assess indicators are used to evaluate the results of each algorithm after performing comparative experiments on the four contrast images mentioned above. Table 1 shows the mean values of each algorithm's enhancement results for multiple images on various evaluation indicators. According to the results, as seen in Figure 5, the image processed by the algorithm in this paper introduces less noise, retains more effective image information, and has a better processing performance than other algorithms. It also shows that this algorithm presented in this paper shows high practical application value and can meet the requirements in this field.

3.2 Image enhancement visual effect comparison (dataset 2)

As shown in Figure 6, the experimental results indicate that the previous six algorithms achieved different degrees of color restoration, but could not simultaneously achieve good enhancement effects. The Rayleigh algorithm led to excessive compensation in the red channel and the loss of many local details, while the ICM and UCM algorithms did not perform well in color restoration of green-tinted images, resulting in overcompensation and color deviation. Although CLAHE and RGHS algorithms significantly increased the image contrast, their color restoration effects still need to be improved. However, the experimental method in this paper achieved excellent results for multiple underwater images, which could more effectively correct color deviation, significantly improve contrast, and retain image details.

The average values of the algorithmically enhanced results of multiple images over different evaluation metrics are shown in Table 2, such as AG (Average Gradient), UCIQE (Underwater Color Image Quality Evaluation metric) (Yang and Sowmya, 2015), UIQM (Underwater Image Quality Measures) (Panetta et al., 2015), tone (Chroma) and Entropy. The bold data within the tables represent the maximum value of the column data.

TABLE 1 Objective evaluation of underwater image quality.

METHOD	CLAHE	Rayleigh	RGHS	UDCP	GAN-RS	OURS
PSNR	22.1945	15.3600	17.5609	19.3710	17.1422	21.0493
Gradient	2.8410	3.9744	2.7206	1.2144	1.8089	4.1863
Entropy	13.8542	16.4866	15.2486	13.5358	13.7250	16.5132
Chroma	0.7804	0.5631	0.7093	0.6749	0.6144	0.8041

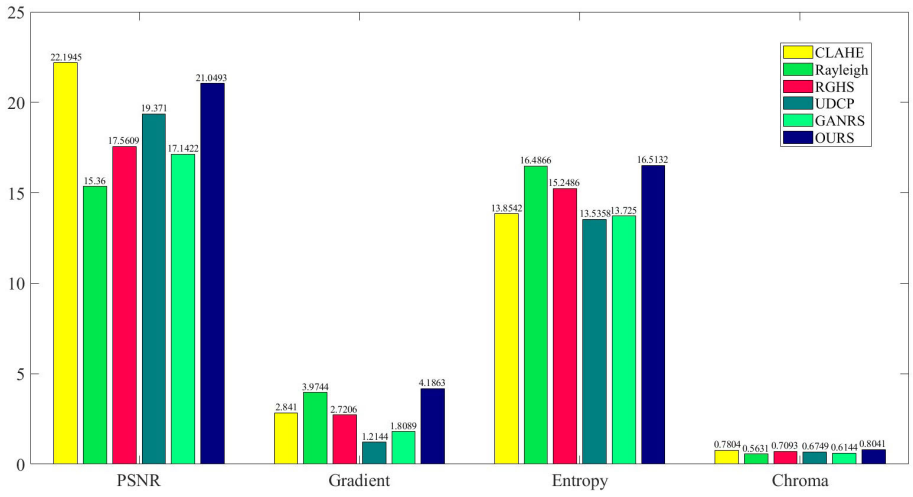


FIGURE 5
Visual evaluation of underwater image quality (Dataset 1).

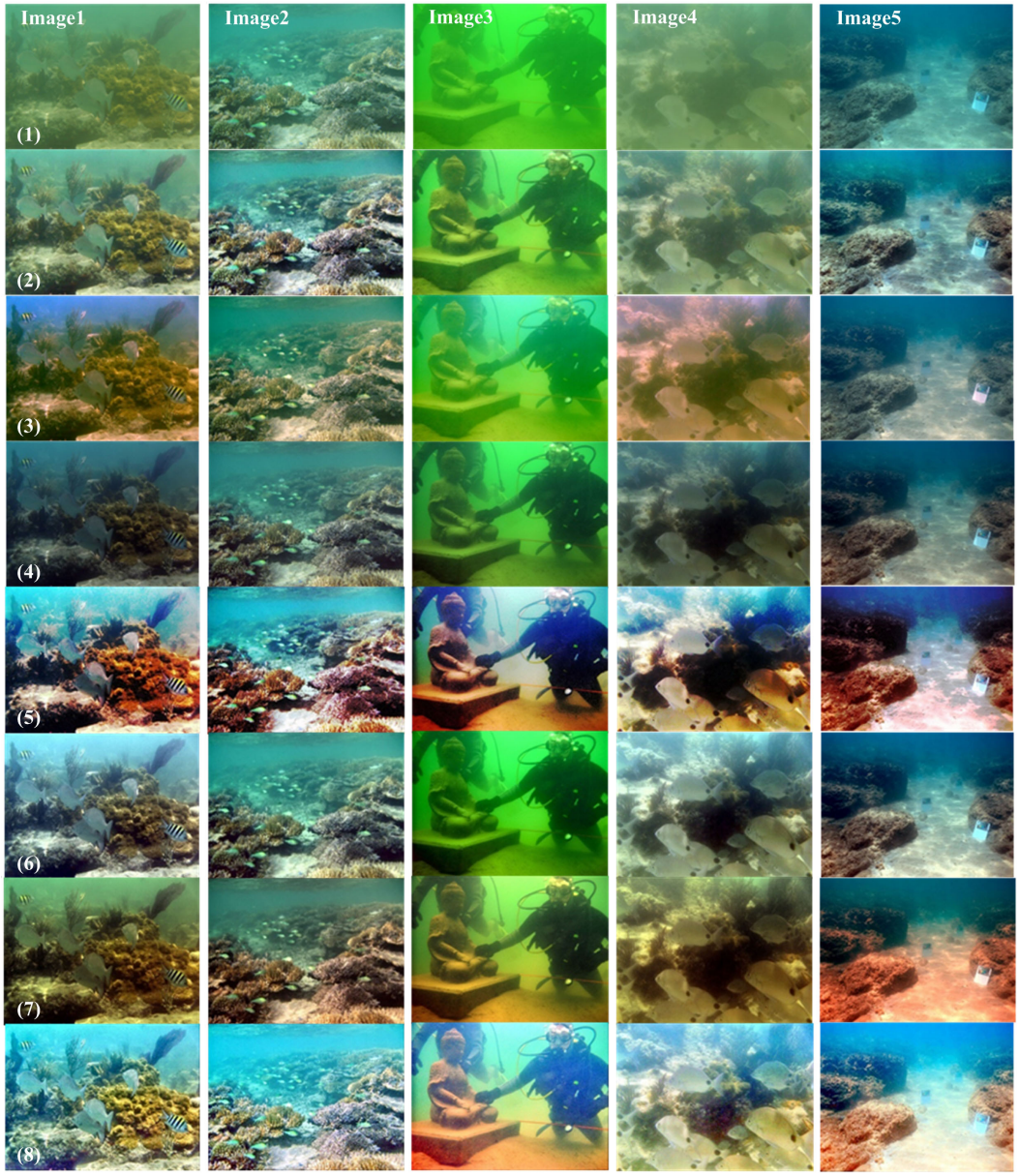


FIGURE 6
The results comparison of underwater images from different sources. (1) The original; (2) CLAHE; (3) IBLA; (4) ICM; (5) Rayleigh; (6) RGHS; (7) UCM; (8) Ours.

TABLE 2(A) The comparison of results with Image 1.

Method	CLAHE	IBLA	ICM	Rayleigh	RGHS	UCM	Ours
AG	6.8776	5.9204	4.4001	11.5741	8.8502	7.153	**13.2663**
UCIQE	0.5372	0.6133	0.5264	0.6374	0.6408	0.6042	**0.6411**
UIQM	1.2473	1.1661	1.1358	1.2888	1.3006	1.2154	**1.4265**
Chroma	**0.8135**	0.7936	0.6928	0.4692	0.7393	0.6584	0.7819
Entropy	14.7762	15.0386	13.7117	15.5033	15.0503	14.7392	**15.6574**

Bold values means the best result.

TABLE 2(B) The comparison of results with Image 2.

Method	CLAHE	IBLA	ICM	Rayleigh	RGHS	UCM	Ours
AG	13.8597	9.6572	8.8445	**15.5622**	9.0636	8.921	15.2386
UCIQE	0.5974	0.5727	0.5712	0.6271	0.6211	0.6313	**0.633**
UIQM	1.3257	1.2008	1.2137	1.3773	1.3034	1.3726	**1.492**
Chroma	**0.8135**	0.6009	0.5362	0.7715	0.7458	0.4322	0.695
Entropy	15.3721	14.9253	14.4406	15.5692	14.9615	14.7066	**15.5797**

Bold values means the best result.

TABLE 2(C) The comparison of results with Image 3.

Method	CLAHE	IBLA	ICM	Rayleigh	RGHS	UCM	Ours
AG	4.0601	2.7305	2.965	6.481	3.1915	4.334	**6.5679**
UCIQE	0.4899	0.5034	0.5675	0.6376	0.578	0.6234	**0.6387**
UIQM	1.1075	0.954	1.1899	1.2996	1.0518	1.1141	**1.3322**
Chroma	0.7029	0.6994	0.709	0.5895	0.5997	0.6156	**0.7098**
Entropy	14.7915	14.6287	14.371	15.3776	14.4284	14.9684	**15.5584**

Bold values means the best result.

TABLE 2(D) The comparison of results with Image 4.

Method	CLAHE	IBLA	ICM	Rayleigh	RGHS	UCM	Ours
AG	3.6143	3.4656	2.9932	6.1934	5.1815	4.8108	**7.2521**
UCIQE	0.439	0.511	0.5233	0.6175	0.6179	0.6019	**0.6276**
UIQM	1.0364	1.3517	1.3699	1.5013	1.3135	1.2299	**1.5411**
Chroma	**0.827**	0.8013	0.529	0.5281	0.7786	0.7067	0.8097
Entropy	13.8838	14.386	13.6957	**16.3354**	14.8868	14.4471	15.9288

Bold values means the best result.

TABLE 2(E) The comparison of results with Image 5.

Method	CLAHE	IBLA	ICM	Rayleigh	RGHS	UCM	Ours
AG	6.0148	4.1758	4.3871	7.2555	5.8684	5.7459	**6.8584**
UCIQE	0.5746	0.5558	0.5922	0.6835	0.6587	0.6239	**0.6874**
UIQM	1.2685	1.3445	1.3998	1.4825	1.2947	1.0311	**1.4871**
Chroma	**0.8586**	0.7858	0.4803	0.7897	0.7149	0.5494	0.7969
Entropy	14.9596	13.4666	13.81	15.0161	14.4415	14.5126	**15.1991**

Bold values means the best result.

By comparing various underwater image processing methods and the proposed algorithm in this paper, as seen in Figure 7, the method processed image obtained the maximum UCIQE value and UIQM value, which demonstrates that the algorithm performs well in enhancing underwater degraded images. Meanwhile, our method also shows significant superiority in terms of image average gradient and entropy. In terms of image chroma, the proposed algorithm performs well in enhancing underwater green-tinted degraded images. The above experimental results indicate that the proposed fusion algorithm has excellent performance in enhancing underwater images.

3.3 Potential applications

Underwater image preprocessing is used to create high-quality underwater images for use in other applications. The feature matching test in this paper is performed using the SIFT (Lowe,

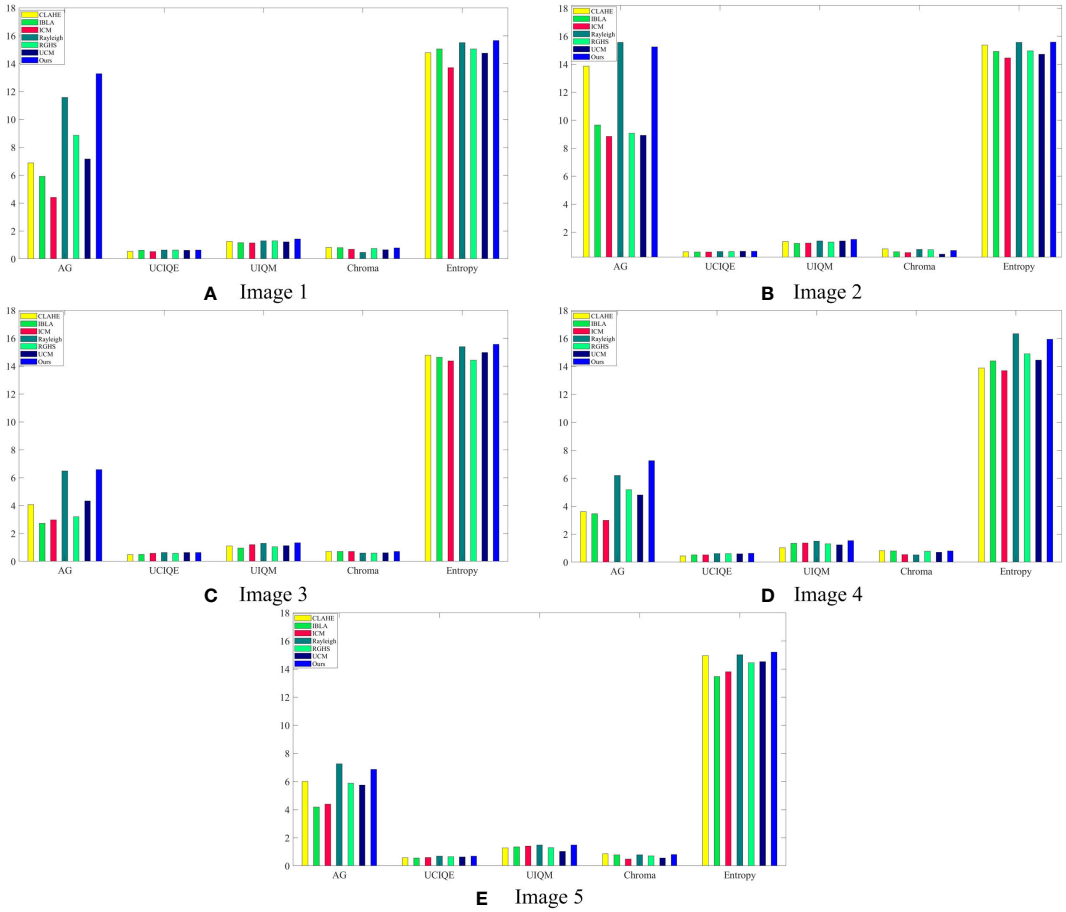


FIGURE 7
The comparison of result in image 1 to 5.

2004). We find the correspondence between two sets of similar underwater images under the same experimental conditions, compare the number of feature points before and after image processing, and verify the practical application of this algorithm's effectiveness.

Figure 8 depicts the result of feature point matching and comparison. According to the comparison results, it can be inferred that the images number with accurately matched feature points has increased as a result within this method. According to the above results, it can be inferred that after image fusion based on this method, the corresponding fusion image quality is significantly improved, which can better meet the subsequent recognition requirements and show high application value.

The image processed by ours has a good application performance in the feature extraction process, according to application test results. At the same time, the experiment used feature point matching processed by various enhancement algorithms. Table 3 exhibits the experimental comparison tests result. The numbers of image matching feature points processed by the method in this study are higher than those of other methods, indicating that it performs better in real-world applications.

4 Conclusions

Images captured in offshore waters often suffer from low contrast, uneven colors, and varying degrees of blur. To address these issues, we propose a new fusion algorithm that employs color compensation, homomorphic filtering, and L-channel histogram equalization technology to enhance the visual quality of underwater images in shallow sea water through multi-scale fusion processing.

Our algorithm significantly improves the visibility of underwater images in a variety of shallow sea scenes, enhancing color restoration and sharpening effects as shown in subjective image visual effect demonstrations.

Experimental comparison tests showed that utilizing our method for image preprocessing significantly enhances the quality of relevant underwater vision tasks. However, it should



FIGURE 8
SIFT feature matching. (1) Original; (2) CLAHE; (3) Rayleigh; (4) RGHS; (5) Ours.

TABLE 3 Comparison of the number of matching features points between original and processed images.

IMAGE SET	ORIGINAL	CLAHE	Rayleigh	RGHS	OURS
GROUP 1	5	66	60	41	79
GROUP 2	2	51	27	25	55

be noted that the proposed method can lead to overcompensation and correction of colors. In future work, we aim to improve the color restoration and further enhance underwater image quality.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://li-chongyi.github.io/proj_benchmark.html.

Author contributions

HZ proposed the main frame of the underwater images enhancement and its process, and revised the manuscript. LG assisted to complete this process and experiments, and drafted a manuscript. XL assisted to complete data analysis and draw. FL assisted to improve this frame and get better results. JY assisted to revise the manuscript to make it more fluent.

Funding

This research has obtain the support of the Key research projects of Qingdao Science and Technology Plan (Grant Number: 22-3-3-hygg-30-hy), the Natural Science Foundation of Shandong Province (Grant Number: ZR2022ZD38), and Basic Research Projects of Qilu University of Technology (Grant Number: 2022PX053).

Acknowledgments

The authors want to thank reviewers for their suggestions in our revision of manuscripts.

Conflict of interest

LG was employed by Civil Aviation Logistics Technology Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abdul Ghani, A. S., and Mat Isa, N. A. (2015). Enhancement of low quality underwater image through integrated global and local contrast correction. *Appl. Soft Computing* 37, 332–344. doi: 10.1016/j.asoc.2015.08.033

Ancuti, C. O., and Ancuti, C. (2013). Single image dehazing by multi-scale fusion. *IEEE Trans. Image Processing* 22, 3271–3282. doi: 10.1109/TIP.2013.2262284

Ancuti, C., Ancuti, C. O., Haber, T., and Bekaert, P. (2012). "Enhancing underwater images and videos by fusion," in *2012 IEEE Conference on Computer Vision and Pattern Recognition. (IEEE: Providence, RI, USA)*, 81–88. doi: 10.1109/CVPR.2012.6247661

Ancuti, C. O., Ancuti, C., Vleeschouwer, C. D., and Sbert, M. (2020). Color channel compensation (3C): a fundamental pre-processing step for image enhancement. *IEEE Trans. Image Processing* 29, 2653–2665. doi: 10.1109/TIP.2019.2951304

Chang, H., Cheng, C. Y., and Sung, C. C. (2018). Single underwater image restoration based on depth estimation and transmission compensation. *IEEE J. Oceanic Eng.*, 44 1–20. doi: 10.1109/JOE.2018.2865045

Chen, T., Wang, N., Wang, R., Zhao, H., and Zhang, G. (2021). One-stage CNN detector-based benthonic organisms detection with limited training dataset. *Neural Networks* 144, 247–259. doi: 10.1016/j.neunet.2021.08.014

Chen, X., Yu, J., Kong, S., Wu, Z., Fang, X., and Wen, L. (2019). Towards real-time advancement of underwater visual quality with GAN. *IEEE Trans. Ind. Electronics* 66, 9350–9359. doi: 10.1109/TIE.2019.2893840

Drews-Jr, P., do Nascimento, E., Moraes, F., Botelho, S., and Campos, M. (2013). "Transmission estimation in underwater single images," in *International Conference on Computer Vision - Workshop on Underwater Vision*. (IEEE:Sydney, NSW, Australia), 825–830.

Guo, J. C., Li, C. Y., Guo, C. L., and Chen, S. J. (2017). Research progress of underwater image enhancement and restoration methods. *J. Image Graphics* 22, 0273–0287. doi: 10.11834/jig.20170301

Huang, D., Wang, Y., Song, W., Sequeira, J., and Mavromatis, S. (2018). "Shallow-water image enhancement using relative global histogram stretching based on adaptive parameter acquisition," in *24th International Conference on Multimedia Modeling - MMM2018*. (MultiMedia Modeling: Bangkok, Thailand).

Hummel, R. (1977). Image enhancement by histogram transformation. *Comput. Graphics Image Processing* 6, 184–195. doi: 10.1016/S0146-664X(77)80011-7

Iqbal, K., Odetayo, M., James, A., Salam, R. A., and Talib, A. Z. H. (2010). "Enhancing the low quality images using unsupervised colour correction method," in *2010 IEEE International Conference on Systems, Man and Cybernetics*. (IEEE:Istanbul, Turkey), 1703–1709.

Iqbal, K., Salam, R. A., Osman, A., and Talib, A. Z. (2007). Underwater image enhancement using an integrated colour model. *IAENG Int. J. Comput. Sci.* 34, 239–244.

Jiang, Q., Gu, Y., Li, C., Cong, R., and Shao, F. (2022). Underwater image enhancement quality evaluation: benchmark database and objective metric. *IEEE Trans. Circuits Syst. Video Technol.* 32 (9), 5959–5974. doi: 10.1109/TCSVT.2022.3164918

Jiao, Z., and Xu, B. (2010). Color image illumination compensation based on HSV transform and homomorphic filtering. *Comput. Eng. Applications* 46, 142–144. doi: 10.3778/j.issn.1002-8331.2010.30.042

Li, K., Wu, L., Qi, Q., Liu, W., Gao, X., Zhou, L., et al. (2022). Beyond single reference for training: underwater image enhancement via comparative learning, in *IEEE Transactions on Circuits and Systems for Video Technology*, 2022. 3225376. doi: 10.1109/TCSVT

Li, J., Xu, W., Deng, L., Xiao, Y., Han, Z., and Zheng, H. (2023). Deep learning for visual recognition and detection of aquatic animals: a review. *Rev. Aquacul.* 15 (2), 409–433. doi: 10.1111/raq.12726

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60, 91–110. doi: 10.1023/B:VISI.0000029664.99615.94

Panetta, K., Gao, C., and Agaian, S. (2015). Human-visual-system-inspired underwater image quality measures. *IEEE J. Oceanic Eng.* 41 (3), 541–551. doi: 10.1109/JOE.2015.2469915

Peng, Y. T., and Cosman, P. C. (2017). Underwater image restoration based on image blurriness and light absorption. *IEEE Trans. image processing* 26 (4), 1579–1594. doi: 10.1109/TIP.2017.2663846

Pisano, E. D., Zong, S., Hemminger, B. M., DeLuca, M., Johnston, R. E., Muller, K., et al. (1998). Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms. *J. Digital Imaging* 11, 193–200. doi: 10.1007/BF03178082

Qi, Q., Li, K., Zheng, H., Gao, X., Hou, G., and Sun, K. (2022). SGUIE-net: semantic attention guided underwater image enhancement with multi-scale perception. *IEEE Trans. Image Processing* 31, 6816–6830. doi: 10.1109/TIP.2022.3216208

Sahu, P., Gupta, N., and Sharma, N. (2014). A survey on underwater image enhancement techniques. *Int. J. Comput. Applications* 87, 19–23. doi: 10.5120/15268-3743

Sun, S., Wang, H., Zhang, H., Li, M., Xiang, M., Luo, C., et al. (2022). Underwater image enhancement with reinforcement learning. *IEEE J. Oceanic Eng.*, 1–13. doi: 10.1109/JOE.2022.3152519

Wang, N., Chen, T., Liu, S., Wang, R., Karimi, H. R., and Lin, Y. (2023). Deep learning-based visual detection of marine organisms: a survey. *Neurocomputing* 532, 1–32. doi: 10.1016/j.neucom.2023.02.018

Wang, Y., Song, W., Fortino, G., Qi, L.-Z., Zhang, W., and Liotta, A. (2019). An experimental-based review of image enhancement and image restoration methods for underwater imaging. *IEEE Access* 7, 140233–140251. doi: 10.1109/ACCESS.2019.2932130

Wang, N., Wang, Y., and Er, M. J. (2022). Review on deep learning techniques for marine object recognition: architectures and algorithms. *Control Eng. Practice* 118, 104458. doi: 10.1016/j.conengprac.2020.104458

Yang, M., Hu, J., Li, C., Rohde, G., Du, Y., and Hu, K. (2019). An in-depth survey of underwater image enhancement and restoration. *IEEE Access* 7, 123638–123657. doi: 10.1109/ACCESS.2019.2932611

Yang, M., and Sowmya, A. (2015). An underwater color image quality evaluation metric. *IEEE Trans. Image Process* 24, 6062–6071. doi: 10.1109/TIP.2015.2491020

Yu, X., Qu, Y., and Hong, M. (2019). Underwater-GAN: underwater image restoration via conditional generative adversarial network. *Lecture Notes Comput. Sci.* 11188, 66–75. doi: 10.1007/978-3-030-05792-3_7

Zhou, J., Sun, J., Zhang, W., and Lin, Z. (2023a). Multi-view underwater image enhancement method via embedded fusion mechanism. *Eng. Appl. Artif. Intelligence* 121, 105946. doi: 10.1016/j.engappai.2023.105946

Zhou, J., Yang, T., Chu, W., and Zhang, W. (2022). Underwater image restoration via backscatter pixel prior and color compensation. *Engineer. App. Art. Intelligence* 111, 104785. doi: 10.1016/j.engappai.2022.104785

Zhou, J., Zhang, D., and Zhang, W. (2023b). Cross-view enhancement network for underwater images. *Eng. Appl. Artif. Intelligence* 121, 105952. doi: 10.1016/j.engappai.2023.105952

frontiers | Frontiers in Marine Science

TYPE Methods
PUBLISHED 26 May 2023
DOI 10.3389/fmars.2023.1094190

Check for updates

OPEN ACCESS

EDITED BY
Mark C. Benfield,
Louisiana State University, United States

REVIEWED BY
Shinnosuke Nakayama,
Stanford University, United States
Peng Ren,
China University of Petroleum (East China), China

*CORRESPONDENCE
Zhiyong Zhang
zhang.zhiyo@northeastern.edu

RECEIVED 09 November 2022
ACCEPTED 03 May 2023
PUBLISHED 26 May 2023

CITATION
Zhang Z, Kaveti P, Singh H, Powell A, Fruh E and Clarke ME (2023) An iterative labeling method for annotating marine life imagery. *Front. Mar. Sci.* 10:1094190. doi: 10.3389/fmars.2023.1094190

COPYRIGHT
© 2023 Zhang, Kaveti, Singh, Powell, Fruh and Clarke. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

An iterative labeling method for annotating marine life imagery

Zhiyong Zhang[1]*, Pushyami Kaveti[1], Hanumant Singh[1], Abigail Powell[2], Erica Fruh[2] and M. Elizabeth Clarke[2]

[1]College of Engineering, Northeastern University, Boston, MA, United States, [2]Northwest Fisheries Science Center, National Oceanic and Atmospheric Administration (NOAA), Seattle, WA, United States

This paper presents a labeling methodology for marine life data using a weakly supervised learning framework. The methodology iteratively trains a deep learning model using non-expert labels obtained from crowdsourcing. This approach enables us to converge on a labeled image dataset through multiple training and production loops that leverage crowdsourcing interfaces. We present our algorithm and its results on two separate sets of image data collected using the Seabed autonomous underwater vehicle. The first dataset consists of 10,505 images that were point annotated by NOAA biologists. This dataset allows us to validate the accuracy of our labeling process. We also apply our algorithm and methodology to a second dataset consisting of 3,968 completely unlabeled images. These image categories are challenging to label, such as sponges. Qualitatively, our results indicate that training with a tiny subset and iterating on those results allows us to converge to a large, highly annotated dataset with a small number of iterations. To demonstrate the effectiveness of our methodology quantitatively, we tabulate the mean average precision (mAP) of the model as the number of iterations increases.

KEYWORDS

iterative labeling, active learning, Faster R-CNN, NOAA, Amazon MTurk, auto-approval, background label

1 Introduction

Technologies for imaging the deep seafloor have evolved significantly over the last three decades (Durden et al., 2016). These technologies have enabled the study and monitoring of the spatiotemporal changes of marine life in the vast ocean space. They should ultimately enable us to conduct more efficient fishery independent surveys, yielding improved stock assessments and ecosystem-based management (Francis et al., 2007). Manned submersibles, Remotely Operated Vehicles (ROVs), Autonomous Underwater Vehicles (AUVs) (Singh et al., 2004b), towed vehicles (Taylor et al., 2008), and bottom-mounted and midwater cameras (Amin et al., 2017) have all contributed to an explosion of data in terms of our ability to obtain high-resolution, true-color (Kaeli et al., 2011) camera imagery underwater.

The reality, however, is that extracting actionable information from our large underwater image datasets remains a challenging task. The ability to process the data is not proportional to the rate at which the data is acquired, as traditional methods were resource-intensive in terms of manpower, time, and cost. Efforts are underway to analyze the imagery with various levels of automation using tools from machine learning for a variety of fisheries and habitat monitoring applications, including coral reefs (Singh et al., 2004a; Gleason et al., 2007; Purser et al., 2009; Chen et al., 2021), starfish (Clement et al., 2005; Smith and Dunbabin, 2007), scallops (Dawkins et al., 2017), and commercially important groundfish (Tolimieri et al., 2008).

In parallel, there have been significant developments in deep learning (Krizhevsky et al., 2012; Simonyan and Zisserman, 2014), which further propelled these efforts by truly leveraging the availability of large amounts of data. Multiple works have explored the use of standard deep convolutional neural networks for image segmentation and classification (Ramani and Patrick, 1992; Anantharajah et al., 2014; Boom et al., 2014; Cutter et al., 2015; Fisher et al., 2016; Marburg and Bigham, 2016; Sung et al., 2017; Kaveti and Singh, 2018; Wang et al., 2021). Reinforcement learning has been used to enhance underwater imagery to improve the performance of object detection networks, (Wang et al., 2023; yu Wang et al., 2023). These works have helped marine biologists analyze underwater imagery far more efficiently.

1.1 Generation of labeled underwater datasets

The remarkable success of deep learning techniques is primarily due to the availability of large labeled datasets. A number of public underwater image databases, such as FathomNet (Katija et al., 2021), EcoTaxa (Blue-Cloud, 2019), DeepFish (Saleh et al., 2020), WildFish++ (Zhuang et al., 2021), and BIIGLE 2.0 (Langenkämper et al., 2017), have come into existence in recent years. These works provide a platform and tools for annotating, uploading, and downloading annotated images, and sometimes also training or testing machine learning models. Generating labeled datasets by manually going through vast amounts of video and image streams is a time-consuming task. Several efforts have been initiated toward machine learning-assisted automation for annotating underwater datasets. CoralNet 1.0 (Chen et al., 2021) is a data repository that also deploys a feature extractor network pre-trained on a large collection of data to generate annotations of coral reefs automatically. (Zurowietz et al., 2018) propose a multi-stage method where an auto encoder network generates training proposals that are filtered by human observers and used to train a segmentation network, the results of which are further reviewed manually.

However, these annotation approaches require human experts with marine biology knowledge, which makes it difficult to generalize and scale to huge volumes of data. In fact, there are a large number of underwater image datasets available with no efficient means to label them. One such example is shown in Figure 1. The absence of well-labeled data is still a primary factor limiting the widespread use of machine learning techniques for marine science research.

One simple solution is to utilize crowdsourcing platforms involving non-expert human users, such as Mechanical Turk (Crowston, 2012) and Zooniverse (Simpson et al., 2014) Crowdsourcing platforms are fairly inexpensive and highly efficient for the rapid generation of annotated datasets. But the results for specialized imagery, such as that associated with marine biology, are often mixed and unreliable. Our own experience has shown that some workers annotate images with randomly placed labels, which requires a prohibitive amount of time and effort spent approving or rejecting these results.

1.2 Performance enhancement on crowdsourcing platforms

Many human-machine collaboration methods have been proposed to improve the efficiency of human in-the-loop annotation. Branson et al. (2010) presents an interactive, hybrid human-computer method for image classification. Deng et al. (2014) focuses on multi-label annotation, which finds the correlation between objects in the real world to reduce the human computation time required for checking their existence in the image. Russakovsky et al. (2015) asks human annotators to answer a series of questions to check and update the predicted bounding boxes, while Wah et al. (2011) queries the user with binary questions to locate the part of the object. Vijayanarasimhan and Grauman (2008) incrementally updates the classifier by requesting multi-level annotations, ranging from full segmentation to a present/absent flag on the image. Kaufmann et al. (2011) and Litman et al. (2015) adapt different models from motivation theory and have studied the effect of extrinsic and intrinsic motivation on worker performance.

Some recent research has shown that when non-experts are trained and clearly instructed on the annotation protocol, they can produce accurate results (Cox et al., 2012; Matabos et al., 2017; Langenkämper et al., 2019), thus demonstrating the potential for combining citizen science with machine learning. Kaveti and Akbar (2020) designed an enhanced MTurk interface and added a guided practice test to achieve higher annotation accuracy. Bhattacharjee and Agrawal (2021) simplified complex tasks on MTurk by combining batches, dummy variables, and worker qualifications. Our work is most similar to LSUN (Yu et al., 2015), in that they hid ground truth labels in the task to verify worker performance and allowed multiple workers to label the same image for quality control.

Thus, we propose a human-in-the-loop annotation methodology that can label very large datasets automatically by combining machine learning with Mechanical Turk crowdsourcing. We utilize a unique iterative process with auto-approval that allows us to check the quality of the workers algorithmically, precisely, efficiently, and without any human intervention. We can also use the same techniques for converting historical expert annotations, as shown in Figure 2A, to quickly create labeled data sets for machine learning that are critically required for fisheries and ecosystem-based management applications.



FIGURE 1
Underwater image samples from one of the datasets with no annotations. There are very large marine life related image datasets that are freely available but are not annotated. These would require significant efforts from experts in the field to label.

In contrast to LSUN (Yu et al., 2015), we only label once per object during the iterative labeling process if the category is not controversial. We define our task as working with individual objects in an image, as opposed to considering all the objects in an entire image. Additionally, we remove qualification tests and add tutorials to lower the barriers for workers to enter our tasks. In this way, we can provide the simplest form of the task to Mechanical Turk workers.

2 The iterative labeling process

The overview of our method for the iterated labeling process for underwater images is illustrated in Figure 3. The process begins by building an initial deep learning model for making bounding box predictions on a small subset of underwater images. These predictions are then published to a crowdsourcing platform with a well-designed assistive interface for validation. An auto-approval

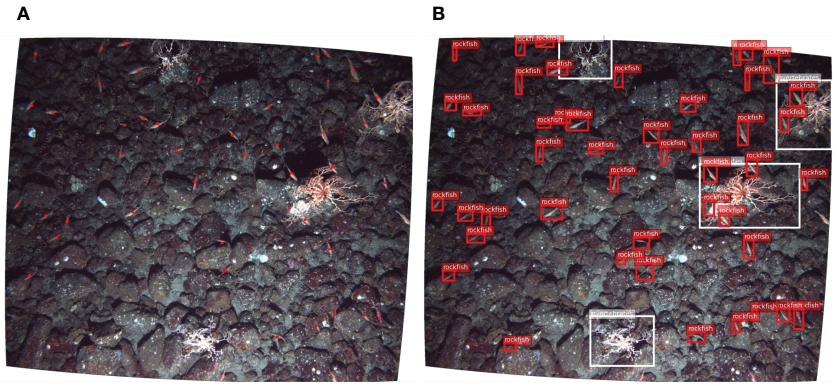


FIGURE 2
NOAA annotation ground truth **(A)** Underwater images annotated by NOAA marine biologists, with dot annotations on each object. **(B)** Extended dot annotations to bounding box labels with MTurk workers.

method filters out bad labels from the crowdsourcing platform. The filtered labels are added to the dataset and used for further training to generate new predictions. Therefore, we start with a small set of annotations and increase the number of annotations with each loop until all objects in all images have been labeled. Figure 4 shows an example of the predict-update loop for a single image.

2.1 The initial model

We start with a small seed dataset labeled by marine biologists. This serves as our initial dataset, which we use to train our deep learning object detection model. The seed dataset should consist of different forms of the object that we are about to label. In our case, this data is not large enough to completely train a high accuracy model, but it is sufficient to make reasonable predictions to feed into the first iteration of our process.

As the iterative labeling process does not have real-time constraints, we chose Faster R-CNN (Ren et al., 2015) as the object detection network in combination with ResNet-50 (He et al., 2015) as the backbone network. Feature Pyramid Networks (Lin et al., 2016) were applied for multi-scale object detection. We built the network based on Detectron2 (Wu et al., 2019). We trained the object detection network on 2 RTX 2080 GPUs with a batch size of 2 for 60 epochs. Since the batch size is very small, group normalization (Wu and He, 2018) was used instead of batch normalization. Typically, we use less than 100 images for the initial dataset, and the initial data only takes a few hours to label.

After training the initial model, we utilize it to predict the learned object categories on new unlabeled image data. However, as there is no ground truth available for this data, we cannot be certain if these predictions are true positives. To address this issue, we enlist workers from Mechanical Turk to classify and correct the predictions.

2.2 Assistive annotation interface design

In this section, we describe the design and development of the user interface on MTurk used to facilitate the human-in-the-loop learning process. One of the key aspects of the interface is presenting the user with a convenient way to determine the accuracy of the deep learning model's predictions, and to annotate them if they are correct. These correct object detections are then used as ground truth labels to continue training the deep learning model. The fundamental idea is that through a series of predict (using our algorithm), correct and update (with Mechanical Turk workers), and train (using our algorithm) loops, we will end up with a superior model.

The most common interface design for labeling object instances in images on MTurk requires workers to detect all objects in the image and draw bounding boxes for each object before moving on to the next image. This process can be cumbersome when there are a lot (> 30) of instances per image to label and is especially challenging when the dataset consists of unique, specialized categories of objects. This can also affect the worker's motivation to perform the task (Kaveti and Akbar, 2020). We have made a few novel design choices to construct our MTurk annotation interface, as described below. Figure 5 shows a snapshot of our assistive annotation interface.

2.2.1 Tutorial/examples of annotations

One of the challenges of underwater datasets is that they contain unique and uncommon objects. Moreover, the workers on MTurk come from diverse backgrounds with variability in experience and expertise. To address this issue, we have dedicated a small portion of the interface to showcase a set of labeling examples for the various marine species encountered in the dataset. This helps to familiarize workers with the dataset and improve the quality of their labeling.

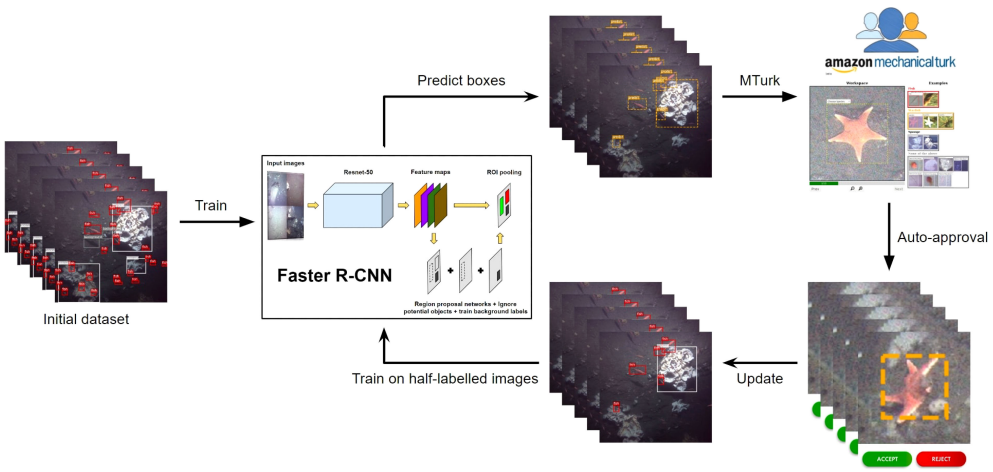


FIGURE 3
The complete iterative labeling process: We begin by training the initial dataset with Faster R-CNN (Ren et al., 2015) in order to obtain an initial model. Next, we use this initial model to predict objects in a new image set, and publish the prediction boxes to MTurk for correction. Auto-approval filters the labeling results. Then, we update the labels and train on half-labeled images to obtain a new model and prediction set for the next loop. Finally, we converge to a completely labeled dataset in 3-6 iterations.

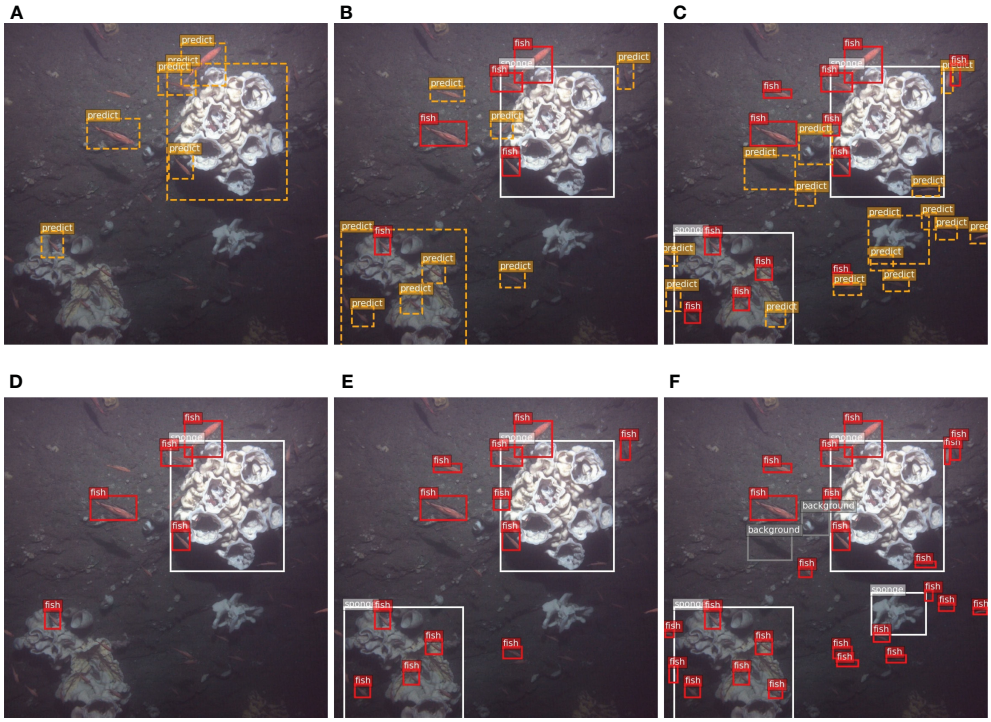


FIGURE 4
An example of the iterative labeling process. The orange dashed boxes represent the predictions of each loop. These prediction boxes are published to MTurk for correction. The updated labels, based on the MTurk results, are then used for the next iteration **(A)** Loop 1 predict, **(B)** Loop 2 predict, **(C)** Loop 3 predict, **(D)** Loop 1 update, **(E)** Loop 2 update, **(F)** Loop 3 update.

2.2.2 Labeling cues

Instead of asking workers to find all possible instances of categories in a raw image, we provide several labeling cues to make it easy for them. We show the predictions made by the deep learning model as a dashed bounding box. The workers are then asked to adjust it to tightly fit the object and choose the species from a dropdown menu. These features help correct localization and classification losses during supervision. Sometimes, the background in images can be mistakenly predicted as a species. To address this issue, we added a "None of the above" option to the species dropdown menu, which corresponds to the background.

2.2.3 UI controls

The images in our underwater dataset can contain 40-50 instances of relevant objects per image. Sometimes, these instances can be really small and occluded by other objects due to overlap, as shown in Figure 4. Therefore, we choose to zoom in and display each bounding box prediction individually, rather than showing all of the boxes at the same time. This allows workers to focus on a single object at a time, which is beneficial for labeling tiny objects and also improves user performance when adjusting the bounding boxes.

2.3 The auto-approval process

The biggest drawback of the MTurk platform is with respect to the quality control of workers. Although MTurk allows one to select workers based on certain criteria or through a test, requesters often end up spending a lot of time and resources reviewing annotation results. This negates the purpose of wanting to create a fully automated human-in-the-loop annotation process. Therefore, we have developed an auto-approval mechanism to assess how well workers are performing and to accept or reject annotations without any intervention.

The auto-approval is accomplished by randomly hiding ground truth tests in the labeling tasks. Each MTurk task consists of nine labeling tasks and one ground truth test task. The ground truth labels are obtained from a manual labeling, which comes from the initial and validation datasets. We compare the worker's labeled bounding box to the ground truth bounding box, and compute the intersection over union (IOU) of the two bounding boxes. We accept the worker's annotations only if the IoU score is higher than the threshold of 0.75. LSUN (Yu et al., 2015) proposes a similar method, using hidden ground truth data to validate the MTurk labeling results. However, they use the entire image as a labeling task, while we use every single object.

2.3.1 Double checking identifications

The incorrect classification of objects can lead to incorrect training. Therefore, even if a sub-task has passed the hidden ground truth test, we still need to double-check the class that is chosen. If the selected class is different from the predicted class, we add the sub-task to the republish list. Meanwhile, we change the class of the predicted box to the one selected by the current worker.

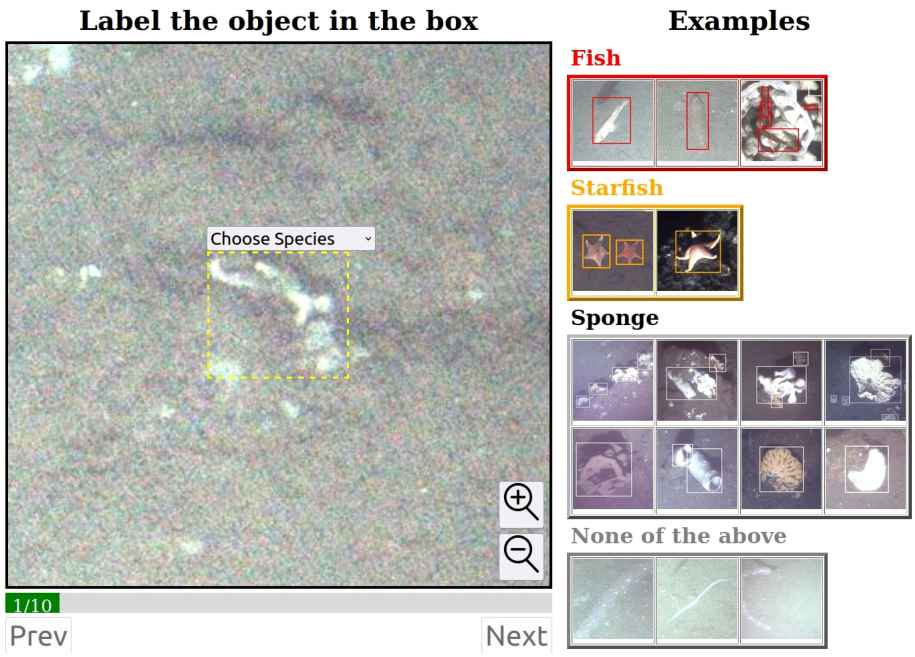


FIGURE 5
The Assistive Interface is designed to help MTurk workers by providing them with a tool to focus on a specific object. By simply identifying the species and fitting the bounding box, the labels' reliability is significantly improved. The ground truth is hidden at the last task to implement auto-approval.

This means that the class of the object is determined only if two consecutive workers choose the same category. Otherwise, the prediction box would be repeatedly republished under this mechanism. If an object is actually a background, it would be republished at least twice to fully determine that it is the background.

To get a sense of the efficiency and cost of the process, we examined one representative batch of tasks that was given to MTurk workers. In this batch, there were 4,583 tasks. Each task required 9 labels and 1 ground truth test, and cost three cents, which works out to a cent for three labels. On average, each task took 3 minutes and 5 seconds to complete, and our tasks are easy to complete. For the entire batch, it took about 6 hours to finish all the tasks. Out of the 4,583 tasks in this particular batch, 3,413 tasks were auto-approved as passed, while 1,170 were rejected.

2.4 Training on half-labeled images

In the first iteration, where the prediction is based on the initial model, not all object instances in the images will be discovered, and the accuracy of the predictions cannot be guaranteed. This is because the initial model is trained only on a small seed dataset, which is insufficient to fully train the model. These predictions are sent to MTurk for correction. The new bounding boxes are then used to supervise the training of our deep learning model, which in turn makes new and more accurate predictions. However, since the object labels of the images are incomplete, some issues arise in the training process. Therefore, we make modifications to the training phase, including feeding appropriate training data and loss functions to suit our iterative labeling process. The detailed changes to the loss function can be found in 2.4.4.

2.4.1 Modifications to Faster R-CNN to avoid negative mining of potential objects

During the training of an object detection model, if an object is not labeled in the images, it will be implicitly treated as a background class. This is especially true for algorithms such as SSD (Liu et al., 2015) and Faster R-CNN (Ren et al., 2015), which use negative hard sampling to train the background class. In SSD, the top N highest confidence predictions that do not match any ground truth are selected and trained as negative samples. Meanwhile, Faster R-CNN randomly selects a certain percentage of prediction boxes without matching ground truths as negative samples. However, this can cause serious issues with our training because if half of the objects in the image are not labeled, it will prevent the trained model from converging.

The solution to this issue is to identify unlabeled potential objects and avoid training them as negative samples. When the prediction confidence score of an anchor exceeds a specific threshold and there are no ground truth objects that match that prediction, it implies that the model thinks there may be a potential object at that spot. Therefore, this object should be ignored in the training process to be discovered later, as shown in Figure 6. In the Region Proposal Network (RPN) of Faster R-CNN, we mark all the prior anchors whose confidence score exceeds 0.5 without a ground truth label as "ignored". We exclude them from being selected as negative samples for loss calculations and also prevent them from

being selected to enter the next stage of the process, which is the region of interest (ROI) layer.

2.4.2 Training background labels

In the previous section, we described how to avoid training potential true positive predictions as a background class. In this section, we discuss how to correctly train the false positive (background) class. During the iterative labeling process, some predictions are false positives and are corrected as "background" by the MTurk workers. These background labels can be used in the training process.

In the Region Proposal Network (RPN), instead of randomly selecting negative samples, the boxes that are updated as the "background" class from the MTurk auto-approval process should be trained. When the number of negative samples is significant, the probability of being trained as a potential object is very low. This is valuable because it increases the precision of our object detection model and avoids ignoring potential objects. We do not calculate the localization loss of background labels as they are negative samples, and their use ends with the RPN. Training background labels properly can reduce false positives, in other words, increasing the precision of the model.

2.4.3 Data augmentation

We also perform data augmentation to generate more training samples. All the images are put through the following transformations: a flip of the image horizontally and vertically, adjustments to brightness by scaling the intensity randomly between 0.8 and1.2, and a random scaling factor corresponding to 0.8 to 1.0 of the image size.

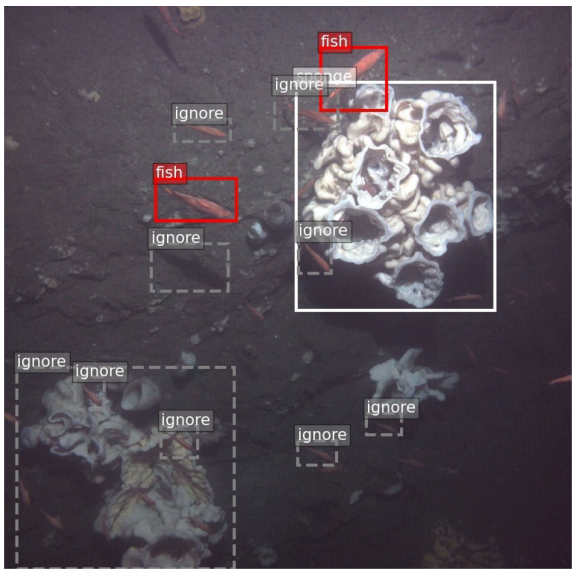


FIGURE 6
Ignoring potential objects: The dashed bounding boxes represent potential objects with a significant classification score that have not yet been verified and labeled. These unlabeled objects may be wrongly classified as negative samples during training iterations, leading to an incorrect model. To avoid this, we detect potential objects using a classification score threshold and exclude them from the training process.

2.4.4 Loss function

Taking into account the above mentioned changes to the training phase, the loss function can be divided into four components:

- The classification loss, $\sum_i L_{cls}(p_i, p_i^*)$, where the predicted labels have object class ground truths associated with them. ground truth bounding boxes are obtained from MTurk after auto-approval. N: RPN mini-batch size
- The classification loss $\sum_j L_{cls}(p_j, p_j^*)$, where a background class ground truth box is associated with the predicted label. This ground truth is also obtained from MTurk after the auto-approved label is selected as back-ground.
- The classification loss $\sum_k L_{cls}(p_k, p_k^*)]$, where the predicted box does not have any ground truth box associated with it but the prediction score with respect to an object class is less than 0.5. In this case we consider this as a negative sample.
- The regression loss $\lambda \frac{1}{N} \sum_i L_{reg}(t_i, t_i^*)$ which is computed for the predicted labels which have object class ground truth boxes associated with them.

Putting all the components together the loss function can be written as

$$L(\{p_i\},\{p_j\},\{p_k\},\{t_i\}) = \frac{1}{N}[\sum_i L_{cls}(p_i, p_i^*) + \sum_j L_{cls}(p_j, p_j^*) + \sum_k L_{cls}(p_k, p_k^*)] + \lambda \frac{1}{N} \sum_i L_{reg}(t_i, t_i^*)$$

The classification loss is:

$$L_{cls} = -[p^* \cdot log(p) + (1 - p^*) \cdot log(1 - p)]$$

The localization loss is:

$$L_{reg} = \begin{cases} 0.5 \mid t - t^* \mid^2, & if \ |t - t^*| < 1 \\ |t - t^*| - 0.5, & otherwise \end{cases}$$

where i is the index of an anchor in a mini-batch, whose ground truth is an object. j is the index of an anchor, whose ground truth is a labeled background. k is the index of an anchor, which has no ground truth and p_k is lower than the ignore threshold. $p_{i,j,k}$ is the predicted probability of being an object. $p^*_{i,j,k}$ is the ground truth probability where 1 indicates that it is foreground. 0 means background. Here $p_i^* = 1, p_{j,k}^* = 0$. t_i is a vector representing the 4 parameterized coordinates of the prediction bounding box. t_i^* is the ground truth box associated with a positive anchor. λ is the balancing parameter of object and localization loss.

3 Results and discussions

3.1 Labeling a ground truth dataset

We have a large dataset with dot annotations provided by NOAA marine biologists (Figure 2A). These annotations were made before the advent of machine learning techniques and are unsuitable for machine learning applications due to the absence of bounding boxes around the objects. However, this dataset is ideal for setting up, testing, and validating our efforts. We could then transfer to other datasets with completely unlabeled data, as we discuss in the next section.

We publish these dot labels to MTurk workers using our assistive interface (see Figure 5). The workers can extend the dot annotations to create tight and accurate bounding boxes with the help of the instructions. An example of the extended bounding boxes is shown in Figure 2B. We consider them as ground truth labels to validate the iterative labeling process.

We divided our dataset into two parts, using 51 images as the initial dataset and 632 images as our validation dataset (Table 1). We then applied our iterative labeling process to the remaining 9822 images.

Table 1 shows the iterative labeling results. We ran six iterations to annotate the dataset. The initial dataset is very small (534 labels, 51 images), and the trained model is relatively poor (0.6 mAP). In the first loop, most of the rockfish were labeled, as these are easy for the deep learning model to identify. As the images were half-labeled, we chose to ignore the threshold of 0.5 to prevent training the model on rockfish with prediction scores over 0.5. As the loops iterated, the mAP and recall rate increased, enabling the trained model to detect more rockfish. In the final loop, the mAP and recall rate stopped growing, indicating that the model was unable to detect any more rockfish. We used this as a stopping mechanism for our iterations.

There are a reasonable number of rockfish that are very hard to detect. Typically, these are small and have low contrast (see Figure 7A). To help our algorithm cope with these issues, we crop the large-size image (2448 x 2050) into nine sub-images, each measuring 896 x 896. During prediction, we crop the image in the same way to maintain scale consistency. We also adjust the contrast of the images to perform data augmentation. In the end, about 82% of the rockfish are labeled correctly with very few false positives.

Along with rockfish labels, we also generate background labels to identify false positives. These false positives typically include starfish or invertebrates that resemble fish (see Figure 7B).

We should also point out that the NOAA dataset has been annotated to a greater level of taxonomic resolution, including coral, flatfish, groundfish, etc. The classification of the data to such levels uses very detailed markings and is an interesting and open problem beyond the scope of this work.

3.2 Labeling a dataset with sponges

Our second illustrative dataset, the Pacstorm dataset, contains three categories of marine organisms: fish, starfish, and sponge. In this case, we used 98 images as the initial dataset and 302 as our validation data. We ran the iterative labeling process on the remaining 3,568 images to generate labels (see Table 2).

In this dataset, fish and starfish are easy to identify and label, but sponges are far more challenging. The reasons for this are manifold. The sponges have many different forms (as shown in Figure 8A). Some sponges have a hole on top, while others do not. Some sponges look like white rocks, and others look like white dots. Sponges also have different colors; while most of them are white, some are brown, and dead sponges are black. The trickiest problem is that the sponges can group together (as shown in Figure 8B), making it hard to decide whether to annotate all of them with one label or annotate them separately. Some sponges are covered in mud, with only a small part of them exposed (as shown in

TABLE 1 NOAA dataset with ground truth validation.

Initial dataset					
	Rockfish	Images	mAP/50	recall	precision
	534	51	0.601	0.648	0.758
Iterative labeling process					
Loop	Rockfish	Coverage	mAP/50	recall	precision
1	54906	0.638	0.680	0.703	0.906
2	60508	0.703	0.724	0.752	0.868
3	65084	0.756	0.778	0.803	0.864
4	67299	0.782	0.792	0.817	0.877
5	68829	0.800	0.824	0.854	0.824
6	70609	0.821	0.828	0.858	0.808

In total, we have 91,228 rockfish dot annotations spread over 10,505 images. These annotations were created by NOAA marine biologists. We used 534 labels (51 images) as the initial dataset, and 4654 labels (632 images) as the validation dataset. We used the remaining 86,041 labels (9,822 images) to validate the iterative labeling process. NOAA originally provided dot annotations instead of box annotations. We used the same assistive interface to generate ground truth bounding boxes. The mean average precision calculated at an IOU threshold of 0.50 (mAP/50) is a common metric used to evaluate the performance of an object detection model, and we evaluated our work in a similar manner.

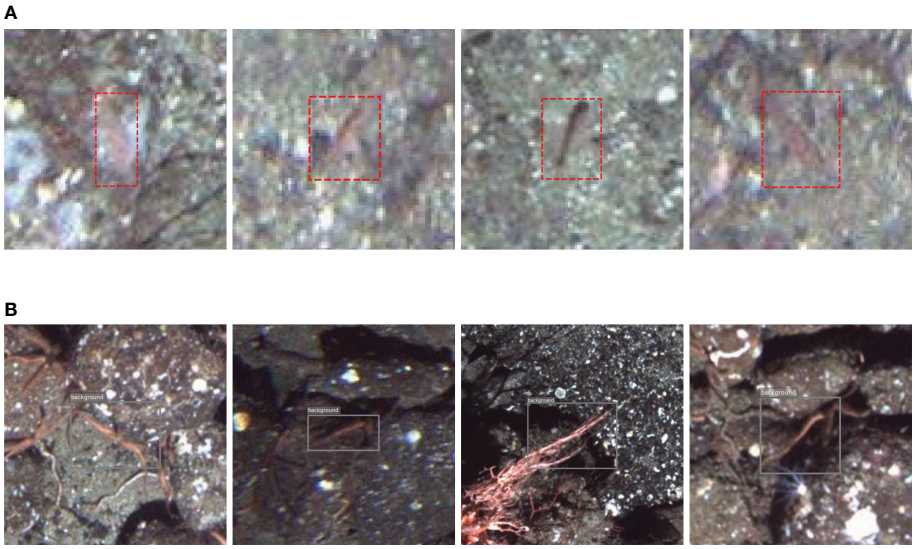


FIGURE 7
(A) Examples of rockfish that our algorithm missed (false negatives). Usually, these specimens are very small and have low contrast. **(B)** Examples of background labels (false positives) for rockfish. Usually, the false positives are either parts of starfish or other invertebrates. Training background labels properly can increase the precision of the model.

Figure 8C). The variety of cases not only confuses the deep learning model but also the MTurk workers. When annotating the initial and validation dataset, these problems make it difficult to maintain consistency in labeling patterns for sponges.

To overcome the problem of different shapes and colors, we presented a large number of sponge examples alongside the assistive annotation interface for worker training. By grouping the sponges together, we can avoid predicting small sponges within a large labeled sponge group.

In the final count, we labeled 12,660 sponges, 3,588 fish, and 2,241 starfish in 3,568 images (Table 2). The recall rate roughly shows the coverage of the iterative labeling process. In this case, about 90% of the fish and sponges were detected and labeled, and over 83% of the sponges were well-labeled. Additionally, we trained an efficient model with an mAP of about 0.86, corresponding to these labels.

4 Conclusion

In this paper, we present a method for rapidly labeling large underwater datasets. We demonstrate that this method is robust, effective, and efficient for annotating a large number of images containing difficult classes. We began with a small initial dataset and utilized an iterative labeling process that gradually generates

TABLE 2 The Pacstorm dataset which consists of fish, starfish and sponges.

Initial dataset										
	Count			Recall			Precision			mAP/50
	Fish	Starfish	Sponge	Fish	Starfish	Sponge	Fish	Starfish	Sponge	All
	169	84	247	0.804	0.914	0.583	0.816	0.814	0.772	0.743
Iterative labeling process										
Loop	Count			Recall			Precision			mAP/50
	Fish	Starfish	Sponge	Fish	Starfish	Sponge	Fish	Starfish	Sponge	All
1	3306	2233	9225	0.864	0.957	0.696	0.907	0.981	0.801	0.822
2	3586	2238	10758	0.872	0.943	0.834	0.894	0.985	0.754	0.861
3	3788	2241	12660	0.881	0.938	0.828	0.909	0.975	0.727	0.860

To measure the recall and precision of trained model, we manually annotated a validation dataset of 302 images, with 611 fish, 210 starfish, and 1262 sponges.

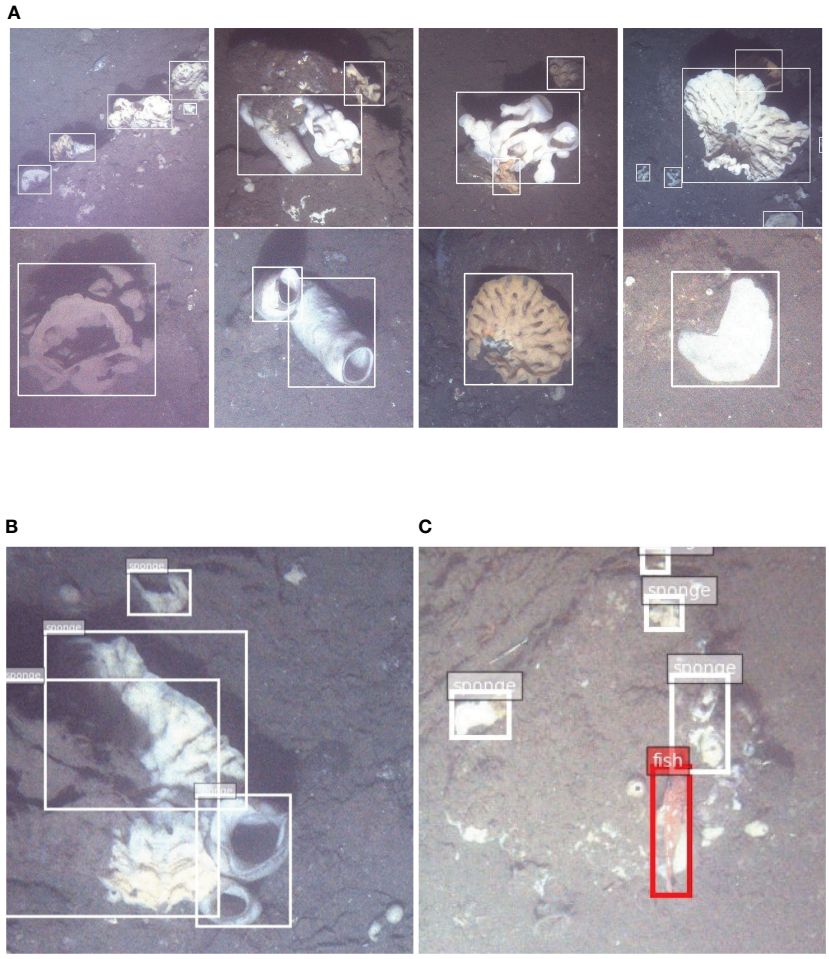


FIGURE 8
(A) Examples of different forms of sponges. Some sponges have a hole on top, while others do not. Some sponges look like white rocks, and others look like white dots. Sponges also come in different colors. While most of them are white, some are brown, and dead sponges are black. We presented these examples to the MTurk workers to help them identify the sponges. **(B)** Sometimes sponges are grouped together, which make it very hard to label them individually. **(C)** Some sponges are covered in mud, with only small part of them exposed. This make us hard to determine the labeling standard.

bounding box annotations. Our method results in a dataset with high coverage of rockfish, starfish, and sponge annotations after only a few iterations.

We first obtained the NOAA dataset, which only had dot annotations. We utilized MTurk workers to extend the dots to bounding boxes with the help of an assistive labeling interface. Then, we used these annotations as ground truth to validate our approach. We applied the iterative labeling process to 9,822 images and labeled 82% of the rockfish.

Next, we applied the same process to the empty Pacstorm dataset that we wanted to label, which included the challenging sponge class. After three iterations, we were able to label 90% of the fish and starfish and 83% of the sponges.

Both datasets are freely available for other researchers to use via our website. A direct link to the website is available in the Data Availability Statement below. We hope that this data, as well as the algorithm, can serve as a benchmark for validating various machine learning methodologies for marine biology related applications.

Author's note

Author AP was employed under contract at NOAA by Lynker Technologies at the time of researching this study.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession

number(s) can be found below: https://fieldroboticslab.ece.northeastern.edu/resources/.

Author contributions

PK contributed to the main conceptual ideas and designed the study. ZZ implemented the proposed method and performed experimental analysis on the datasets. HS, MC, and AP supervised the project and gave valuable feedback on the results. AP, EF, and MC provided the underwater fisheries datasets used in this study. EF helped verify the annotations generated by the proposed method by comparing them with manual annotations. ZZ took the lead in writing the manuscript and PK wrote a few sections in the manuscript. All authors contributed to the article and approved the submitted version.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Amin, R., Richards, B. L., Misa, W. F. X. E., Taylor, J. C., Miller, D. R., Rollo, A. K., et al. (2017). The modular optical underwater survey system (MOUSS) for in situ sampling of fish assemblages. *Sensors (Basel Switzerland)* 17, 1–8. doi: 10.3390/s17102309

Anantharajah, K., Ge, Z., McCool, C., Denman, S., Fookes, C., Corke, P., et al. (2014). "Local inter-session variability modelling for object classification," in *IEEE Winter Conference on Applications of Computer Vision*. 309–316.

Bhattacharjee, A., and Agrawal, M. (2021). Process design to use amazon mturk for cognitively complex tasks. *IT Prof.* 23, 56–61. doi: 10.1109/MITP.2020.2983395

Blue-Cloud (2019). *Ecotaxa*. Available at: https://blue-cloud.org/data-infrastructures/ecotaxa

Boom, B., He, J., Palazzo, S., Huang, P. X., Beyan, C., Chou, H.-M., et al. (2014). A research tool for long-term and continuous analysis of fish assemblage in coral-reefs using underwater camera footage. *Ecol. Inf.* 23, 83–97. doi: 10.1016/j.ecoinf.2013.10.006

Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., et al. (2010). Visual recognition with humans in the loop. in. *Eur. Conf. Comput. Vision*. doi: 10.1007/978-3-642-15561-1_32

Chen, Q., Beijbom, O., Chan, S., Bouwmeester, J., and Kriegman, D. J. (2021). "A new deep learning engine for coralnet," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 3686–3695.

Clement, R., Dunbabin, M. D., and Wyeth, G. (2005). Towards robust image detection of crown-of-thorns starfish for autonomous population monitoring. *Environ. Sci.*

Cox, T. E., Philippoff, J., Baumgartner, E. S., and Smith, C. M. (2012). Expert variability provides perspective on the strengths and weaknesses of citizen-driven intertidal monitoring program. *Ecol. Appl. Publ. Ecol. Soc. America* 22 (4), 1201–1212. doi: 10.1890/11-1614.1

Crowston, K. (2012). "Amazon Mechanical turk: a research tool for organizations and information systems scholars," in *Shaping the future of ICT research* (Berlin, Heidelberg: Springer Berlin Heidelberg), 210–221.

Cutter, G. R., Stierhoff, K., and Zeng, J. (2015). "Automated detection of rockfish in unconstrained underwater videos using haar cascades and a new image dataset: labeled fishes in the wild," in *2015 IEEE Winter Applications and Computer Vision Workshops*. 57–62.

Dawkins, M., Sherrill, L., Fieldhouse, K., Hoogs, A., Richards, B. L., Zhang, D. C., et al. (2017). "An open-source platform for underwater image and video analytics," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 898–906.

Deng, J., Russakovsky, O., Krause, J., Bernstein, M. S., Berg, A. C., and Fei-Fei, L. (2014). "Scalable multi-label annotation," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA: Association for Computing Machinery), 3099–3102.

Durden, J. M., Schoening, T., Althaus, F., Friedman, A., Garcia, R. V., Glover, A. G., et al. (2016). Perspectives in visual imaging for marine biology and ecology: from acquisition to understanding. *Oceanogr. Mar. Biol.* 54, 1–72. doi: 10.1201/9781315368597-2

Fisher, R. B., Shao, K.-T., and Chen-Burger, Y.-H. J. (2016). "Overview of the fish4knowledge project," in *Fish4Knowledge*.

Francis, R., Hixon, M. A., Clarke, M., Murawski, S., and Ralston, S. (2007). Ten commandments for ecosystem-based fisheries scientists. *Fisheries* 32, 217–233. doi: 10.1577/1548-8446(2007)32[217:TCFBFS]2.0.CO;2

Gleason, A. C. R., Reid, R. P., and Voss, K. J. (2007). Automated classification of underwater multispectral imagery for coral reef monitoring. *OCEANS* 2007, 1–8. doi: 10.1109/OCEANS.2007.4449394

He, K., Zhang, X., Ren, S., and Sun, J. (2015). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

Kaeli, J. W., Singh, H., Murphy, C., and Kunz, C. (2011). "Improving color correction for underwater image surveys," in *OCEANS'11 MTS/IEEE KONA*. 1–6.

Katija, K., Orenstein, E. C., Schlining, B., Lundsten, L., Barnard, K., Sainz, G., et al. (2021). Fathomnet: a global image database for enabling artificial intelligence in the ocean. *Sci. Rep.* 12, 15914. doi: 10.1038/s41598-022-19939-2

Kaufmann, N., Schulze, T., and Veit, D. J. (2011). "More than fun and money. worker motivation in crowdsourcing - a study on mechanical turk," in *Americas conference on information systems*.

Kaveti, P., and Akbar, M. N. (2020). "Role of intrinsic motivation in user interface design to enhance worker performance in amazon mturk," in *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*.

Kaveti, P., and Singh, H. (2018). "Towards automated fish detection using convolutional neural networks," in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*. 1–6.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Commun. ACM* 60, 84–90. doi: 10.1145/3065386

Langenkämper, D., Simon-Lledó, E., Hosking, B., Jones, D. O. B., and Nattkemper, T. W. (2019). On the impact of citizen science-derived data quality on deep learning based classification in marine images. *PLoS One* 14. doi: 10.1371/journal.pone.0218086

Langenkämper, D., Zurowietz, M., Schoening, T., and Nattkemper, T. W. (2017). Biigle 2.0 - browsing and annotating large marine image collections. *Front. Mar. Sci.* 4. doi: 10.3389/fmars.2017.00083

Lin, T.-Y., Dollár, P., Girshick, R. B., He, K., Hariharan, B., and Belongie, S. J. (2016). "Feature pyramid networks for object detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 936–944.

Litman, L., Robinson, J., and Rosenzweig, C. (2015). The relationship between motivation, monetary compensation, and data quality among us- and india-based workers on mechanical turk. *Behav. Res. Methods* 47, 519–528. doi: 10.3758/s13428-014-0483-x

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S. E., Fu, C.-Y., et al. (2015). "Ssd: single shot multibox detector," in *European Conference on Computer Vision* (Cham: Springer International Publishing), 21–37.

Marburg, A., and Bigham, K. (2016). "Deep learning for benthic fauna identification," in *OCEANS 2016 MTS/IEEE Monterey*. 1–5.

Matabos, M., Hoeberechts, M., Doya, C., Aguzzi, J., Nephin, J., Reimchen, T. E., et al. (2017). Expert, crowd, students or algorithm: who holds the key to deep-sea imagery 'big data' processing? *Methods Ecol. Evol.* 8, 996–1004. doi: 10.1111/2041-210X.12746

Purser, A., Bergmann, M., Lundälv, T., Ontrup, J., and Nattkemper, T. W. (2009). Use of machine-learning algorithms for the automated detection of cold-water coral habitats: a pilot study. *Mar. Ecol. Prog. Ser.* 397, 241–251. doi: 10.3354/meps08154

Ramani, N., and Patrick, P. H. (1992). "Fish detection and identification using neural networks-some laboratory results," in *IEEE Journal of Oceanic Engineering*, Vol. 17. 364–368.

Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). "Faster r-cnn: towards real-time object detection with region proposal networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39. 1137–1149.

Russakovsky, O., Li, L.-J., and Fei-Fei, L. (2015). "Best of both worlds: human-machine collaboration for object annotation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2121–2131.

Saleh, A., Laradji, I. H., Konovalov, D. A., Bradley, M., Vázquez, D., and Sheaves, M. (2020). A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Sci. Rep.* 10, 14671. doi: 10.1038/s41598-020-71639-x

Simonyan, K., and Zisserman, A. (2014). "Very deep convolutional networks for large-scale image recognition," in *CoRR abs/1409*. 1556.

Simpson, R. J., Page, K. R., and Roure, D. C. D. (2014). "Zooniverse: observing the world's largest citizen science platform," in *Proceedings of the 23rd International Conference on World Wide Web*.

Singh, H., Armstrong, R. A., Gilbes, F., Eustice, R. M., Roman, C., Pizarro, O., et al. (2004a). Imaging coral i: imaging coral habitats with the seabed auv. *Subsurface Sens. Technol. Appl.* 5, 25–42. doi: 10.1023/B:SSTA.0000018445.25977.f3

Singh, H., Can, A., Eustice, R. M., Lerner, S., McPhee, N. M., and Roman, C. (2004b). "Seabed auv offers new platform for high-resolution imaging," in *Eos, Transactions Am. Geophys. Union* Vol. 85. 289–296.

Smith, D. V., and Dunbabin, M. D. (2007). "Automated counting of the northern pacific sea star in the derwent using shape recognition," in *9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications (DICTA 2007)*. 500–507.

Sung, M., Yu, S.-C., and Girdhar, Y. A. (2017). "Vision based real-time fish detection using convolutional neural network," in *OCEANS 2017*, Aberdeen. 1–6.

Taylor, R., Vine, N., York, A., Lerner, S., Hart, D., Howland, J. C., et al. (2008). "Evolution of a benthic imaging system from a towed camera to an automated habitat characterization system," in *OCEANS 2008*. 1–7.

Tolimieri, N., Clarke, M., Singh, H., and Goldfinger, C. (2008). Evaluating the seabed auv for monitoring groundfish in untrawlable habitat. *Mar. Habitat Mapping Technol. Alska* doi: 10.4027/mhmta.2008.09

Vijayanarasimhan, S., and Grauman, K. (2008). "Multi-level active prediction of useful image annotations for recognition," in *NIPS*.

Wah, C., Branson, S., Perona, P., and Belongie, S. J. (2011). "Multiclass recognition and part localization with humans in the loop," in *2011 International Conference on Computer Vision*. 2524–2531.

Wang, H., Sun, S., Bai, X., Wang, J., and Ren, P. (2023). A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes. *IEEE J. Oceanic Eng.* 48, 443–461. doi: 10.1109/JOE.2022.3226202

Wang, H., Sun, S., Wu, X., Li, L., Zhang, H., Li, M., et al. (2021). "A yolov5 baseline for underwater object detection," in *OCEANS 2021*, San Diego – Porto. 1–4.

Wu, Y., and He, K. (2018). "Group normalization," in *International Journal of Computer Vision*, Vol. 128. 742–755.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019) *Detectron2*. Available at: https://github.com/facebookresearch/detectron2.

Yu, F., Zhang, Y., Song, S., Seff, A., and Xiao, J. (2015). Lsun: construction of a large-scale image dataset using deep learning with humans in the loop. *ArXiv*. bs/1506.03365. doi: 10.48550/arXiv.1506.03365

yu Wang, H., Sun, S., and Ren, P. (2023). "Meta underwater camera: a smart protocol for underwater image enhancement," in *ISPRS Journal of Photogrammetry and Remote Sensing*. 462–481. Available at: https://www.sciencedirect.com/science/article/pii/S0924271622003227.

?>Zhuang, P., Wang, Y., and Qiao, Y. (2021). "Wildfish++: a comprehensive fish benchmark for multimedia research," in *IEEE Transactions on Multimedia*, Vol. 23. 3603–3617.

Zurowietz, M., Langenkämper, D., Hosking, B., Ruhl, H. A., and Nattkemper, T. W. (2018). Maia–a machine learning assisted image annotation method for environmental monitoring and exploration. *PLoS One* 13. doi: 10.1371/journal.pone.0207498

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 31 May 2023
DOI 10.3389/fmars.2023.1163831

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Yuan Zhou,
Tianjin University, China
Shenghui Rong,
Ocean University of China, China

*CORRESPONDENCE
Huifang Xu
17069@gench.edu.cn

RECEIVED 11 February 2023
ACCEPTED 09 May 2023
PUBLISHED 31 May 2023

CITATION
Song W, Liu Y, Huang D, Zhang B, Shen Z and Xu H (2023) From shallow sea to deep sea: research progress in underwater image restoration. *Front. Mar. Sci.* 10:1163831. doi: 10.3389/fmars.2023.1163831

COPYRIGHT
© 2023 Song, Liu, Huang, Zhang, Shen and Xu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

From shallow sea to deep sea: research progress in underwater image restoration

Wei Song[1], Yaling Liu[1], Dongmei Huang[2], Bing Zhang[3], Zhihao Shen[1] and Huifang Xu[4]*

[1]Digital Ocean Laboratory, Shanghai Ocean University, Shanghai, China, [2]College of Electronics and Information Engineering, Shanghai University of Electric Power, Shanghai, China, [3]Institute of Deep Sea Science and Engineering, Chinese Academy of Sciences, Sanya, Hainan, China, [4]College of Information Technology, Shanghai Jian Qiao University, Shanghai, China

Underwater images play a crucial role in various fields, including oceanographic engineering, marine exploitation, and marine environmental protection. However, the quality of underwater images is often severely degraded due to the complexities of the underwater environment and equipment limitations. This degradation hinders advancements in relevant research. Consequently, underwater image restoration has gained significant attention as a research area. With the growing interest in deep-sea exploration, deep-sea image restoration has emerged as a new focus, presenting unique challenges. This paper aims to conduct a systematic review of underwater image restoration technology, bridging the gap between shallow-sea and deep-sea image restoration fields through experimental analysis. This paper first categorizes shallow-sea image restoration methods into three types: physical model-based methods, prior-based methods, and deep learning-based methods that integrate physical models. The core concepts and characteristics of representative methods are analyzed. The research status and primary challenges in deep-sea image restoration are then summarized, including color cast and blur caused by underwater environmental characteristics, as well as insufficient and uneven lighting caused by artificial light sources. Potential solutions are explored, such as applying general shallow-sea restoration methods to address color cast and blur, and leveraging techniques from related fields like exposure image correction and low-light image enhancement to tackle lighting issues. Comprehensive experiments are conducted to examine the feasibility of shallow-sea image restoration methods and related image enhancement techniques for deep-sea image restoration. The experimental results provide valuable insights into existing methods for addressing the challenges of deep-sea image restoration. An in-depth discussion is presented, suggesting several future development directions in deep-sea image restoration. Three main points emerged from the research findings: i) Existing shallow-sea image restoration methods are insufficient to address the degradation issues in deep-sea environments, such as low-light and uneven illumination. ii) Combining imaging physical models with deep learning to restore deep-sea image quality may potentially yield desirable results. iii) The application potential of unsupervised and zero-shot learning methods in deep-sea image restoration warrants further investigation, given their ability to work with limited training data.

KEYWORDS

shallow-sea image restoration, deep-sea image restoration, image formation, physical model, prior, deep learning

1 Background

The ocean contains many unknown organisms and vast energy sources, which play an important role in sustaining life on earth. The exploitation of marine resources, the development of the marine economy, and the strengthening of the marine industry have become integral components of countries' strategic planning and progress. Underwater image processing is essential for ocean exploration; however, the complexity of the marine environment often leads to severely degraded image quality. The differing rates of light attenuation at various wavelengths in the ocean cause images to predominantly appear blue–green. In addition, microorganisms and suspended particles in the water absorb most of the light energy and deflect its direction, resulting in low-contrast and blurred images. These factors significantly impact the efficacy of many underwater vision systems. Image restoration is a technique that involves reversing the imaging process used to produce low-quality images. Underwater image restoration technology aims to enhance image visibility, eliminate color casts, and stretch contrast to effectively improve the visual quality of input images, thereby increasing the efficiency of underwater operations. Furthermore, the restored images highlight scenes and objects, thus serving as a preprocessing step in underwater image research. This can facilitate advanced tasks, such as target detection, recognition, and classification, and ultimately improve the observation and processing of underwater information.

In contrast to images taken on land, images taken by underwater imaging systems often suffer from low contrast, loss of detail, color distortion, low light or non-uniform illumination, and reduced visual ranges as a result of the influence of complex underwater imaging environments and lighting environments. The degradation of underwater images has caused great inconvenience to practical applications and further research. The principle of underwater optical imaging can be seen in Figure 1. The attenuation of light under water is primarily caused by absorption and scattering effects, leading to degraded image quality such as reduced contrast and blurriness. In addition, different wavelengths of light have varying rates of attenuation when traveling underwater, which results in color distortion in the images. In clear water, red light is the first to disappear, at a depth of 5 meters, followed by orange light at 10 meters. Blue light, with the shortest visible wavelength, can travel the farthest in water, which causes underwater images to have an undesirable blue–green hue. The presence of small particles, plankton, and dissolved organic matter in the water frequently causes significant noise issues in underwater imaging and exacerbates the impact of backscattering.

The deep sea, broadly defined as the depth of the ocean where natural light does not penetrate (NOAA, 2022), is characterized by extreme conditions such as low temperatures, darkness, and high pressure, making exploration difficult (Paulus, 2021). Remote-operated vehicles (ROVs) equipped with underwater optical photography technology become an indispensable means of deep-sea exploration. However, images captured in the depths of the dark ocean using artificial light sources are subject to a combination of light attenuation, scattering interference, and uneven illumination, resulting in images with strong halo effects that are less clear than those taken in shallower waters. Therefore, improving the quality of deep-sea images and extracting more useful information from them

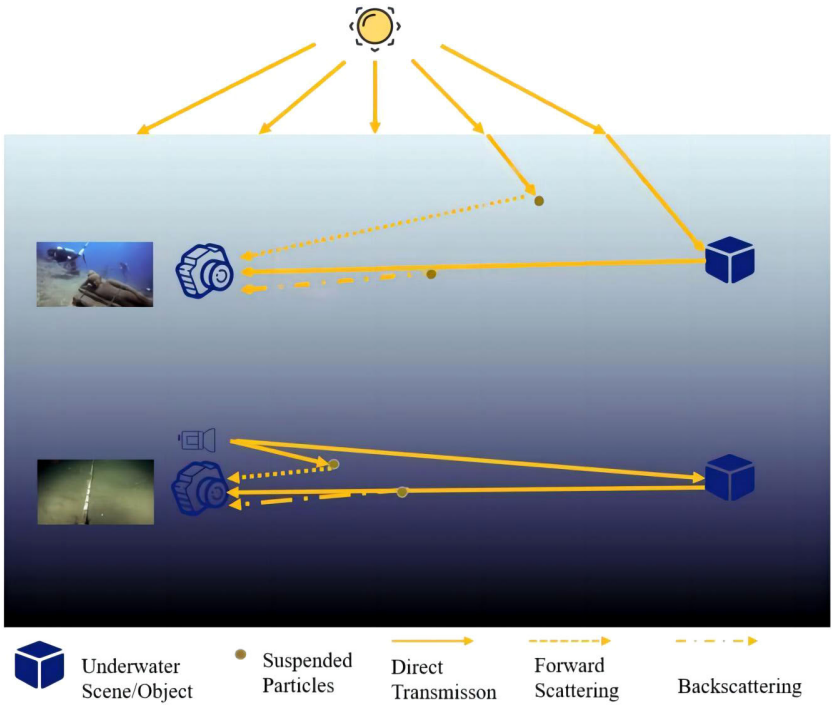


FIGURE 1
Schematic diagram of underwater optical imaging.

is vital to promote deep-sea exploration and to discover new deep-sea phenomena.

In the research transition from shallow-sea to deep-sea image restoration methods, the core issue is the composition of the light source in the underwater imaging process. Although natural light alone or in combination with artificial light can serve as the light source in shallow-sea imaging, artificial light sources are essential in deep-sea imaging because of the absence of natural light. Artificial light sources have different characteristics from natural light and can result in non-uniform lighting, creating bright spots in the middle of the light source and dark spots around the edges of deep-sea images. Furthermore, inherent image degradation problems arise because of the absorption and scattering of light source propagation in artificial light sources.

Numerous studies have been developed to improve the quality of underwater images (Ancuti et al., 2018; Anwar and Li, 2020; Wang et al., 2022). A majority focused on designing direct image enhancement techniques or networks without taking the principles of underwater imaging into account. Others concentrated on developing underwater image restoration techniques that reverse the underwater imaging process to recover the original image. This study focuses on underwater image restoration rather than enhancement for two reasons. First, non-physical model-based underwater image enhancement methods can enhance the visual quality of images to some degree but do not consider the unique optical characteristics of underwater imaging, resulting in color distortions, artifacts, and increased noise. Second, the effectiveness of deep learning-based underwater image enhancement techniques depends heavily on the quality of the training data used. However, obtaining suitable datasets, particularly for deep-sea environments, remains a significant challenge owing to their scarcity. Although various reviews of underwater image enhancement (Wang et al., 2019b; Anwar and Li, 2020; Fayaz et al., 2021) exist, there is still a lack of systematic overview to bridge the gap between shallow-sea studies and deep-sea studies.

After a systematic review, this research paper summarizes the challenges and advanced solutions for shallow-sea image restoration to provide a reliable reference for researchers in the related fields. The study then shifts its focus to deep-sea image restoration, summarizing the difficulties faced in this field, examining the connections and differences between shallow-sea and deep-sea image restoration research, exploring fields, such as exposure and low-light image enhancement, and summarizing feasible methods for deep-sea image recovery. The contributions of this study are as follows.

(1) This study categorizes recent methods for restoring shallow-sea images into three groups, physical model-based methods, prior-based methods, and deep learning-based methods, which integrate physical models. It offers an in-depth analysis of the fundamental concepts and essential features of these techniques, and provide a comprehensive overview of their classification.

(2) This study provides an overview of the latest research advancements, challenges, and promising research directions in deep-sea image restoration. Considering two causes of the degradation of deep-sea images, the deep-sea environment and artificial light sources, this study reviews the related research for potential solutions to these problems. Techniques for shallow-sea image restoration provide valuable insights for addressing degradation issues arising from underwater environments, such as color cast and blur. The degradation problem caused by artificial light sources has been approached with solutions such as layer decomposition and the integration of deep learning and physical models.

(3) Experiments have been carried out extensively to assess the effectiveness of shallow-sea image restoration, low-light image enhancement, and exposure correction techniques in handling deep-sea images. The findings reveal that, although shallow-sea images have improved in color correction to some extent, the issue of image light sources has become more pronounced, and some prior techniques have not been effective in deep-sea environments. On the other hand, low-light image enhancement and exposure correction can improve uniform illumination and increase brightness; however, they also come with drawbacks such as worsening color cast. Using the results of the analysis, this study discusses the key scientific challenges that need to be addressed in the field of underwater image restoration, from shallow-sea to deep-sea image restoration, and provides insight into potential future research directions.

2 Shallow-sea image restoration methods

In general, restoration techniques model the degradation and apply an inverse process to recover the original image. Therefore, research on underwater image restoration focuses initially on the development of a physical model that conforms to the principle of underwater image formation. Although a more comprehensive imaging model can be obtained by taking into account various factors that influence the imaging process, a simpler model can often be applied to a wider range of scenarios. Underwater image restoration is based on prior knowledge from degradation principles or statistical data.

In this section, underwater image restoration methods are classified into three categories. The first category focuses on building a physical model that is aligned with the principle of underwater image formation. The second category utilizes prior knowledge from degradation principles or statistical data to make more accurate estimates of unknowns in the imaging model. The third category is a combination of an underwater imaging physical model and a deep learning approach for underwater image restoration.

2.1 Physical model-based shallow-sea image restoration methods

Currently, the image formation models (IFMs) employed in the field of underwater image restoration are one of four types: the

atmospheric light scattering (Koschmieder) model (Koschmieder, 1924), the simplified underwater formation model, the revised underwater formation model, of which the Akkaynak–Treibitz model (Akkaynak et al., 2017) is the most widely used, and the Retinex model.

2.1.1 Koschmieder model

The Koschmieder model is an imaging model that accurately explains the principle of image degradation caused by atmospheric conditions through physical analysis (Koschmieder, 1924). As a result, it has been applied to various fields such as underwater image restoration, restoration of foggy images, and low-light image enhancement. The Koschmieder model can be described as:

$$I(x) = J(x)t(x) + A(1 - t(x)). \tag{1}$$

$$t(x) = \exp(-\beta d(x)). \tag{2}$$

In the Koschmieder model, I and J represent the degraded and undegraded underwater images captured by the camera, respectively, A denotes the background light, and t denotes the transmittance.

Lu et al. (2015) developed a simplified underwater imaging model that takes into account the combined effects of both natural and artificial light sources. They used an energy attenuation model to describe the lighting, and the model can be formulated as follows:

$$E_W^c(x) = E_L^c(x) + E_A^c(x), c \in \{R, G, B\}. \tag{3}$$

The $E_W^c(x)$, $E_L^c(x)$, and $E_A^c(x)$ illuminances represent the total illuminance, natural light source, and artificial light source, respectively. By incorporating the Koschmieder model, a new imaging model formula has been derived:

$$I^c(x) = \left((E_A^c(x) \cdot N_{\text{rer}}(c)^{D(x)} + E_L^c(x) \cdot N_{\text{rer}}(c)^{d(x)}) \cdot \rho^c(x)\right) \times t^c(x) + (1 - t^c(x))A^c, c \in \{R, G, B\}. \tag{4}$$

The Koschmieder model is a useful tool for accurately describing the physical degradation of images and has been widely applied in various fields, including low-light image enhancement, image dehazing, and underwater image restoration. However, the model has some limitations. Specifically, it considers only the effects of absorption and scattering on the imaging process, while ignoring other factors that can lead to significant image degradation, such as the absorption of different wavelengths of light by water.

2.1.2 Simplified underwater image formation model

Many physical model-based methods in underwater image restoration rely on simplified models and their derivatives. In accordance with the principle of underwater imaging, light is affected by absorption and scattering in water, resulting in the degradation of underwater images such as blue-green cast and blur. The formation of an underwater image is often considered a linear combination of direct transmission E_d, backscattering E_b, and forward scattering components E_f, as described below:

$$E_t = E_d + E_b + E_f. \tag{5}$$

In underwater image restoration research, the direct transmission component and backscattering component are typically considered the key parts, whereas the forward scattering component is usually difficult to obtain and has a relatively minor impact on the formation of underwater images, and, thus, is often neglected. A simplified underwater image formation model (IFM) (Narasimhan and Nayar, 2000; Fattal, 2008; Narasimhan and Nayar, 2008) is used to mathematically simulate the underwater degradation process, which can be expressed as:

$$I^c(x) = J^c(x)t^c(x) + A^c(1 - t^c(x)), c \in \{R, G, B\}, \tag{6}$$

where I represents the underwater degraded image, J represents the undegraded image captured by the camera, A represents the background light, c represents the red, green, and blue (RGB) color channel, and t represents the transmittance according to the light attenuation law, which can be further expressed as the attenuation index (Zhao et al., 2015):

$$t^c(x) = \exp(-\beta^c d(x)), \tag{7}$$

where d represents the water depth and β is the attenuation coefficient. The mathematical expression of the IFM is very similar to that of the Koschmieder model. Even so, we still consider the IFM an independent part for two reasons: (1) the Koschmieder model is an "accurate" description of the imaging process in the atmosphere, whereas the IFM is a "simulation" of the underwater imaging process under the analysis of the underwater environment and certain assumptions; and (2) research based on the Koschmieder model is often used to develop a new physical model of underwater imaging, whereas research based on the IFM is used to estimate the transmission map and background light more accurately under specific prior conditions in order to obtain a restored image with enhanced quality.

Not all underwater scenes can be effectively modeled using the simplified underwater IFM. To address the issue of water types and artificial light source interference in underwater images, Chiang and Chen (2012) considered the difference between the attenuation of different light wavelength and adjusted the normalized residual energy ratio Nrer based on that of Ocean Type I (extremely clear waters) as follows:

$$\text{Nrer}(\lambda) = \begin{cases} 0.8 - 0.85 \text{ if } \lambda = 650 - 750\ \mu m\ (\text{R}), \\ 0.93 - 0.97 \text{ if } \lambda = 490 - 550\ \mu m\ (\text{G}), \\ 0.95 - 0.99 \text{ if } \lambda = 400 - 490\ \mu m\ (\text{B}), \end{cases} \tag{8}$$

where λ is the wavelength. In underwater scenes, there is a relationship between the transmittance and the normalized residual energy ratio:

$$t^c(x) = \text{Nrer}(c), c \in \{R, G, B\}. \tag{9}$$

Then, the underwater imaging model considering the artificial light source, blur, and wavelength attenuation can be expressed as:

$$I_c(x) = \left((E_c^A(x) \cdot Nrer(c)^{D(x)} + E_C^L \cdot Nrer(c)^{d(x)}) \cdot \rho_c(x) \right) \cdot Nrer(c)^{d(x)} + (1 - Nrer(c)^{d(x)}) \cdot A^c, c \in \{R, G, B\} \quad (10)$$

Simplified underwater IFMs are widely used in shallow-sea image restoration research and have achieved satisfactory results. However, they have significant limitations in deep-sea image restoration research. The simplified underwater IFM attributes the degradation of underwater images to three factors: the absorption and scattering characteristics of water, the distance between the target and the camera, and the geometric angle between the light source, the camera, and the target. It is an approximate model derived by reverse-deriving the degradation process through computer simulation of the underwater imaging process, neglecting the forward scattering component. In the deep-sea environment, the forward scattering component is a crucial factor that cannot be ignored, and the composition of the imaging light source differs significantly from that in the shallow-sea environment. Therefore, computer simulations based on shallow-sea imaging environments cannot accurately describe the degradation of images in deep-ocean environments.

Moreover, the simplified models used in the field of underwater image restoration are based on the assumption that the light sources is parallel natural light, such as sunlight. Although a few models consider the presence of artificial light sources during the imaging process, they are often considered auxiliary light sources with negligible effects on imaging. However, in the deep-sea environment, without natural light, an artificial light source with a bright center and dark surroundings becomes the only light source for imaging, resulting in an inaccurate description of the degradation process of deep-sea images by underwater imaging models. Furthermore, the deep-sea environment is different from the shallow-sea environment, and the absorption and scattering of light in deep-sea environments differ from those in general shallow-water environments. Therefore, simplified underwater imaging models are not suitable for deep-sea image enhancement and restoration.

2.1.3 Akkaynak–Treibitz model

The Akkaynak–Treibitz model is proposed as an alternative to the IFM model currently used in underwater image restoration. Akkaynak et al. (2017) conducted *in situ* experiments in the Red Sea and the Mediterranean Sea, and found that attenuation coefficients of light depend on the imaging range and object reflectivity. The study also quantified the error arising from neglecting such dependencies. Building on these findings, Akkaynak and Treibitz (2018) proposed a revised underwater physical imaging model, as expressed in Equation 11. In the revised model, the attenuation coefficients of the direct transmission component and the backscattering component are different, and the relationship between the distance between the camera and the target and the direct transmission component is mainly investigated:

$$I^c(x) = J^c(x)e^{-\beta_c^D(v_D)z} + A^c(1 - e^{-\beta_c^B(v_B)z}), c \in \{R, G, B\}, \quad (11)$$

where v_D and $v_B$are both vectors and $v_D = \{z, \rho, E, Sc, \beta'\}$ and $v_B = \{E, Sc, b, \beta'\}$, z represents the distance between the camera and the target, ρ represents the reflectivity, E is the irradiance, Sc is the camera response function, β' is the light scattering coefficient, and b is the physical scattering attenuation coefficient of the water body.

Subsequently, Akkaynak and Treibitz identified a functional dependence between the direct transmission attenuation coefficient β_c^D and the camera–target distance z , as described in Equation 12. They proposed the "sea-thru" underwater image restoration method (Akkaynak and Treibitz, 2019) based on this relationship, along with a practical approach for estimating the parameters of the corrected model:

$$\beta_c^D(z) = a \times \exp(b \times z) + c \times \exp(d \times z), \quad (12)$$

where a and c are coefficients related to the type of water body and their values can be calculated based on the relevant data measured on site, and d is the depth of the water.

The Akkaynak–Treibitz model can be regarded as an enhancement of the simplified underwater IFM through optimization. This entails introducing non-uniform attenuation coefficients for the direct transmission component and backscattering transmission component and establishing distinct correlations between the two-component attenuation coefficient and the camera–target distances. Although the Akkaynak–Treibitz model has been further confirmed by many scholars in the field of shallow-sea image restoration and has led to the development of effective shallow-sea image restoration methods, it is still an approximate model simulating the imaging process of shallow-sea degradation.

2.1.4 Retinex model

The Retinex theory (Land and McCann, 1971; Land, 1977) is an effective method for addressing complex lighting issues in images. It can balance dynamic range compression, edge enhancement, and color preservation in image processing. Many researchers have applied it to the fields of underwater image enhancement and restoration. The implementation of Retinex requires certain assumptions, such as that the color of objects as seen by the human eye is the result of the object's reflection of light under different conditions, and that all colors in nature are composed of fixed wavelengths of the three primary colors, red, green, and blue. Meanwhile, the color of objects in the real world depends solely on the object's reflection properties and is not affected by the non-uniformity of lighting, resulting in color constancy.

Based on the Retinex theory, the Retinex model (Land and McCann, 1971; Land, 1977) is represented by the following equation:

$$S(x) = L(x) \cdot R(x), \quad (13)$$

where $L(x)$ represents the illumination component, background information, or global information, $R(x)$ represents the reflectance component or the attributes of the photographed object, $S(x)$ represents the observed image, $x$represents the pixel, and the symbol " $\cdot$} denotes pixel multiplication.

The Retinex model has achieved good results in the fields of underwater image enhancement and low-light image enhancement. Kimmel et al. (2003) first proposed an optimized algorithm for the Retinex model based on a variational framework, which has inspired the development of methods based on a variational framework to address the problem of underwater image degradation. Zhuang et al. (2021) proposed a Bayesian optimization algorithm for a single-frame underwater imaging model based on multiorder gradient priors for reflectance and illuminance enhancement, without the need for additional prior knowledge of underwater imaging. Later, Zhuang et al. (2022), proposed a modified variational model with different reflectance and illumination priors that are independent of prior knowledge of underwater imaging.

Based on the Retinex theory, Zhang and Peng (2018) proposed to use the global background light color as the light source color to restore the underwater image color, and proposed an imaging model that considered both the underwater imaging degradation principle and the light source characteristics, as follows:

$$I^c(x) = L^c M^c(x) t^c(x) + L^c(1 - t^c(x)), c \in \{R, G, B\}, \quad (14)$$

where L is the light source color and M is the surface reflectance.

The Retinex model differs significantly from the three physical imaging models mentioned earlier. Most shallow-sea image restoration methods that utilize the Retinex model achieve accurate estimation of both the illumination and reflection components through different mathematical derivations. Such methods have the advantage of being faster, but often require additional prior knowledge of underwater imaging and thus are subject to the limitations of prior knowledge. Therefore, the Retinex shallow-sea image restoration method without additional prior knowledge cannot guarantee good results in deep-sea image restoration.

To sum up, the physical imaging model applied in shallow-sea image restoration lacks generalizability in deep-sea image restoration. Therefore, it is necessary and feasible to construct deep-sea imaging physics based on the environmental characteristics of the deep sea and the light source characteristics of deep-sea imaging combined with deep-sea-collected images.

2.2 Prior-based shallow-sea image restoration methods

Based on prior knowledge, the unknown quantities in the physical model, transmission map and background light, are estimated more accurately.

He et al. (2011) introduced the dark channel prior (DCP) method for dehazing natural land images by leveraging the fog imaging model. They creatively solved the problem of dehazing natural land images by estimating background light and transmission maps. The DCP method is based on a statistical prior known as the dark channel, which is derived from the observation that, in most outdoor haze-free images, pixels in non-sky regions have at least one color channel with very low luminance values. The dark channel is defined as follows:

$$J^{dark}(x) = \min_{c \in \{r,g,b\}} \left(\min_{y \in \Omega(\times)} (J^c(y)) \right). \quad (15)$$

Based on this statistical prior, the estimation of ambient light was suggested by selecting the brightest points in the top 0.1% of the dark channel of the observed image, and the transmission map could be calculated using the following formula:

$$\tilde{t}(x) = 1 - \omega \min_c \left(\min_{y \in \Omega(x)} \left(\frac{I^c(y)}{A^c} \right) \right), \quad (16)$$

where the variable $\omega (0 < \omega \leq 1)$ is used to make the restored image more realistic. A value of 0.95 is typically employed for ω.

Although the DCP method is not effective when applied directly to underwater images, it has inspired many other underwater image restoration methods (Hautière et al., 2008; Carlevaris-Bianco et al., 2010). The underwater dark channel prior (UDCP) method accounts for the fact that water absorbs different wavelengths of light differently, with the transmission distance of red light being shorter. Drews et al. (2016) found that, although the DCP method fails in the red channel of underwater images, the blue and green channels are still suitable for the DCP method. Consequently, they applied the DCP method to the blue-green channel of a degraded underwater image, resulting in significant improvement in the restored image. Galdran et al. (2015) have proposed the red channel prior (RCP) method, as shown in Equation 17, which restores the color of shortwave-related underwater images based on the red wavelength with the fastest attenuation. These methods can be considered variants of the DCP method:

$$J^{RED}(x) = min\left(\min_{y \in \Omega(x)} (1 - J^R(y)), \min_{y \in \Omega(x)} (J^G(y)), \min_{y \in \Omega(x)} (J^B(y)) \right). \quad (17)$$

As the RCP method is effective in restoring artificially illuminated areas of underwater images, Zhou et al. (2021a) combined the RCP method with a quadratic guidance filter to refine the transmission map in underwater image restoration. Chiang and Chen (2012) corrected the color of underwater images by compensating for the attenuation of different colors of light along the propagation path and used the DCP method to achieve defogging. Peng et al. (2018) proposed the generalized dark channel prior (GDCP) method, which estimates ambient light through depth-dependent color changes, and calculates the scene transmission through the difference between the observed value and the estimated value. This method applies to a wide range of scenarios. Li et al. (2016b) proposed a new underwater dark channel prior model that combines the grayscale world assumption to achieve blue-green channel dehazing and red channel color correction, and used an adaptive exposure map to adjust the color of the image. Gao et al. (2016) proposed the bright

channel prior (BCP) method, which is suitable for underwater images and can restore underwater images by estimating background light and transmission map through the bright channel, drawing on prior knowledge of the dark channel.

In contrast to the DCP method, the maximum intensity prior (MIP) method (Carlevaris-Bianco et al., 2010) uses the attenuation difference between the three color channels of an underwater image to estimate the depth of the scene and restore the image. The MIP method involves comparing the maximum intensity of the red channel with the maximum intensity of the green and blue channels on a small image patch. It then calculates the difference between the maximum intensity of the red channel and the maximum intensity of the green and blue channels using the following formula:

$$D(x) = \max_{x\in\Omega, c\in R} I^c(x) - \max_{x\in\Omega, c\in\{B,G\}} I^c(x). \tag{18}$$

Here, the transmission at the point x is estimated by the following formula:

$$\tilde{t}(x) = D(x) + \left(1 - \max_x D(x)\right). \tag{19}$$

Wang et al. (2017) proposed the maximum attenuation identification (MAI) method, which is based on a simple prior knowledge of underwater imaging: that the intensity of light decays as an exponential function of distance. They rewrote the simplified underwater imaging model as follows:

$$I(x) = J(x)\xi(x) + A(1 - \xi(x)), \tag{20}$$

and, further, estimated the attenuation ξ as:

$$\xi \to 1 - \frac{1 - max_{y\in\Omega(x)}\left(I^R(y)\right)}{1 - A^R(x)}. \tag{21}$$

Peng et al. (2015) observed that in underwater images the scenes that are farther away from the camera appear more blurred. Based on this observation, they proposed a blur prior (BP) to estimate the distance between the scene point and the camera in order to obtain the depth map of the underwater image and then restore the degraded image. This method is effective under different lighting conditions. Peng and Cosman (2017) later proposed a new method called image blurriness and light absorption (IBLA), which takes into account the absorption characteristics of underwater light and further optimizes the estimation of the depth map and background light. They proposed a new hypothesis that scene points that retain more red light in the red channel map are closer to the camera, which is used to estimate the depth map $\widetilde{d_R}$, as expressed in the following formula:

$$\widetilde{d_R} = 1 - F_s(R), \tag{22}$$

where F_s is a stretching function:

$$F_s(V) = \frac{V - \min(V)}{\max(V) - \min(V)}, \tag{23}$$

where V is a vector, which can represent the red channel R, the MIP, and the BP. The final depth map of IBLA is obtained by combining the three estimated depth maps.

The principle of the minimum information loss prior (MILP) states that the underwater imaging model can be mapped from the transmission map to the undegraded image; however, the input value range is $[0-255]$ and its effective mapping range is $[\alpha-\beta]$. Li et al. (2016a) proposed an effective underwater image dehazing algorithm that combines the MILP to restore the visibility, color, and natural appearance of underwater images. They also proposed a simple but effective contrast ratio enhancement algorithm based on the histogram prior, which improves the contrast and brightness of underwater images.

Song et al. (2018) proposed the underwater light attenuation prior (ULAP) method based on the observation of a large number of underwater images. The calculation of the depth map using the ULAP method is as follows:

$$d(x) = \mu_0 + \mu_1 m(x) + \mu_2 v(x). \tag{24}$$

In this formula, m represents the maximum value of the blue-green channel intensity and v represents the intensity value of the red channel.

Inspired by the color-line algorithm for land image dehazing (Fattal, 2014), Berman et al. (2016) found that by clustering the pixels of haze-free color images using k-means, each color cluster in the RGB space was distributed along a straight line, which they called the haze line. They used this discovery to achieve image depth map estimation and haze-free image restoration. Later, Menaker et al. (2017) introduced the haze line into the field of underwater image restoration and restored the image by combining the blue-to-green and blue-to-red channel attenuation ratio and the extracted parameters in the existing water-type library. They also chose the best-restored image based on the grayscale world assumption. Berman et al. (2020) further optimized the method by automatically selecting the best-restored image based on the color distribution of the underwater image. Bekerman et al. (2020) proposed a robust underwater image restoration algorithm that estimates attenuation from image color distribution and estimates veiling light from scene objects based on the underwater optical characteristics.

Zhou et al. (2021b) proposed an underwater background light estimation model based on flatness, hue, and brightness feature priors, which adaptively selects the most obvious features according to the input image to obtain more accurate background light and transmission map estimation. This method is inspired by the underwater scene prior.

Underwater image restoration methods that combine multiple prior advantages also continue to be developed (Zhao et al., 2015; Li et al., 2016b; Peng and Cosman, 2017). For instance, Zhang and Peng (2018) used two kinds of priors, MIP and UDCP, and saliency-guided multi-feature fusion to restore salient areas of underwater images. Zhou et al. (2021c) also developed a new method for underwater depth estimation that combines the advantages of the revised physical model of underwater imaging with priors and includes image segmentation and smoothing. In Table 1, a summary of the prior-based shallow-sea image restoration methods is provided.

10.3389/fmars.2023.1163831

TABLE 1 A summary of prior-based methods for shallow-sea image restoration.

Physical model	Characteristic	Year	Methods	Priori principle
Koschmieder model	Physically accurate models; the model is simple; and the model has a wide range of applications	2010	MIP (Carlevaris-Bianco et al., 2010)	The difference in attenuation between the RGB color channels of an underwater image
		2015	RCP (Galdran et al., 2015)	Red channel correction for underwater based on the DCP
		2016	UDCP (Drews et al., 2016)	Underwater DCP using G-B channels for transmission estimation
			BCP (Gao et al., 2016)	Bright channel prior based on the DCP
		2017	MAI (Wang et al., 2017)	The difference in attenuation between the RGB color channels of an underwater image
		2018	ULAP (Song et al., 2018)	The difference between blue-green light attenuation and the attenuation of red light underwater
		2020	Berman's (Berman et al., 2020)	Haze line prior
			Bekerman's (Bekerman et al., 2020)	Image color distribution
Image formation model	Approximate simulation of underwater imaging process; the model is relatively simple; and it is for underwater imaging only	2015	BP (Peng et al., 2015)	Scenes farther from the camera tend to be blurrier
		2016	Li's (Li et al., 2016b)	Blue-green channels dehazing and red channel correction based on DCP
			Li_HDP (Li et al., 2016a)	Histogram distribution prior
		2017	IBLA (Peng and Cosman, 2017)	Scenes farther from the camera tend to be blurrier; scene points that retain more red light in the red channel map are closer to the camera
		2018	GDCP (Peng et al., 2018)	Generalization of DCP
		2020	Hou's (Hou et al., 2020b)	Establish an underwater total variation model based on UDCP, in which UDCP is used to estimate the transmission map
		2021	Zhou's (Zhou et al., 2021a)	Secondary-guided transmission map optimization based on DCP
			Zhou and Wang's (Zhou et al., 2021b)	Based on flatness, hue, and lightness feature priors
Revised formation model	Optimization of underwater imaging models; more parameters involved in the model; and generally applicable to different underwater scenes	2019	Sea-thru (Akkaynak and Treibitz, 2019)	Estimates backscatter using the underwater derivation method of the DCP method, and uses the spatially varying illuminant to obtain the range-dependent attenuation coefficients

DCP, Dark Channel Prior; MIP, Maximum Intensity Prior; RCP, Red Channel Prior; UDCP, Underwater Dark Channel Prior; MAI, Maximum Attenuation Identification; ULAP, Underwater Light Attenuation Prior; BP, Blur Prior; Li_HDP, Li'method based on Histogram Distribution Prior; IBLA, underwater image restoration based on Image Blurriness and Light Absorption; GDCP, Generalization of the Dark Channel Prior.

Currently, there are two types of prior knowledge used in the field of shallow-sea image restoration: objective principles under the environmental conditions of shallow-sea imaging, and general statistical phenomena in shallow-sea images. However, the applicability of these priors in deep-sea conditions needs to be verified. In addition, the prior knowledge used in shallow-sea image restoration should be optimized for deep-sea imaging conditions. Another approach is to extract objective principles and common phenomena from the specific imaging environment and images of the deep sea and use these to inform the development of a joint prior method that combines the advantages of different prior methods to achieve the most accurate parameter estimation for deep-sea images.

2.3 Deep learning-based shallow -sea image restoration combined with physical models

Deep learning has gained popularity in underwater image restoration and has shown promising results in recent years. Anwar and Li (2020) have classified deep learning networks into five categories, namely, encoder–decoder networks, modular design networks, multibranch designs, depth-guided networks, and dual-generator generative adversarial networks (GANs), and provided detailed introductions to these networks. Although most deep learning networks prioritize directly generating visually appealing images, a few seek to recover more realistic images by leveraging the knowledge of the image degradation process, which may overcome the lack of ground-truth underwater images. Depth-guided networks, for instance, consider the relationship between depth and the estimation of transmission ratio and background light in the underwater imaging model, making it a valuable technique for shallow-sea image restoration. Eigen et al. (2014) applied neural networks to depth estimation, and researchers have subsequently combined depth prediction with the underwater IFM to achieve significant advancements in underwater image restoration (Hou et al., 2020a). In addition to these methods, there are other ways to restore images by integrating physical imaging models with deep learning networks. This section aims to investigate various approaches that combine deep learning techniques with physical imaging models, such as the Koschmieder model, the IFM, and the Akkaynak–Treibitz model, for the restoration of shallow-sea images.

2.3.1 Koschmieder model-based approach

Kar et al. (2021) proposed a multidomain image restoration method based on the Koschmieder model and zero-shot learning. In this approach, the network is trained using the degraded image and the degraded image generated by the Koschmieder model, and then the learned mapping is used to transfer between the undegraded image and the degraded image to obtain the restored image. The network estimates the unknown parameters of background light and transmission map in the Koschmieder model separately. The projection estimation network is implemented using multiscale feature extraction and feature selection of color channels, as illustrated in Figure 2C. When applied to the field of underwater image restoration, this method requires compensation for the red channel, which is performed as follows:

$$CF(x) = (\mu I^G - \mu I^R)\bar{I}^R(x)I^G(x)\,. \tag{25}$$

$$I^R = I^R + CF\,. \tag{26}$$

2.3.2 IFM-based approach

Lu et al. (2018) were among the first to use deep learning technology to tackle the problem of underwater image depth estimation, proposing a method based on optical cameras and deep convolutional neural networks for real-world underwater images. Ding et al. (Ding et al., 2017) used a convolutional neural network to estimate a depth map from a white balance-corrected image, which was then directly converted into a transmission map. Cao et al. (2018) proposed two network models, one for estimating the background light and the other for estimating depth. In the depth estimation network, two depth networks were overlaid to preserve both global features and local details, and the rough depth map was connected to the first layer of the refining network to preserve more detailed information. Pan et al. (2018) improved the contrast of underwater images using white balance and DehazeNet (Cai et al., 2016). They fused the two using a Laplacian pyramid and applied an edge enhancement algorithm to the fused image. DehazeNet estimated the transmission map and obtained the contrast-enhanced image based on the IFM. As shown in Figure 2A, Yan and Zhou (2020) creatively employed an imaging model as a constraint for network training, using the underwater image imaging model as a feedback controller for a GAN network to ensure that the estimation results were more realistic and consistent with the real image. In addition, a domain adaptation mechanism was introduced in the network to eliminate the domain difference between synthetic and real images.

2.3.3 Akkaynak–Treibitz model-based approach

The Akkaynak–Treibitz model integrates with deep learning methods in two ways. One is by generating synthetic image data for deep learning network training; the other is by guiding the deep learning network to estimate the physical model parameters to restore underwater images. As shown in Figure 2B, Liu et al. (2021) estimated the parameters of the revised underwater imaging physical model through an advanced global–local feature fusion network and restored the image under the guidance of the Akkaynak–Treibitz model. Desai et al. (2021) took advantage of the underwater parameter sensitivity of the Akkaynak–Treibitz model to propose reliable estimation methods for the relevant parameters. They used the reference image and its depth map as input to synthesize the underwater dataset and then used the synthetic dataset to train a conditional GAN network for underwater image restoration. Han et al. (2022) synthesized the reference images in the real underwater Heron Island coral reef dataset (HICRD) based on the new attenuation coefficient and background light estimation method. They proposed a network that uses a conditional GAN network and contrastive learning to improve the mutual information between the original image and the restored image. Lu et al. (2021) used an encoder network to extract features for the background light, backscattered transmission map, and direct transmission map based on the revised underwater IFM. Three independent decoder networks estimated these three components simultaneously. A scene attention module was designed in the network to refine the results. Finally, the estimated value was brought into the IFM to obtain the underwater restored image.

In the field of shallow-sea image restoration, combining physical models and deep learning methods has shown great potential and achieved remarkable results. Therefore, it is reasonable to explore the effectiveness of this approach in deep-sea image restoration as well. However, the shortage of deep-sea

10.3389/fmars.2023.1163831

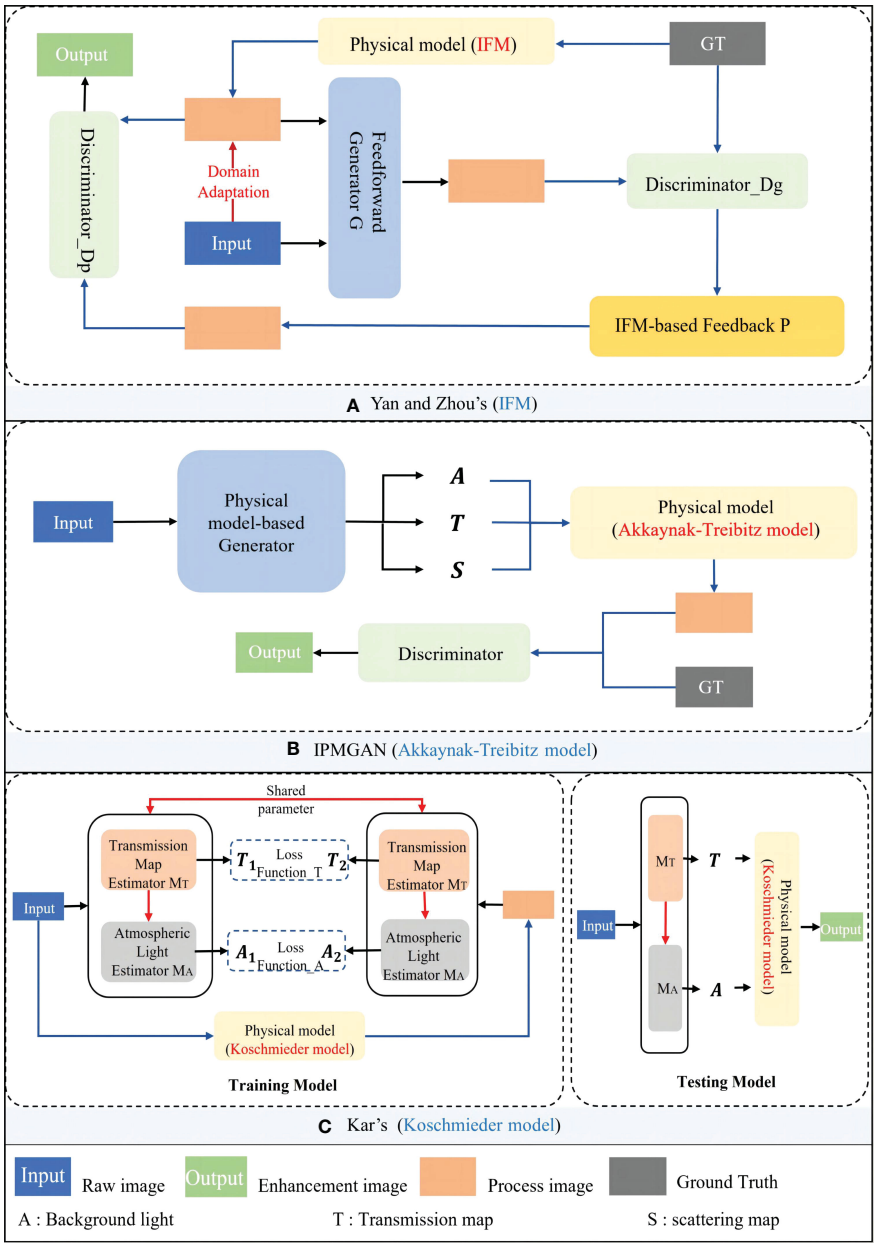


FIGURE 2
Shallow-sea image restoration methods based on the fusion of deep learning and physical models. **(A)** The deep learning network is based on the image formation model (IFM) (Yan and Zhou, 2020). **(B)** The deep learning network is based on the Akkaynak–Treibitz model (Liu et al., 2021). **(C)** The deep learning network is based on the Koschmieder model (Kar et al., 2021).

image data and the absence of reliable reference images have posed a challenge for traditional deep learning methods. Combining physical models with deep learning can reduce reliance on reference data to some extent. On the one hand, using a proper physical model to simulate the degradation process of deep-sea images we can construct deep-sea image datasets based on a large number of land images. On the other hand, physical models can serve as a constraint for the deep learning network to enable fast training with limited data. Alternatively, physical models can be integrated with deep-sea images to transform the image restoration process into a parameter estimation or linear solution problem, which can be solved more easily. Furthermore, exploring unsupervised deep learning methods, such as zero-shot learning in the field of deep-sea image restoration, is also promising. These methods could potentially improve the quality of deep-sea image restoration without relying on large numbers of labeled data.

3 Deep-sea image restoration methods

The exploration from shallow-sea to deep-sea environments presents significant challenges for imaging and observation owing to the absence of light in deeper waters. Artificial light sources must

be used for imaging but result in image degradation such as low light and non-uniform illumination. Current research on illumination problems in underwater imaging is limited. Figure 3 demonstrates a transition from shallow-sea to deep-sea image restoration, highlighting other relevant approaches to exposure and low-light enhancement to address the problems caused by artificial light sources.

In this research paper, the current methods for deep-sea image restoration are divided into two categories. The first category includes general methods that can be utilized to tackle specific problems in deep-sea images and the second category consists of methods designed specifically for deep-sea images.

3.1 General image restoration models applied to deep-sea images

A general image restoration method can be applied to the field of deep-sea image restoration by taking into account the light source problem during the imaging process or by generalizing the method used to solve degradation problems in shallow-sea images. This can help to mitigate the degradation caused by light source issues in deep-sea images to some extent.

The specific degradation issue in deep-sea images, as distinguished from that in shallow-sea images, lies in the use of artificial light sources. Therefore, studies that target lighting effects, such as vignetting, halo, and uneven illumination and exposure, can achieve good results in deep-sea image restoration. The general image restoration methods that have strong generalization capabilities can be used to address specific degradation issues present in deep-sea images by considering the light source problem in the imaging process. Researchers, such as Wen et al. (2013), have achieved good results in restoring deep-sea images using the underwater optical imaging model and the underwater dark channel estimation method. Lu et al. (2016) proposed a solution to the halo problem caused by artificial light sources, rather than the more general problems of deep sea image restoration such as color correction and brightness distribution, or outside the shallow-water image restoration process, to address degradation caused by light sources. Lu et al. (2015) considered a scenario where both ambient light and artificial light sources exist in enhancing shallow-water images and proposed an ambient light estimation algorithm based on color lines, a local adaptive filtering algorithm to enhance images, and correction of color bias based on spectral features, followed by illumination compensation for dark regions of the image to achieve global contrast enhancement of underwater images. Li's method (Li et al., 2020) took into account the improper installation of underwater light sources, lighting unevenness caused by environmental factors, and local overexposure, and proposed an adaptive filter correction to lighting and combined image segmentation and an image enhancement exponential metric to improve the adaptiveness of filter parameters.

In shallow-sea image restoration, the imaging models and prior knowledge used remain valid even when lighting conditions change. Such methods often have advantages in deep-sea image restoration. Wavelength compensation and image dehazing (WCID) proposed by Chiang and Chen (2012) determines the influence of artificial light sources on the imaging process by comparing the separated foreground and background intensity and compensates for the difference in light attenuation caused by artificial light sources. Color restoration is then done based on the residual energy ratio of different color channels and the scene depth combined with the corresponding attenuation. Li et al. (2018b) proposed a layer-wise transmission fusion method and a color-line background light estimation method to improve the illumination problem of single-input images by removing scattering. Deng's method (Deng et al., 2019) considered attenuation under different lighting conditions based on a new scene depth estimation. The background light is estimated based on the grayscale opening and scene depth estimation to avoid pixels in white objects and artificial lighting areas being mistakenly estimated as background light, and the defogged image can be obtained based on the estimated background light and transmission map. Although DCP and MIP are often ineffective owing to underwater illumination conditions, the IBLA method (Peng and Cosman, 2017) estimates the scene depth based on image blurriness and light absorption, which is more suitable for different lighting conditions. The GDCP method (Peng et al., 2018) estimates the background light based on the color change-dependent scene depth estimation and estimates the scene transmission from the difference between the observed intensity and the estimated intensity, which is suitable for image restoration under various special environment lighting and turbid media conditions. The RCP method (Galdran et al., 2015) focuses on the problem of light spots in images caused by artificial light sources rather than the low-illumination problem of deep-sea images.

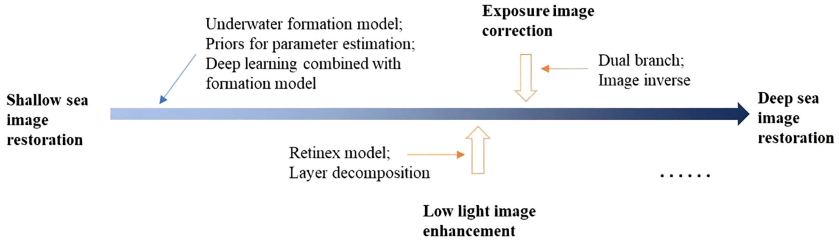


FIGURE 3
Relevant research directions to deep-sea image restoration.

However, despite their ability to generalize, the methods that are primarily designed for shallow-sea image restoration may not fully take into account the unique differences and lighting conditions present in deep-sea environments. Although these methods can still be applied to deep-sea image restoration, they may require further optimization to fully address the specific challenges of this environment.

3.2 Specially-designed models for deep-sea image restoration

Considering deep-sea image restoration based on the knowledge of shallow-sea imaging is a solid starting point, but the methods developed for shallow-sea image restoration may not fully address the unique and complex challenges of deep-sea imaging. Therefore, it is important to research new image restoration methods specifically tailored for deep-sea environments. For example, Wen et al. (2013) proposed a new underwater imaging model and transmittance estimation method for extreme underwater environments such as deep-sea and turbid waters. This model draws inspiration from the fog image imaging model (Narasimhan and Nayar, 2000; Narasimhan and Nayar, 2003; Fattal, 2008; Tan, 2008), but takes into account the additional effects of underwater absorption and scattering on imaging. The new imaging model is described as:

$$I^c(x) = J^c(x) \cdot t_\beta^c(x) + A^c \cdot t_\alpha(x), c \in \{R, G, B\}, \quad (27)$$

where t_β^c represents the proportion of scene radiation that reaches the camera directly, and t_α represents the sum of the effects of underwater absorption and scattering.

Liu et al. (2019) addressed the issue of regional color shift caused by the use of colored or uneven artificial light sources in deep-sea imaging by focusing on the illumination characteristics of deep-sea images and incorporating them into a simplified underwater imaging model. They proposed a frequency-domain-based hue estimation method to correct global color shift and combined it with scattering correction to improve pixel-level color shift and contrast. Subsequently, Liu et al. (2022) utilized the underwater simplified IFM and illumination parameters to simulate imaging principles under different lighting conditions and synthesized the first underwater uneven illumination dataset. They then used this dataset to train a proposed multiresolution image feature reconstruction convolutional neural network for deep-sea image enhancement.

The field of deep-sea image restoration is of great research value and significance as it allows for the full utilization of information in deep-sea images, which is beneficial for further deep-sea exploration tasks. However, in comparison to shallow-sea image restoration, research in this field is lacking. The complex deep-sea imaging environment and the unique characteristics of deep-sea images urgently require further study.

3.3 Analysis of deep-sea image restoration problems

Degradation problems in deep-sea image restoration can be divided into two categories: one is the color shift, low contrast, and blur caused by underwater characteristics; the other is low light, non-uniform illumination, and noise caused by artificial light sources. The restoration of underwater images has been analyzed in detail in Section 2. To address the degradation problem caused by artificial light-assisted imaging, Cao et al. (2020) proposed NUICNet, a fully connected network suitable for deep-sea images with an illumination correction loss. NUICNet views the underwater uneven illumination image as the product of the additive combination of the ideal image and the illumination layer and solves the problem with two modules: feature fusion and illumination layer separation. The feature extraction module combines the input image with parameters trained on the benchmark dataset (ImageNet; Deng et al., 2009) as hypercolumn features; the illumination layer separation module outputs the ideal image and illumination layer through an end-to-end network using the hypercolumn features as input.

Nevertheless, many deep learning-based image enhancement methods are supervised, requiring a large number of paired training data that consist of high-quality ground-truth images with diverse content. Currently, there is a dearth of deep-sea image data and no established deep-sea benchmark dataset with reference images. The problem of degradation induced by artificial light sources in deep-sea images could be tackled by drawing inspiration from research in related fields, such as exposure image correction and low-light image enhancement. Shallow-sea image enhancement methods based on deep learning would also be beneficial for restoring deep-sea images or serve as a valuable reference, given the success of these methods in eliminating various degradations of shallow-sea images.

3.3.1 Exposure image correction

At present, exposure errors remain a primary concern in camera imaging. These errors can be divided into two categories: overexposure, where certain areas in the image appear too bright and washed out, and underexposure, where certain areas appear too dark. Both types of exposure problems can occur in the same image, and they are common issues in deep-sea images. Therefore, research in the field of exposure can be leveraged to inspire the development of methods for deep-sea image restoration.

Wang et al. (2019a) proposed a network that employs local and global feature encoders to learn the mapping from underexposed images to illumination maps in order to achieve well-exposed images based on the Retinex model. Instead of directly learning the mapping from underexposure to the corrected image, this network learns the mapping from the illumination layer to the corrected image in order to preserve global features, such as color distribution, average brightness, and scene category, as well as local features, such as contrast, sharp details, intensity, shadow, and highlights. The network is constructed with dual modules for local and global feature extraction and smooths the output illumination map to obtain a high-precision illumination map. Figure 4A illustrates the network structure and implementation process of the method.

To address the issue of uneven exposure in deep-sea images, several methods have been proposed. Yu et al. (2018) presented a method that uses image segmentation to determine local exposure

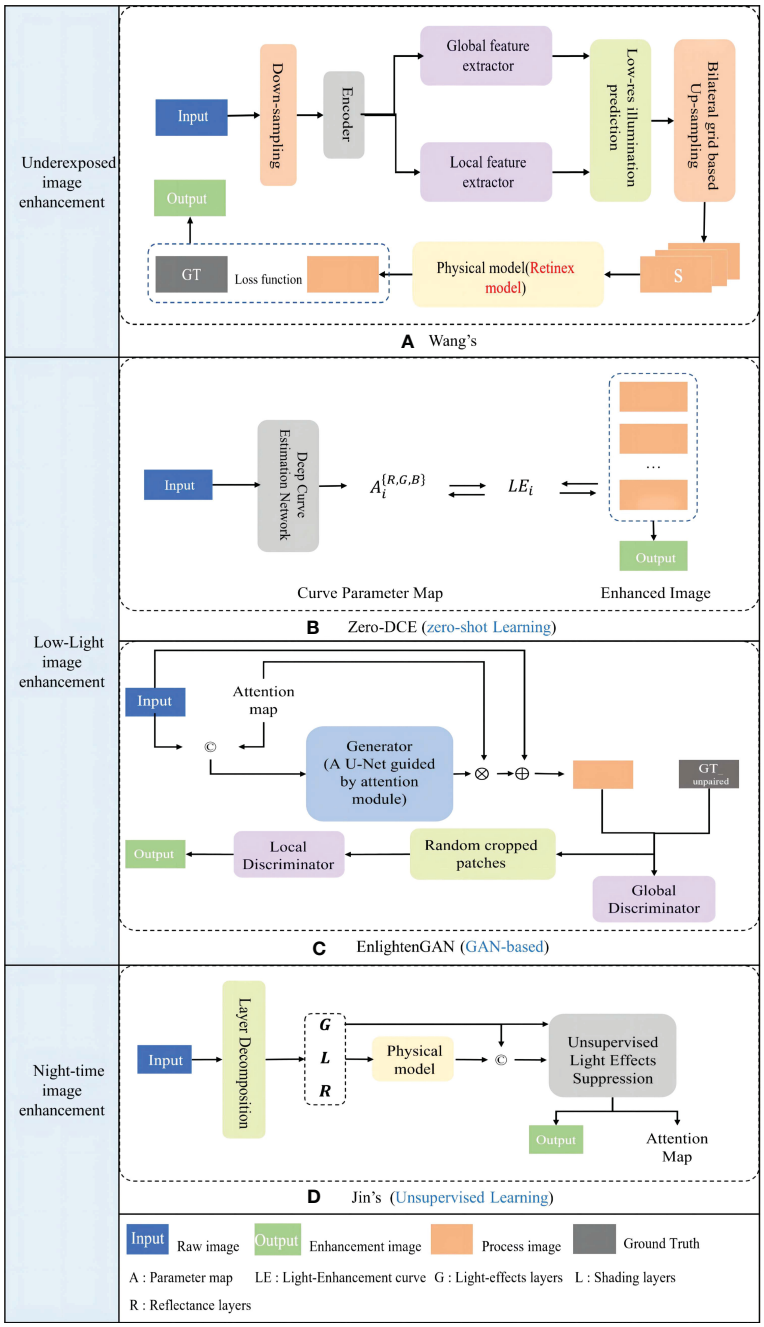


FIGURE 4
Representative deep learning network models. **(A)** The deep learning network of Wang's method (Wang et al., 2019a). **(B)** The deep learning network of Zero-DCE (Guo et al., 2020). **(C)** The deep learning network of EnLightenGAN (Jiang et al., 2021). **(D)** The deep learning network of Jin's method (Jin et al., 2022).

and apply it to the entire image. The resulting image is a fusion of images with different exposure levels to achieve a corrected image. Zhang et al., (2019a) considered both overexposure and underexposure in images and proposed a dual-illumination estimation network, which uses guidance to fuse corrected images with the input image to obtain a well-exposed image. Afifi et al. (2021) tackled the same problem by breaking down exposure correction into the two sub-problems of detail enhancement and color enhancement and proposed a coarse-to-fine deep network, which was trained on a constructed paired dataset and successfully solved the sub-problems.

The study of exposure correction in images, particularly those with multiple exposures, holds valuable insights for addressing the degradation caused by artificial light sources in deep-sea images. As data collection in the field of exposure research is relatively straightforward, there is an abundance of reliable paired training datasets. However, the differences between these datasets and those of the deep-sea environment make it necessary to adapt exposure

correction methods to the unique characteristics of the deep sea and reduce their dependence on training data.

3.3.2 Low-light image enhancement

Research on low-light image enhancement can provide valuable insights for deep-sea image restoration, as the deep sea is also considered a low-light environment. In low-light conditions, images captured by cameras often have issues such as loss of detail, reduced contrast, poor visibility, and noise.

For low-light image enhancement, Lore et al. (2017) proposed a method that utilizes stacked sparse denoising autoencoders to learn latent features in low-light images and to obtain an output image with minimal noise and optimized contrast. Guo et al. (2017) proposed a new low-light image restoration method based on the Retinex model, which initializes an illumination map by selecting the maximum value in the pixel channel and refining it with the structure prior, ultimately producing an illumination-corrected image based on the refined illumination map. Li et al. (2018a) proposed a four-layer fully convolutional neural network, in which the first two layers focus on high-light areas, the third layer focuses on low-light areas, and the last layer is used to reconstruct the illumination map. The gamma-corrected illumination map and the original image are combined using the Retinex model to produce a well-exposed image. Fu et al. (2016) proposed a weighted variational model for estimating reflection and illumination maps from input images. This model can suppress noise and estimate more detailed reflection maps than the traditional Retinex model.

Guo et al. (2020) took into consideration low light and uneven illumination caused by different illumination conditions and proposed the zero-deep curve estimation (Zero-DCE) network, as shown in Figure 4B. This network does not rely on paired data and transforms image enhancement into a curve estimation problem, iteratively finding the best-fitting curve pair and adjusting the original image pixel by pixel to achieve image illumination correction. A lightweight network of Zero-DCE is named Zero-DCE++ (Li et al., 2021b).

Jiang et al. (2021) introduced unpaired training into the field of low-light image enhancement for the first time. The network adopts a PatchGAN-based global–local double discriminator structure to solve the problem of overexposure and underexposure simultaneously. In addition, the network incorporates a self-attention mechanism known as U-Net (Ronneberger et al., 2015) to improve the visual effect of brightness correction in regions of varying illumination. The network details are shown in Figure 4C.

For night image enhancement, Jin et al. (2022) performed layer decomposition using three independent unsupervised networks. They used the light effect layer to guide the light suppression module, reducing the influence of light effects and enhancing the dark areas. The detailed network structure is shown in Figure 4D.

In addition, Zhang et al., (2019b) proposed a KinD network to decouple the original image space into illumination components and reflection components and take images with different exposure levels as inputs for their proposed model. The illumination adjustment module in the model can adjust the illumination level according to specific needs. Later, Zhang et al. (2021) further optimized the low-light image enhancement effect by introducing a multiscale brightness attention module and abandoning the U-Net network model structure of the reflectance restoration module in the KinD network, resulting in the KinD++ network.

Research on low-light image enhancement has shown promising results in brightness correction and noise suppression through the use of the Retinex layer decomposition method. However, to apply this method to deep-sea image restoration, it is necessary to take into account the unique characteristics of the deep-sea environment and reduce reliance on training data.

3.4 Deep learning-based methods design

Deep learning-based methods are becoming mainstream in shallow-sea image quality improvement research, but their reliance on training data needs careful consideration when they are designed for deep-sea images. The following potential solutions are considered.

First, some well-trained, supervised deep learning models have demonstrated good generalization and robustness to effectively solve challenging underwater image quality enhancement problems, such as Ucolor (Li et al., 2021a) and U-shape (Peng et al., 2023). Ucolor is a multicolor space deep network model that uses the transmission map estimation output by GDCP to guide network model training, offering advantages that combine traditional and deep learning methods for richer image feature extraction. U-shape is based on the transformer network and is strengthened by a self-attention mechanism and a multicolor space loss function designed according to the human vision principle. This kind of supervised model could serve as a fundamental model for deep-sea image restoration.

Second, semisupervised and unsupervised learning methods are less dependent on data and are better suited to the current situation in which reliable reference data cannot be obtained. For instance, Semi-UIR (Huang et al., 2023), a semisupervised underwater image restoration method based on the mean teacher approach, incorporates unpaired data into the model training process and introduces pseudo-reference images and contrastive regularization to counteract network overfitting. The unsupervised method UDnet (Saleh et al., 2022) requires only degraded images, with a reference image generated by a conditional variational autoencoder with probabilistic adaptive instance normalization and a multicolor space stretching module.

Other semi-supervised and unsupervised learning methods based on GANs or zero-shot learning can help deep-sea image quality enhancement network design. The combination of imaging models and GANs, as shown in Figure 2, has produced promising results in enhancing underwater image quality. However, when integrating the Retinex model into deep learning methods for low-illumination image enhancement, several limitations must be considered. The ideal assumption used in Retinex-based low-light image enhancement methods, that reflectivity is the final enhancement result, may still impact the final outcome. In addition, despite the use of the Retinex theory, deep networks may still be at risk of overfitting (Li et al., 2021b). Similar considerations should be taken into account for deep learning-

based restoration methods that integrate physical models, including the fusion strategy, the assumptions of the physical model, and the need to prevent overfitting. Refer to Table 2 for a detailed examination of some representative network models. It is worth considering whether or not supervised shallow-sea image enhancement networks, such as Ucolor and U-shape, known for their robustness, can achieve ideal results in deep-sea image enhancement. The impact of deep networks on different levels of data dependency will also be analyzed in the next section.

4 Experiment analysis

In order to extend the application of underwater image restoration to the deep sea, this section uses both the shallow-sea image dataset and the deep-sea underwater image dataset to conduct subjective and objective evaluations. The results of the experiments will be analyzed and summarized to highlight the strengths and weaknesses of each prior-based method in deep-sea image restoration. In addition, visual examples of some classic and advanced deep-sea image enhancement, low-light image enhancement, exposure image correction, and shallow-sea image enhancement methods will be applied to the OceanDark dataset to further investigate reliable techniques for deep-sea image restoration.

4.1 Experiment setup

In order to reflect the advantages and characteristics of each method, all the experiment methods adopted in this research paper are based on the open-source code from the original studies and are tested using the Linux+ NVIDIA RTX 3090 GPU experimental environment.

The experiment datasets used are the real shallow-sea underwater image enhancement benchmark dataset (UIEB) (Li et al., 2019) and the deep-sea underwater image dataset OceanDark (Porto Marques et al., 2019). Detailed information on the datasets can be found in Table 3. In the comparison experiment, the underwater image colorfulness measure (UIQM) (Panetta et al., 2016), underwater color image quality evaluation (UCIQE) (Yang and Sowmya, 2015), and the blind/reference less image spatial quality evaluator (BRISQUE) (Mittal et al., 2012) were selected as three no-reference underwater image quality evaluation indicators to quantitatively evaluate the enhancement effects of different methods on deep-sea degraded images.

The experimental methods used in this study include a selection of prior-based shallow-sea image restoration methods, including DCP (Kaiming He et al., 2011), MIP (Carlevaris-Bianco et al., 2010), IBLA (Peng and Cosman, 2017), ULAP (Song et al., 2018), UDCP (Drews et al., 2016), GDCP (Peng et al., 2018), and (Li et al., 2016a). The aim is to assess the applicability of these methods in the deep-

TABLE 2 A summary of representative deep learning-based methods incorporated with physical models.

Method	Learning type*	Model-based			Layer decomposition	Loss function
		Retinex	UIFM	Others		
Wang's (Wang et al., 2019a)	S	√				Reconstruction loss; smoothness loss; color loss
Zero-DCE (Guo et al., 2020)	Z					Spatial consistency loss; exposure control loss; color constancy loss; illumination loss
EnlightenGAN (Jiang et al., 2021)	U					Global self-feature preserving loss; local self-feature preserving loss; global generator loss; local generator loss
Jin's (Jin et al., 2022)	U			√	√	Initial loss; gradient exclusion loss; color constancy loss; reconstruction loss
Yan and Zhou's (Yan and Zhou, 2020)	S		√			Adversarial loss; cycle loss; pixel loss; coral loss
IPMGAN (Liu et al., 2021)	S		√			GAN loss; L_1 distance loss; SSIM loss
Kar's (Kar et al., 2021)	Z		√			Transmission relation loss; light similarity loss; saturated pixel loss; gray-world assumption loss; total variation loss

*S, supervised learning; U, unsupervised learning; UIFM, underwater imaging formation model; Z, zero-shot learning.
The "√" indicates the model type and layer decomposition applied by this method.

TABLE 3 Datasets information.

Dataset	Year	Image category	Characteristics
UIEB	2019	Shallow-sea images	Real-world underwater dataset, containing 890 images with reference images and 183 images without reference images. The imaging light sources consist of full natural light, full artificial light source, and a combination of half natural and half artificial light.
OceanDark	2019	Deep-sea images	Real-world underwater dataset, 183 deep-sea images without reference images; artificial light source.

sea environment and analyze their advantages and limitations. In addition, the experiments were also conducted with a variety of low-light image enhancement methods, such as low-light image enhancement (LIME) (Guo et al., 2017), joint enhancement and denoising (JED) (Ren et al., 2018), LightenNet (Li et al., 2018a), KinD (Zhang et al., 2019b), Wang's method (Wang et al., 2019c), Zero-DCE (Guo et al., 2020), Zero-DCE++ (Li et al., 2021c), the robust Retinex decomposition network (RRDNet) (Zhu et al., 2020) and KinD++ (Zhang et al., 2021), nighttime image enhancement methods, such as Jin's method (Jin et al., 2022); and underwater low-light and poor visibility methods, such as L^2uwe (Marques and Branzan Albu, 2020), MLLE (Zhang et al., 2022), and hyper-laplacian reflectance priors (HLRP) (Zhuang et al., 2022). A set of deep learning-based methods that have shown excellent performance in shallow-sea image enhancement were also employed. They are divided into the supervised methods Ucolor (Li et al., 2021a) and U-shape (Peng et al., 2023), the semisupervised method Semi-UIR (Huang et al., 2023), and the unsupervised methods UDnet (Saleh et al., 2022) and Kar's method (Kar et al., 2021). These methods aim to assist the design of new deep-sea image degradation problems. The significance of the image enhancement scheme was analyzed, with advantages and limitations in enhancing deep-sea images discussed. In total, 25 methods were compared and analyzed to determine their effectiveness in enhancing deep-sea images by addressing issues related to underwater light absorption and scattering, low light caused by artificial light sources, and uneven illumination.

4.2 Experiment results

4.2.1 Results of prior-based underwater image restoration

Deep-sea images and shallow-sea images share a common problem: color shift and blur that are caused by underwater light absorption and reflection. Thus, a natural consideration is whether or not we can apply shallow-sea image restoration methods to deep-sea images to deal with the color shift and blur problem. However, there are objective differences between deep and shallow sea environments. To verify this, experiments were conducted with prior-based shallow-sea image restoration methods using both UIEB and OceanDark datasets.

Based on the objective evaluation results in Tables 4, 5, the shallow-sea image restoration methods showed improvements in both UIQM (Panetta et al., 2016) and UCIQE (Yang and Sowmya, 2015) metrics for the UIEB and the OceanDark deep-sea dataset compared with the scores of "raw" images. UIQM is a combination of colorfulness, sharpness, and contrast, and UCIQE is also a linear combination of image characteristics such as chroma, saturation, and contrast. UIQM and UCIQE may assign high ratings to images with severely degraded naturalness (e.g., the ULAP-enhanced images score higher). In contrast, BRISQUE based on natural scene statistics is more suitable to evaluate the quality of enhanced deep-sea images, and the lower the score, the better. Comparing the metric values in Table 5 with those in Table 4, it can be concluded that these shallow-sea image restoration methods perform worse on OceanDark than on UIEB. This proves that deep-sea images suffer from more severe degradation than shallow-sea images.

To analyze the challenges encountered when applying shallow-sea image restoration methods to deep-sea image restoration, the visual effects of the different methods are shown in Figure 5. The DCP method produces deep-sea images with a more severe blue-green tint than other methods and fails to restore images with white targets. The deep-sea images restored using the MIP method have more bright and dark areas. Both IBLA and ULAP can effectively enhance contrast, but they each introduce false colors and are more sensitive to degradation caused by artificial light sources, resulting in over-dark and bright areas with a significant loss of image details. Although both GDCP and UDCP are based on the underwater DCP, they produce conflicting results in the restoration of deep-sea images. UDCP causes an overall decrease in image brightness, whereas GDCP overexposes deep-sea images. Li's method, based on minimum information loss and histogram prior, has achieved the best visual effect in terms of color correction and texture detail preservation, but it makes bright areas too bright and introduces obvious blocky artifacts.

TABLE 4 Objective evaluations of classic shallow-sea image restoration methods on UIEB dataset.

	Raw	DCP	MIP	IBLA	ULAP	UDCP	GDCP	Li's
BRISQUE	25.36	25.61	27.36	24.78	25.40	**24.64**	25.06	32.90
UIQM	1.854	3.208	3.053	3.239	3.630	3.623	2.147	**4.418**
UCIQE	0.5006	0.5279	0.5454	0.5638	0.5767	0.5720	0.6015	**0.6742**

The best-performing results are indicated in bold font.

TABLE 5 Objective evaluations of classic shallow-sea image restoration methods on OceanDark dataset.

	Raw	DCP	MIP	IBLA	ULAP	UDCP	GDCP	Li's
BRISQUE	32.59	31.29	32.32	35.88	31.64	32.59	29.74	**28.42**
UIQM	1.652	2.077	2.269	1.995	2.462	2.151	1.686	**2.587**
UCIQE	0.5448	0.5653	0.6291	0.5813	**0.6328**	0.5626	0.5639	0.5813

The best-performing results are indicated in bold font.

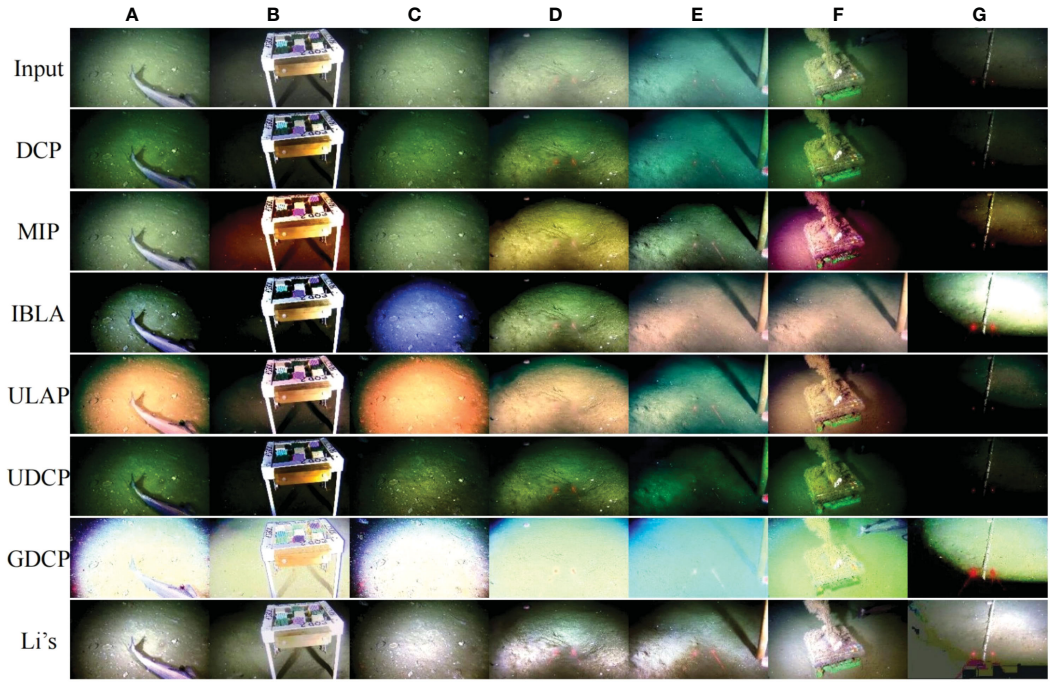


FIGURE 5
(A–G) represent column numbers. The visual effect of different prior-based methods on the OceanDark dataset.

Following the above analysis, it is clear, both subjectively and objectively, that the priori-based methods designed for shallow-sea images have a certain level of effectiveness; however, they cannot be directly applied to deep-sea image restoration.

4.2.2 Results of the methods for complex environmental problems

A further problem of deep-sea images is low light and uneven illumination caused by artificial light sources. As discussed in Section 3.3, the methods that are purposely designed for image exposure correction and low-light image enhancement might be useful in improving the quality of deep-sea images. To verify this idea, we performed a group of experiments and demonstrated their results using various deep-sea images.

Considering that there are few methods specifically designed for deep-sea images, we selected and compared 14 methods that might be effective in addressing some problems caused by the deep-sea environment. Listed in Table 6, these methods were originally developed for various fields, such as underwater images (e.g., L^2uwe, MLLE, HLRP), low-light images [e.g., LIME, JED, LightenNet, KinD, KinD++, RRDNet, Wang's method (Wang et al., 2019c), Kar's method (Kar et al., 2021)], night images [e.g., Jin's method (Jin et al., 2022)], and over/underexposed images (e.g., Zero-DCE, Zero-DCE++).

The advantages and limitations of these methods for deep-sea image restoration are analyzed in Table 7 and Figure 6, providing a reference for research in deep-sea image restoration. It is important to note that the comparisons of these methods are based on their effectiveness in deep-sea image restoration and may not reflect their overall performance in their respective fields of origin.

With regard to color correction, Figure 6A demonstrates that the methods specifically designed for underwater image enhancement, such as MLLE and HLRP, perform better than those from other fields. Meanwhile, the methods from the low-light and exposure correction field, such as RRDNet, often lack a color correction process and may even introduce new color casts when addressing degradation caused by artificial light sources. When it comes to illumination correction, low-light image enhancement methods, such as LIME, Zero-DCE, Zero-DCE++, KinD, and KinD++, achieve good results, but have limitations in preserving details, correcting color cast, and reducing artifacts in deep-sea images. This highlights the need for further research that incorporates deep-sea characteristics to find solutions.

In terms of handling sudden changes in pixel values, such as the red beam in Figure 7B, methods such as HLRP and L^2uwe are more effective. However, HLRP leads to overexposure in the center of the light source instead of darkening the light source area, and L^2uwe results in a contrast that is too high in the processed deep-sea image. As shown in Figures 6C, D, in extreme examples of deep-sea images neither low-light enhancement nor underwater image enhancement methods have achieved satisfactory results. The severe lack of illumination and the overexposure of foreground targets in deep-sea images requires further research.

According to the objective evaluation results shown in Table 7, underwater image enhancement methods show increases in both UIQM and UCIQE, whereas the low-light image enhancement and nighttime image enhancement methods have led to decreases in these two metrics. This is because UIQM and UCIQE place a greater weight on color measurement, which is not required for low-light image enhancement and nighttime image enhancement as they do

TABLE 6 The methods for complex environmental problems.

Method	Year	Application scenes	Imaging model	Deep learning based
LIME	2017	Low light	Retinex	
JED	2018	Low light	Retinex decomposition	
LightenNet	2018	Low light	Retinex	√
KinD	2019	low light	Retinex decomposition	√
Wang's	2019	Low light	Absorption light scattering model	
Zero-DCE	2020	Low light, exposure	\	√
RRDNet	2020	Low light	Retinex decomposition	√
L^2 uwe	2022	Underwater low light	\	
KinD++	2021	Low light	Retinex decomposition	√
Kar's	2021	Low light, underwater, haze	Koschmieder	√
ZeroDCE++	2021	Low light, exposure	\	√
MLLE	2022	Underwater low visibility	\	
HLRP	2022	Underwater low visibility	Retinex variational correction	
Jin's	2022	Night image	Retinex layer decomposition	√

TABLE 7 Objective evaluations of image enhancement methods in various fields on the OceanDark dataset.

	BRISQUE	UIQM	UCIQE
Raw	32.59	1.652	0.5448
L^2uwe	31.36	3.390	0.5661
MLLE	25.77	3.166	0.5733
HLRP	41.24	1.967	0.5837
LIME	24.36	1.395	0.5265
Wang's	30.80	1.051	0.5295
JED	35.50	0.790	0.5092
LightenNet	28.40	1.187	0.5191
RRDNet	29.06	1.532	0.5517
Zero-DCE	34.40	1.182	0.4624
Zero-DCE++	32.34	2.663	0.4816
KinD	32.68	1.428	0.4968
KinD++	33.31	1.283	0.5001
Jin's	54.70	1.216	0.5342
Kar's	34.19	**4.399**	**0.6259**
Ucolor	25.82	4.059	0.5231
U-shape	**6.988**	3.749	0.5407
UDnet	30.98	3.523	0.5348
Semi-UIR	20.86	4.050	0.5879

The best-performing results are indicated in bold font.

not aim to correct color deviation caused by underwater light absorption. When compared with "raw" images, most image enhancement methods across various fields did not show significant improvements on the BRISQUE index. This indicates that, no matter the method of shallow-sea image restoration—low-light image enhancement, night image enhancement, or exposure image correction—they all have limitations in deep-sea image enhancement. On the BRISQUE index, however, the MLLE

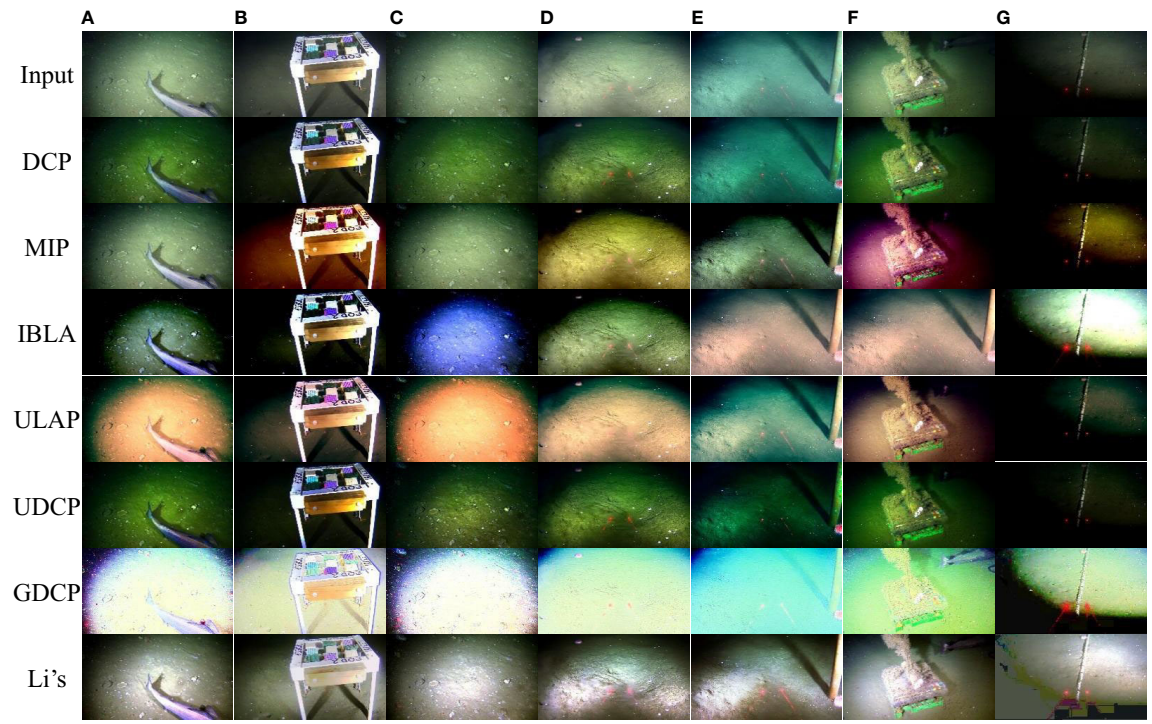


FIGURE 6
(A–G) represent column numbers. The visual effect of different deep learning-based methods on the OceanDark dataset.

method for underwater improvement showed promising results. This is because the technique produces an improved image that is more realistic in terms of both color and content.

4.2.3 Results of deep learning-based underwater image enhancement

In this section, we aim to explore the potential effectiveness of the robust shallow-sea image enhancement method in addressing the degradation of deep-sea images and the influence of various data dependencies on deep learning image enhancement. The OceanDark dataset is used to experiment with supervised deep learning methods, including Ucolor and U-shape methods, the semisupervised learning method Semi-UIR, and the unsupervised deep learning method UDnet and Kar's method. The objective evaluation results with UIQM, UCIQE, and BRISQUE are listed in Table 7.

The visual results, as illustrated in Figure 6, indicate that deep learning-based shallow-sea image enhancement methods, with the exception of Kar's method, exhibit superior visual outcomes in deep-sea image color correction and the retention of underwater environmental details. Notably, the supervised model Ucolor demonstrates distinct advantages in color correction, also evidenced by its UIQM score in Table 7. Furthermore, the U-shape method produces remarkably robust results using the BRISQUE indicator. Compared with the unsupervised methods, the supervised deep learning approach for enhancing shallow-sea images has produced more competitive visual results, but problems remain with low light and uneven illumination created by artificial light sources, and lower lighting may decrease color correction accuracy. Kar's method performed well using the UIQM and UICQE indicators. This is because the technique accounts for how underwater images degrade, producing a restored image with more details preserved.

In terms of implementation efficiency, it is important to note that the running time of the various deep learning methods is not always long. As shown in Table 8, methods such as KinD, Zero-DCE, Zero-DCE++, and Jin's method have relatively shorter running times, making them more suitable for real-time applications. Shallow-sea image restoration methods that utilize deep learning techniques. These methods do not provide a processing time advantage due to the inherent complexity involved in transforming from image to image. However, KinD and KinD++ address the complexity of the image problem by dividing it into two simpler sub-problems. Similarly, Zero-DCE and Zero-DCE++ tackle the problem by estimating curves from the image. As a result, these methods effectively reduce the time cost.

5 Conclusion

This study provides an overview of the current state of research on underwater image restoration, focusing on research gaps between shallow-sea image restoration and deep-sea image restoration. It identifies the causes of degradation in underwater images, classifies and examines existing restoration methods, and evaluates their strengths and weaknesses. By comparing the results of classic shallow-sea image restoration techniques applied to both

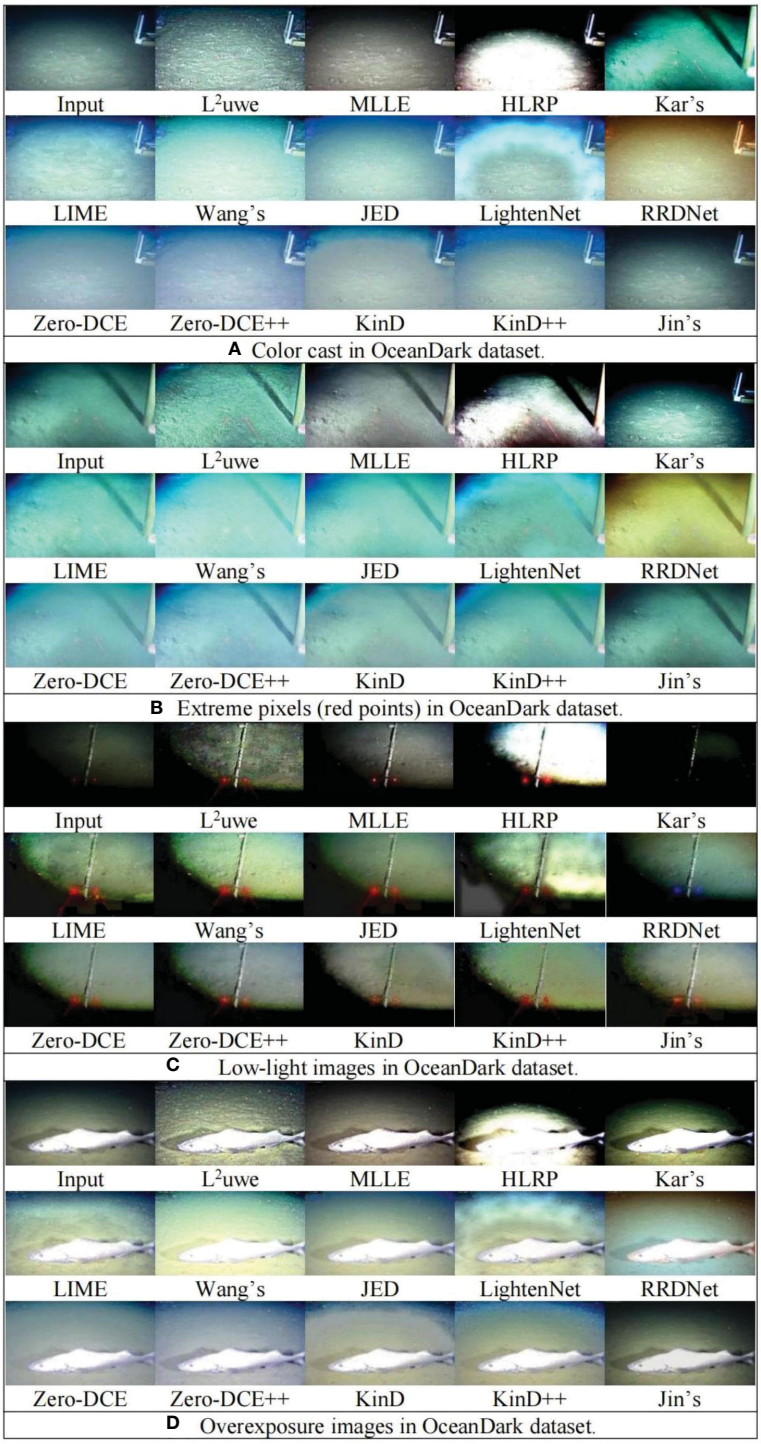


FIGURE 7
(A–D) different degradation types in the deep-sea environment. Comparison results of experiments in the OceanDark dataset.

TABLE 8 Runtime of deep learning-based methods.

Method	KinD	Zero-DCE	RRDNet	KinD++	Zero-DCE++	Jin's	Kar's*	Ucolor	U-shape	UDnet	Semi-UIR
Elapsed time/ms	6.36	1.09	85.64	37.90	4.17	4.86	163299.01	34779.23	419.7	704.04	238.22

*The run time of Kar's method is based on 1000 iterations in order to ensure the quality of image restoration.

shallow-sea and deep-sea datasets, and the results of the latest methods for underwater image enhancement, exposure correction, and low-light enhancement using the deep-sea dataset, this study concludes that existing methods in the related fields are insufficient to address the deep-sea image degradation problem. Following an analysis of the similarities and differences between shallow-sea and deep-sea image degradation and the experimental results, we suggest the following research directions to guide future research on underwater image restoration.

(1) Combining an underwater formation physical model with deep learning techniques has great potential in the domain of deep-sea image restoration. The combination aims to retain two advantages: producing more realistic and naturally restored images and improving the robustness and adaptability of the methods. However, two major challenges must be addressed. (i) The physical model for the deep-sea environment is not well studied. In particular, the existing underwater imaging model cannot accurately express the deep-sea lighting conditions, resulting in a significant reduction of visual areas; and (ii) different underwater scenarios and types of degraded images require high adaptability of the models to meet the demands of practical applications.

(2) Given the current scarcity of deep-sea image datasets, future research in deep-sea image restoration should explore the potential application of unsupervised learning and zero-shot learning. However, the relationship between these learning strategies and deep-sea image restoration is not well understood, and further research is needed to evaluate the effectiveness of unsupervised learning and zero-shot learning in deep-sea image restoration.

(3) To be applicable in real-world environments, methods for deep-sea image restoration should be optimized for real-time performance. However, most existing methods for underwater image restoration require significant processing time. Inspired by the application fields and requirements of low-light image enhancement, improving the real-time performance of deep learning-based underwater image restoration methods can simplify complex image processing procedures, such as estimating curve parameters (Guo et al., 2020) or splitting into multiple sub-problems that are easier to handle (Zhang et al., 2019b).

(4) The establishment of an underwater image quality evaluation system is important. There is a lack of publicly available datasets that can support training deep learning-based deep-sea image restoration methods, and the evaluation systems are not optimal. This hinders the progression of research in this field and the selection of appropriate methods for practical applications.

(5) Aside from what has been mentioned in this research paper, there are more issues related to deep-sea images that are rarely studied. When collecting deep-sea images, the landing of equipment on the seabed can cause an influx of seabed dust, microorganisms, and suspended particles, which often lasts for a long time (even hours) and leads to red-yellowish and blurry images. Developing solutions to address this problem is crucial for practical applications. Although much of the research in underwater image restoration focuses on single images, the practical application of underwater images also extends to videos. However, there is a lack of attention given to the restoration of underwater videos. This research gap needs to be addressed, as underwater videos play a significant role in practical applications. Urgent attention is needed to address processing efficiency and frame-to-frame consistency in underwater video restoration.

Data availability statement

Publicly available datasets were analyzed in this study. These data can be found here:.

OceanDark dataset: https://sites.google.com/view/oceandark/home

UIEB dataset:

Raw: https://drive.google.com/file/d/12W_kkblc2Vryb9zHQ6BfGQ_NKUfXYk13/view?pli=1

References: https://drive.google.com/file/d/1cA-8CzajnVEL4feBRKdBxjEe6hwql6Z7/view

Challenging: https://drive.google.com/file/d/1Ew_r83nXzVk0hlkfuomWqsAIxuq6kaN4/view.

Author contributions

Conceptualization—WS, YL, and HX. Methodology—WS, YL, DH, ZS, and BZ. Original draft—YL, and HX. Experiments—YL and WS. Review, editing, and supervision—WS, DH, BZ, and HX. Investigation and visualization—YL and ZS. Funding acquisition—WS, DH, BZ, and HX. All authors contributed to the article and approved the submitted version.

Funding

This work was funded by the National Natural Science Foundation of China (61972240), and the program for the capacity development of Shanghai local universities by the Shanghai Science and Technology Commission (20050501900).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Afifi, M., Derpanis, K. G., Ommer, B., and Brown, M. S. (2021). "Learning multi-scale photo exposure correction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN, USA: IEEE). 9157–9167. doi: 10.1109/CVPR46437.2021.00904

Akkaynak, D., and Treibitz, T. (2018)A revised underwater image formation model (Accessed 15 Aug. 2021).

Akkaynak, D., and Treibitz, T. (2019)Sea-Thru: a method for removing water from underwater images (IEEE) (Accessed 22 Aug. 2021).

Akkaynak, D., Treibitz, T., Shlesinger, T., Loya, Y., Tamir, R., and Iluz, D. (2017) What is the space of attenuation coefficients in underwater computer vision? (IEEE) (Accessed 22 Aug. 2021).

Ancuti, C. O., Ancuti, C., De Vleeschouwer, C., and Bekaert, P. (2018). Color balance and fusion for underwater image enhancement. *IEEE Trans. Image Process.* 271, 379–393. doi: 10.1109/TIP.2017.2759252

Anwar, S., and Li, C. (2020). Diving deeper into underwater image enhancement: a survey. *Signal Processing: Image Communication* 89, 115978. doi: 10.1016/j.image.2020.115978

Bekerman, Y., Avidan, S., and Treibitz, T. (2020). "Unveiling optical properties in underwater images," in *2020 IEEE International Conference on Computational Photography (ICCP).* (St. Louis, MO, USA). 1–12.

Berman, D., Levy, D., Avidan, S., and Treibitz, T. (2020). Underwater single image color restoration using haze-lines and a new quantitative dataset. *IEEE Trans. Pattern Anal. Mach. Intell.* 438, 2822–2837. doi: 10.1109/TPAMI.2020.2977624

Berman, D., Treibitz, T., and Avidan, S. (2016). Non-local image dehazing (IEEE) (Accessed 2 Nov. 2022).

Cai, B., Xu, X., Jia, K., Qing, C., and Tao, D. (2016). DehazeNet: an end-to-End system for single image haze removal. *IEEE Trans. Image Process.* 2511, pp.5187–5198. doi: 10.1109/TIP.2016.2598681

Cao, K., Peng, Y.-T., and Cosman, P. C. (2018)Underwater image restoration using deep networks to estimate background light and scene depth (IEEE) (Accessed 19 Jun. 2022).

Cao, X., Rong, S., Liu, Y., Li, T., Wang, Q., and He, B. (2020). NUICNet: non-uniform illumination correction for underwater image using fully convolutional network. *IEEE Access* 8, 109989–110002. doi: 10.1109/ACCESS.2020.3002593

Carlevaris-Bianco, N., Mohan, A., and Eustice, R. M. (2010)Initial results in underwater single image dehazing (IEEE) (Accessed 7 Mar. 2022).

Chiang, J. Y., and Chen, Y.-C. (2012). Underwater image enhancement by wavelength compensation and dehazing. *IEEE Trans. Image Process.* 214, 1756–1769. doi: 10.1109/TIP.2011.2179666

Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). ImageNet: a large-scale hierarchical image database. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops). 248–255. doi: 10.1109/CVPR.2009.5206848

Deng, X., Wang, H., and Liu, X. (2019). Underwater image enhancement based on removing light source color and dehazing. *IEEE Access* 7, 114297–114309. doi: 10.1109/ACCESS.2019.2936029

Desai, C., Tabib, R. A., Reddy, S. S., Patil, U., and Mudenagudi, U. (2021)RUIG: realistic underwater image generation towards restoration (IEEE) (Accessed 31 Oct. 2022).

Ding, X., Wang, Y., Zhang, J., and Fu, X. (2017)Underwater image dehaze using scene depth estimation with adaptive color correction (IEEE) (Accessed 19 Jun. 2022).

Drews, P. L. J., Nascimento, E. R., Botelho, S. S. C., and Montenegro Campos, M. F. (2016). Underwater depth estimation and image restoration based on single images. *IEEE Comput. Graphics Appl.* 362, 24–35. doi: 10.1109/MCG.2016.26

Eigen, D., Puhrsch, C., and Fergus, R. (2014)Depth map prediction from a single image using a multi-scale deep network. In: *Advances in neural information processing systems* (Curran Associates, Inc). Available at: https://proceedings.neurips.cc/paper/2014/hash/7bccfde7714a1ebadf06c5f4cea752c1-Abstract.html (Accessed 19 Jun. 2022).

Fattal, R. (2008). Single image dehazing. *ACM Trans. Graphics* 273, 1–9. doi: 10.1145/1360612.1360671

Fattal, R. (2014). Dehazing using color-lines. *ACM Trans. Graphics* 341, 1–14. doi: 10.1145/2651362

Fayaz, S., Parah, S. A., Qureshi, G. J., and Kumar, V. (2021). Underwater image restoration: a state-of-the-art review. *IET Image Process.* 152, 269–285. doi: 10.1049/ipr2.12041

Fu, X., Zeng, D., Huang, Y., Zhang, X.-P., and Ding, X. (2016). A weighted variational model for simultaneous reflectance and illumination estimation (IEEE) (Accessed 27 Sep. 2022).

Galdran, A., Pardo, D., Picón, A., and Alvarez-Gila, A. (2015). Automatic red-channel underwater image restoration. *J. Visual Communication Image Representation* 26, 132–145. doi: 10.1016/j.jvcir.2014.11.006

Gao, Y., Li, H., and Wen, S. (2016). Restoration and enhancement of underwater images based on bright channel prior. *Math. Problems Eng.* 2016, 1–15. doi: 10.1155/2016/3141478

Guo, C., Li, C., Guo, J., Loy, C. C., Hou, J., Kwong, S., et al. (2020)Zero-reference deep curve estimation for low-light image enhancement (IEEE) (Accessed 25 Apr. 2022).

Guo, X., Li, Y., and Ling, H. (2017). LIME: low-light image enhancement *via* illumination map estimation. *IEEE Trans. Image Process.* 262, 982–993. doi: 10.1109/TIP.2016.2639450

Han, J., Shoeiby, M., Malthus, T., Botha, E., Anstee, J., Anwar, S., et al. (2022). Underwater image restoration *via* contrastive learning and a real-world dataset. *Remote Sens.* 1417, 4297. doi: 10.3390/rs14174297

Hautière, N., Tarel, J.-P., DIDIER, A., and Dumont, E. (2008). Blind contrast enhancement assessment by gradient ratioing at visible edges. *Image Anal. Stereology* 27, 87-95. doi: 10.5566/ias.v27.p87-95

He, K., Sun, J., and Tang, X. (2011). Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33(12), 2341–2353. doi: 10.1109/TPAMI.2010.168

Hou, G., Li, J., Wang, G., Yang, H., Huang, B., and Pan, Z. (2020a). A novel dark channel prior guided variational framework for underwater image restoration. *J. Visual Communication Image Representation* 66, 102732. doi: 10.1016/j.jvcir.2019.102732

Hou, G., Zhao, X., Pan, Z., Yang, H., Tan, L., and Li, J. (2020b). Benchmarking underwater image enhancement and restoration, and beyond. *IEEE Access* 8, 122078–122091. doi: 10.1109/ACCESS.2020.3006359

Huang, S., Wang, K., Liu, H., Chen, J., and Li, Y. (2023) *Contrastive semi-supervised learning for underwater image restoration via reliable bank*. Available at: http://arxiv.org/abs/2303.09101 (Accessed 27 Apr. 2023).

Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., et al. (2021). Enlightengan: deep light enhancement without paired supervision. *IEEE Trans. image Process.* 30, 2340–2349. doi: 10.1109/TIP.2021.3051462

Jin, Y., Yang, W., and Tan, R. T. (2022). Unsupervised night image enhancement: when layer decomposition meets light-effects suppression. In: Computer Vision – ECCV 2022. eds., S. Avidan, G. Brostow, M. Cissé, G.M. Farinella and T. Hassner, Cham: Springer Nature Switzerland, pp.404–421.

Kar, A., Dhara, S. K., Sen, D., and Biswas, P. K. (2021). Zero-shot single image restoration through controlled perturbation of koschmieder's model (IEEE) (Accessed 14 May 2022).

Kimmel, R., Elad, M., Shaked, D., Keshet, R., and Sobel, I. (2003). A variational framework for retinex. *Int. J. Comput. Vision* 52, 7–23. doi: 10.1023/A:1022314423998

Koschmieder, H. (1924). Theorie der horizontalen sichtweite. *Beitrage zur Physik der freien Atmosphare*, 33–53.

Land, E. H. (1977). The retinex theory of color vision. *Sci. Am.* 2376, 108–128. doi: 10.1038/scientificamerican1277-108

Land, E. H., and McCann, J. J. (1971). Lightness and retinex theory. *J. Optical Soc. America* 611, 1. doi: 10.1364/JOSA.61.000001

Li, C., Anwar, S., Hou, J., Cong, R., Guo, C., and Ren, W. (2021a). Underwater image enhancement *via* medium transmission-guided multi-color space embedding. *IEEE Trans. Image Process.* 30, 4985–5000. doi: 10.1109/TIP.2021.3076367

Li, C., Guo, J., Cong, R., Pang, Y., and Wang, B. (2016a). Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. *IEEE Trans. Image Process.* 2512, 5664–5677. doi: 10.1109/TIP.2016.2612882

Li, C., Guo, C., Han, L., Jiang, J., Cheng, M.-M., Gu, J., et al. (2021b). Low-light image and video enhancement using deep learning: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 4412, 9396–9416. doi: 10.1109/TPAMI.2021.3126387

Li, C., Guo, C., and Loy, C. C. (2021c). Learning to enhance low-light image *via* zero-reference deep curve estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* 448, 4225–4238. doi: 10.1109/TPAMI.2021.3063604

Li, C., Guo, J., Pang, Y., Chen, S., and Wang, J. (2016b). Single underwater image restoration by blue-green channels dehazing and red channel correction (IEEE) (Accessed 2 Feb. 2023).

Li, C., Guo, J., Porikli, F., and Pang, Y. (2018a). LightenNet: a convolutional neural network for weakly illuminated image enhancement. *Pattern Recognition Lett.* 104, 15–22. doi: 10.1016/j.patrec.2018.01.010

Li, C., Guo, C., Ren, W., Cong, R., Hou, J., Kwong, S., et al. (2019). An underwater image enhancement benchmark dataset and beyond. *IEEE Trans. Image Process.* 29, 4376–4389. doi: 10.1109/TIP.2019.2955241

Li, Y., Lu, H., Li, K.-C., Kim, H., and Serikawa, S. (2018b). Non-uniform de-scattering and de-blurring of underwater images. *Mobile Networks Appl.* 232, 352–362. doi: 10.1007/s11036-017-0933-7

Li, T., Rong, S., Cao, X., Liu, Y., Chen, L., and He, B. (2020). Underwater image enhancement framework and its application on an autonomous underwater vehicle platform. *Optical Eng.* 5908, 1. doi: 10.1117/1.OE.59.8.083102

Liu, X., Gao, Z., and Chen, B. M. (2021). IPMGAN: integrating physical model and generative adversarial network for underwater image enhancement. *Neurocomputing* 453, 538–551. doi: 10.1016/j.neucom.2020.07.130

Liu, Y., Xu, H., Shang, D., Li, C., and Quan, X. (2019). An underwater image enhancement method for different illumination conditions based on color tone correction and fusion-based descattering. *Sensors* 1924, 5567. doi: 10.3390/s19245567

Liu, Y., Xu, H., Zhang, B., Sun, K., Yang, J., Li, B., et al. (2022). Model-based underwater image simulation and learning-based underwater image enhancement method. *Information* 134, 187. doi: 10.3390/info13040187

Lore, K. G., Akintayo, A., and Sarkar, S. (2017). LLNet: a deep autoencoder approach to natural low-light image enhancement. *Pattern Recognition* 61, 650–662. doi: 10.1016/j.patcog.2016.06.008

Lu, H., Li, Y., Uemura, T., Kim, H., and Serikawa, S. (2018). Low illumination underwater light field images reconstruction using deep convolutional neural networks. *Future Generation Comput. Syst.* 82, 142–148. doi: 10.1016/j.future.2018.01.001

Lu, H., Li, Y., Xu, X., Li, J., Liu, Z., Li, X., et al. (2016). Underwater image enhancement method using weighted guided trigonometric filtering and artificial light correction. *J. Visual Communication Image Representation* 38, 504–516. doi: 10.1016/j.jvcir.2016.03.029

Lu, H., Li, Y., Zhang, L., and Serikawa, S. (2015). Contrast enhancement for images in turbid water. *J. Optical Soc. America A* 325, 886. doi: 10.1364/JOSAA.32.000886

Lu, J., Yuan, F., Yang, W., and Cheng, E. (2021). An imaging information estimation network for underwater image color restoration. *IEEE J. Oceanic Eng.* 464, 1228–1239. doi: 10.1109/JOE.2021.3077692

Marques, T. P., and Branzan Albu, A. (2020)L ^{2} UWE: a framework for the efficient enhancement of low-light underwater images using local contrast and multi-scale fusion (IEEE) (Accessed 21 Dec. 2021).

Menaker, D., Treibitz, T., and Avidan, S. (2017). Color restoration of underwater images. *In Proceedings of the British machine vision conference (BMVC),* Eds., T.K. Kim, S. Zafeiriou, G. Brostow and K. Mikolajczyk (Durham, UK: BMVA Press.) 44.1-44.12.

Mittal, A., Moorthy, A. K., and Bovik, A. C. (2012). No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.* 2112, 4695–4708. doi: 10.1109/TIP.2012.2214050

Narasimhan, S. G., and Nayar, S. K. (2000). Chromatic framework for vision in bad weather (IEEE Comput. Soc) (Accessed 19 Jun. 2022).

Narasimhan, S. G., and Nayar, S. K. (2003). Contrast restoration of weather degraded images. *IEEE Trans. ON Pattern Anal. AND Mach. Intell.* 256, 12. doi: 10.1109/TPAMI.2003.1201821

Narasimhan, S. G., and Nayar, S. K. (2008). Vision and the atmosphere (ACM Press) (Accessed 19 Jun. 2022).

NOAA (2022) *What is the "deep" ocean?: ocean exploration facts: NOAA office of ocean exploration and research*. Available at: https://oceanexplorer.noaa.gov/facts/deep-ocean.html (Accessed 2 Feb. 2023). doi: 10.6119/JMST.201808_26(4).0006

Pan, P., Yuan, F., and Cheng, E. (2018). Underwater image de-scattering and enhancing using dehazenet and HWD. *J. Mar. Sci. Technol.* 264, 6. doi: 10.6119/JMST.201808_26(4).0006

Panetta, K., Gao, C., and Agaian, S. (2016). Human-Visual-System-Inspired underwater image quality measures. *IEEE J. Oceanic Eng.* 413, 541–551. doi: 10.1109/JOE.2015.2469915

Paulus, E. (2021). Shedding light on deep-Sea biodiversity–a highly vulnerable habitat in the face of anthropogenic change. *Front. Mar. Sci.* 8, 667048. doi: 10.3389/fmars.2021.667048

Peng, Y.-T., Cao, K., and Cosman, P. C. (2018). Generalization of the dark channel prior for single image restoration. *IEEE Trans. Image Process.* 276, 2856–2868. doi: 10.1109/TIP.2018.2813092

Peng, Y.-T., and Cosman, P. C. (2017). Underwater image restoration based on image blurriness and light absorption. *IEEE Trans. Image Process.* 264, 1579–1594. doi: 10.1109/TIP.2017.2663846

Peng, Y.-T., Zhao, X., and Cosman, P. C. (2015). Single underwater image enhancement using depth estimation based on blurriness (IEEE) (Accessed 30 Aug. 2021).

Peng, L., Zhu, C., and Bian, L. (2023). "U-Shape transformer for underwater image enhancement," in *Computer Vision–ECCV 2022 Workshops*, L. Karlinsky, T. Michaeli and K. Nishino, eds., (Cham: Springer Nature Switzerland). 290–307. doi: 10.1007/978-3-031-25063-7_18

Porto Marques, T., Branzan Albu, A., and Hoeberechts, M. (2019). A contrast-guided approach for the enhancement of low-lighting underwater images. *J. Imaging* 510, 79. doi: 10.3390/jimaging5100079

Ren, X., Li, M., Cheng, W.-H., and Liu, J. (2018). "Joint enhancement and denoising method *via* sequential decomposition," in *2018 IEEE international symposium on circuits and systems (ISCAS)* (Florence, Italy: IEEE), 1–5. doi: 10.1109/ISCAS.2018.8351427

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: convolutional networks for biomedical image segmentation. In: *Medical image computing and computer-assisted intervention – MICCAI 2015, lecture notes in computer science* (Cham: Springer International Publishing). Available at: http://link.springer.com/10.1007/978-3-319-24574-4_28 (Accessed 27 Oct. 2021).

Saleh, A., Sheaves, M., Jerry, D., and Azghadi, M. R. (2022). *Adaptive uncertainty distribution in deep learning for unsupervised underwater image enhancement*. Available at: http://arxiv.org/abs/2212.08983 (Accessed 27 Apr. 2023).

Song, W., Wang, Y., Huang, D., and Tjondronegoro, D. (2018). A rapid scene depth estimation model based on underwater light attenuation prior for underwater image restoration. In: *Advances in multimedia information processing – PCM 2018, lecture notes in computer science* (Cham: Springer International Publishing). Available at: http://link.springer.com/10.1007/978-3-030-00776-8_62 (Accessed 31 Mar. 2022).

Tan, R. T. (2008). "Visibility in bad weather from a single image.," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (Anchorage, AK, USA: IEEE). 1–8. doi: 10.1109/CVPR.2008.4587643

Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., and Li, H. (2022). Uformer: a general U-shaped transformer for image restoration (IEEE) (Accessed 11 Feb. 2023).

Wang, Y.-F., Liu, H.-M., and Fu, Z.-W. (2019c). Low-light image enhancement *via* the absorption light scattering model. *IEEE Trans. Image Process.* 2811, 5679–5690. doi: 10.1109/TIP.2019.2922106

Wang, Y., Song, W., Fortino, G., Qi, L.-Z., Zhang, W., and Liotta, A. (2019b). An experimental-based review of image enhancement and image restoration methods for underwater imaging. *IEEE Access* 7, 140233–140251. doi: 10.1109/ACCESS.2019.2932130

Wang, R., Zhang, Q., Fu, C.-W., Shen, X., Zheng, W.-S., and Jia, J. (2019a) Underexposed photo enhancement using deep illumination estimation (IEEE) (Accessed 2 Feb. 2023).

Wang, N., Zheng, H., and Zheng, B. (2017). Underwater image restoration *via* maximum attenuation identification. *IEEE Access* 5, 18941–18952. doi: 10.1109/ACCESS.2017.2753796

Wen, H., Tian, Y., Huang, T., and Gao, W. (2013). Single underwater image enhancement with a new optical model (IEEE) (Accessed 25 Mar. 2022).

Yan, K., and Zhou, Y. (2020)Underwater image processing by an adversarial network with feedback control. In: *Pattern recognition and computer vision, lecture notes in computer science* (Cham: Springer International Publishing). Available at: http://link.springer.com/10.1007/978-3-030-60633-6_38 (Accessed 19 Mar. 2023).

Yang, M., and Sowmya, A. (2015). An underwater color image quality evaluation metric. *IEEE Trans. Image Process.* 2412, 6062–6071. doi: 10.1109/TIP.2015.2491020

Yu, R., Liu, W., Zhang, Y., Qu, Z., Zhao, D., and Zhang, B. (2018)DeepExposure: learning to expose photos with asynchronously reinforced adversarial learning. In: *Advances in neural information processing systems* (Curran Associates, Inc). Available at: https://proceedings.neurips.cc/paper/2018/hash/a5e0ff62be0b08456fc7f1e88812af3d-Abstract.html (Accessed 2 Nov. 2022).

Zhang, Y., Guo, X., Ma, J., Liu, W., and Zhang, J. (2021). Beyond brightening low-light images. *Int. J. Comput. Vision* 1294, 1013–1037. doi: 10.1007/s11263-020-01407-x

Zhang, Q., Nie, Y., and Zheng, W.-S. (2019a). "Dual illumination estimation for robust exposure correction," in *Computer graphics forum* (England: Wiley Online Library), 243–252. doi: 10.1111/cgf.13833

Zhang, M., and Peng, J. (2018). Underwater image restoration based on a new underwater image formation model. *IEEE Access* 6, 58634–58644. doi: 10.1109/ACCESS.2018.2875344

Zhang, Y., Zhang, J., and Guo, X. (2019b). Kindling the darkness: a practical low-light image enhancer (Accessed 27 Sep. 2022).

Zhang, W., Zhuang, P., Sun, H.-H., Li, G., Kwong, S., and Li, C. (2022). Underwater image enhancement *via* minimal color loss and locally adaptive contrast enhancement. *IEEE Trans. Image Process.* 31, 3997–4010. doi: 10.1109/TIP.2022.3177129

Zhao, X., Jin, T., and Qu, S. (2015). Deriving inherent optical properties from background color and underwater image enhancement. *Ocean Eng.* 94, 163–172. doi: 10.1016/j.oceaneng.2014.11.036

Zhou, J., Liu, Z., Zhang, W., Zhang, D., and Zhang, W. (2021a). Underwater image restoration based on secondary guided transmission map. *Multimedia Tools Appl.* 805, 7771–7788. doi: 10.1007/s11042-020-10049-7

Zhou, J., Wang, Y., Zhang, W., and Li, C. (2021b). Underwater image restoration *via* feature priors to estimate background light and optimized transmission map. *Optics Express* 2918, 28228. doi: 10.1364/OE.432900

Zhou, J., Yang, T., Ren, W., Zhang, D., and Zhang, W. (2021c). Underwater image restoration *via* depth map and illumination estimation based on a single image. *Optics Express* 29 (19), 29864. doi: 10.1364/OE.427839

Zhu, A., Zhang, L., Shen, Y., Ma, Y., Zhao, S., and Zhou, Y. (2020). Zero-shot restoration of underexposed images *via* robust retinex decomposition (IEEE) (Accessed 9 Aug. 2022).

Zhuang, P., Li, C., and Wu, J. (2021). Bayesian Retinex underwater image enhancement. *Eng. Appl. Artif. Intell.* 101, 104171. doi: 10.1016/j.engappai.2021.104171

Zhuang, P., Wu, J., Porikli, F., and Li, C. (2022). Underwater image enhancement with hyper-laplacian reflectance priors. *IEEE Trans. Image Process.* 31, 5442–5455. doi: 10.1109/TIP.2022.3196546

frontiers | Frontiers in Marine Science

TYPE Methods
PUBLISHED 02 June 2023
DOI 10.3389/fmars.2023.1171625

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Jorge Paramo,
University of Magdalena, Colombia
Kai Wang,
Shanghai Ocean University, China

*CORRESPONDENCE
Daniel Marrable
marrabld@gmail.com

RECEIVED 22 February 2023
ACCEPTED 16 May 2023
PUBLISHED 02 June 2023

CITATION
Marrable D, Tippaya S, Barker K, Harvey E, Bierwagen SL, Wyatt M, Bainbridge S and Stowar M (2023) Generalised deep learning model for semi-automated length measurement of fish in stereo-BRUVS. *Front. Mar. Sci.* 10:1171625. doi: 10.3389/fmars.2023.1171625

COPYRIGHT
© 2023 Marrable, Tippaya, Barker, Harvey, Bierwagen, Wyatt, Bainbridge and Stowar. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Generalised deep learning model for semi-automated length measurement of fish in stereo-BRUVS

Daniel Marrable[1*], Sawitchaya Tippaya[1], Kathryn Barker[1], Euan Harvey[2], Stacy L. Bierwagen[3], Mathew Wyatt[4], Scott Bainbridge[3] and Marcus Stowar[3]

[1]Curtin Institute for Computation, Curtin University, Perth, WA, Australia, [2]Curtin University, School of Molecular and Life Sciences, Perth, WA, Australia, [3]Australian Institute of Marine Science, Townsville, QLD, Australia, [4]Australian Institute of Marine Science, Indian Ocean Marine Research Centre, The University of Western Australia, Perth, WA, Australia

Assessing the health of fish populations relies on determining the length of fish in sample species subsets, in conjunction with other key ecosystem markers; thereby, inferring overall health of communities. Despite attempts to use artificial intelligence (AI) to measure fish, most measurement remains a manual process, often necessitating fish being removed from the water. Overcoming this limitation and potentially harmful intervention by measuring fish without disturbance in their natural habitat would greatly enhance and expedite the process. Stereo baited remote underwater video systems (stereo-BRUVS) are widely used as a non-invasive, stressless method for manually counting and measuring fish in aquaculture, fisheries and conservation management. However, the application of deep learning (DL) to stereo-BRUVS image processing is showing encouraging progress towards replacing the manual and labour-intensive task of precisely locating the heads and tails of fish with computer-vision-based algorithms. Here, we present a generalised, semi-automated method for measuring the length of fish using DL with near-human accuracy for numerous species of fish. Additionally, we combine the DL method with a highly precise stereo-BRUVS calibration method, which uses calibration cubes to ensure precision within a few millimetres in calculated lengths. In a human versus DL comparison of accuracy, we show that, although DL commonly slightly over-estimates or under-estimates length, with enough repeated measurements, the two values average and converge to the same length, demonstrated by a Pearson correlation coefficient (r) of 0.99 for n=3954 measurement in 'out-of-sample' test data. We demonstrate, through the inclusion of visual examples of stereo-BRUVS scenes, the accuracy of this approach. The head-to-tail measurement method presented here builds on, and advances, previously published object detection for stereo-BRUVS. Furthermore, by replacing the manual process of four careful mouse clicks on the screen to precisely locate the head and tail of a fish in two images, with two

fast clicks anywhere on that fish in those two images, a significant reduction in image processing and analysis time is expected. By reducing analysis times, more images can be processed; thereby, increasing the amount of data available for environmental reporting and decision making.

KEYWORDS

stereo-BRUVS, deep learning, automated fish length, photogrammetry, machine learning, cameras

1 Introduction

It is estimated that one third of global fish stocks are overfished (Duarte et al., 2020) which impacts the ecosystem services provided by fish (Steneck and Pauly, 2019). Numerous management actions at local, national and international scales will be required to rebuild fish stocks by improving governance, including lowering fishing pressure; implementing harvest controls which limit the types of gear used and the size and number of fish caught; and the use of closed-area management or sanctuaries (MacNeil et al., 2020; Melnychuk et al., 2021). Fishery-dependent information from traps, hook and line, trawls and nets has provided much of the data for monitoring the status of fish populations. With the implementation of closed areas and sanctuaries, there has been an increase in the interest of fishery-independent sampling techniques, as many of the conventional sampling techniques are not permissible. Fishery-independent techniques have largely been based on underwater visual census (UVC) (Brock, 1954). Baited remote underwater video systems (BRUVS) (Ellis and DeMartini, 1995; Cappo et al., 2001; Cappo et al., 2003) can collect a relative abundance of data on a range of fish species from numerous habitats and depths (Harvey et al. 2021). While estimates of abundance are an important metric, accurate and reliable information on the length and size of fish within wild populations is more useful (Jennings and Polunin, 1997; Jennings and Kaiser, 1998). This is because it has been shown that fishing and other impacts decrease the mean length, length frequency and biomass of fish populations (Roberts, 1995; McClanahan et al., 1999). For UVC, biomass is calculated from fish length based on visual estimates by SCUBA divers (Wilson et al., 2018) with the standing biomass of fish thought to be a good metric for expressing the health of fish populations (Friedlander and DeMartini, 2002; Seguin et al., 2022). But these estimates have been demonstrated to be neither accurate nor precise, which can affect biomass estimates (Harvey et al., 2002). Stereo video systems are a more accurate and precise technique for non-destructively estimating the lengths of fish (Harvey and Shortis, 1995; Harvey et al., 2001a; Harvey et al., 2010) and have been modified for use by SCUBA divers (Goetze et al., 2019), remotely operated vehicles (ROVs) (Schramm et al., 2020; Jessop et al., 2022; Hellmrich et al., 2023) and BRUVS (Harvey et al., 2007; Langlois et al., 2020; Harvey et al. 2021).

Determining the size and quantity of fish populations in a specific area is crucial to understanding and assessing the health of fish stocks so that informed decisions can be made about sustainable fishing and management practices (Pauly et al., 2002). Fish measurement provides important information in the context of stock assessment by monitoring changes in the size of fish, which gives insight into the impacts of fishing and other factors on the overall health of fish communities and ecosystems.

Automation has the potential to improve the accuracy, efficiency and consistency of fish measurement (e.g. Shortis, 2015; Marrable et al., 2022) to reliably increase the accuracy of stock assessment information that can then be used to support and design improvements to sustainable fishing practices which protect fish populations and ecosystems. Some benefits of using automation include: **1) improved accuracy** – automated systems can measure fish more precisely than manual methods, reducing the potential for human error; **2) increased efficiency** – automated systems can process large numbers of fish much more quickly than manual methods, reducing the time and effort required for stock assessments; **3) consistent data** – automated systems can provide consistent and standardised measurements, reducing the potential for variation due to differences in the way measurements are taken; **4) reduced labour** – automated systems can reduce the need for manual labour, freeing up resources for other tasks and potentially reducing costs.

1.1 Traditional approaches to measuring fish

Existing methods that enhance manual measurement by using automation and computer vision have the potential to support fishing operations and ecosystem monitoring; however, these remain inaccessible to most small-scale fisheries due to their associated high cost (Andrialovanirina et al., 2020). Even systems that use remote surveillance monitoring to measure, process and count discarded fish *via* video record once the vessel has returned to port have shown that the analytical processing time required is equally as labour intensive (Needle et al., 2015; French et al., 2019). Such examples provide further justification for the need of computer vision tools to increase the efficiency monitoring for managing vessel operations. Similar challenges are faced by those conducting research in aquaculture and fish ecology. There is a seemingly exponential trend in the availability of automated fish detection tools for researchers, yet their documented use is still

minimal, with researchers also requiring ways to measure and track fish (Bradley et al., 2019; Lopez-Marcano et al., 2021).

Assessing the health of fish populations depends on determining the average length of fish in sample population subsets and inferring health in conjunction with other key ecosystem markers. Methods applying the length-based measurement of fish for assessing the health of fisheries have been around for decades (Pauly and Morgan, 1987) with few technological advancements until recently. Manual measurement remains the principal tool in collecting essential management information on board fishing vessels. However, this method is documented as highly time consuming and involves considerable, and potentially harmful, handling of fish to gain accurate measurements (Upton and Riley, 2013). Traditionally, evaluating stock levels has relied on manually measuring fish length, as it is frequently the only possibility where monitoring is limited and collecting length measurements is easier than quantifying a total catch (Rudd and Thorson, 2018). However, this method does not consider the fluctuations in fish recruitment and death rates over time, which is crucial for comprehending the indirect impacts of fishing on predator–prey dynamics and for identifying the factors that influence the structure of fish communities on a larger scale (Jennings and Polunin, 1997). Average length is also considered an operational indicator of fishing impact; whereas indices on the composition of species assemblage are difficult to interpret, average length is well understood and reference points can be set (Rochet and Trenkel, 2003). As well as causing impacts on targeted species, commercial fishing affects bycatch, including by-product and discarded/released species; and sometimes habitats, when fishing gear (e.g. demersal trawling) interacts with the sea floor or benthic zones (Little and Hill, 2021). An increasing range of mechanisms and technical tools is being used to reduce interactions with seabirds, marine mammals, reptiles and other vulnerable species. Such bycatch-reduction measures include tori lines, sprayers, and seal and turtle excluder devices (Cresswell et al., 2022). In Australia, as around the world, guidelines and rules on fish measurement methodology and length quotas are enacted and overseen by governments[1].

1.2 The move toward automation

Monitoring devices and advances in data processing and analysis techniques can, and should, form part of an effective monitoring approach. However, data or capacity limitation is widespread in global fisheries resulting in ineffective or non-existent management as a result of this lack of data and/or an inability to generate statistical estimates of stock status. Significant improvements in management outcomes, leading to conservation and livelihood benefits, could be achieved through cost-effective analytical approaches; these exist, but are hampered by a range of challenges, including data availability and requirements; resources for processing and analysis; and a lack of understanding of costs and advantages (Dowling et al., 2016; Cresswell et al., 2022). Deep learning (DL) can address these challenges by replacing the manual, labour-intensive task of precisely locating the heads and tails of fish with computer-vision-based algorithms (e.g. Marrable et al., 2022). White et al. (2006) were the first to test this method with computer vision on a fishing vessel. Measurement using digital imagery is a growing field and has been successfully implemented with both single image (e.g. Lezama-Cervantes et al., 2017; Monkman et al., 2019; Andrialovanirina et al., 2020; Wibisono et al., 2022), and stereo image (e.g. Johansson et al., 2008; Shafait et al., 2017; Suo et al., 2020; Connolly et al., 2021; Lopez-Marcano et al., 2021; Marrable et al., 2022). Datasets now also exist to explicitly support the development of DL algorithms; for instance, segmentation, classification and size estimation (e.g. DeepFish, Garcia-d'Urso et al., 2022).

Automated fish detection has been demonstrated using a range of computer vision methods of measurement targeting single species for aquaculture (Atienza-Vanacloig et al., 2016; Shi et al., 2020; Yang et al., 2021). Some invasive methods of measurement involve channelling fish past stationary cameras (Miranda and Romero, 2017; Shafait et al., 2017), or methods which use active sources of light, such as sonar (Uranga et al., 2017), which are potentially stressful to the fish. Furthermore, removing fish from the water (White et al., 2006) or measurement on board trawlers (Monkman et al., 2019) adds to fish mortality. These challenges highlight the importance of developing automated methods for non-invasive means of measurement, such as BRUVS.

Although there have been advances in using DL for image analysis, video imagery presents additional complexities and requirements, particularly with regard to curated and structured data (e.g. Marrable et al., 2022).

Recent reviews of machine learning in aquaculture found that there is a need for DL and neural networks to optimise current approaches but have also identified certain pitfalls in the process, including noise, occlusions and dynamic viewing spaces (Yang et al., 2021; Zhao et al., 2021).

Stereo baited remote underwater video systems (stereo-BRUVS) are widely, and increasingly, used as a non-invasive, stressless method for counting and measuring fish in aquaculture, fisheries and conservation management (Harvey and Shortis, 1995; Harvey et al., 2021). Recently, Marrable et al. (2022) demonstrated the application of DL to stereo-BRUVS imagery for the semi-automation of fish identification and early success with species identification. Extending the application of DL to automate fish length measurement would greatly enhance and advance marine environment monitoring, speeding up data collation on localised fish populations and increasing the amount of data that can be processed and used for environmental reporting and decision making. The current limitation of BRUVS is that the data processing is a highly time-consuming manual exercise, prone to human error and is costly, delaying the production of length data and limiting how much BRUVS imagery can be processed (Connolly et al., 2021; Marrable et al., 2022). However, as with species identification, mean length data is highly valuable for determining frequency distributions of fish populations and the

1 https://www.daf.qld.gov.au/business-priorities/fisheries/recreational/recreational-fishing-rules/measuring.

spatial and temporal changes required for environmental assessment and reporting. In addition to cost and processing time, BRUVS is limited by the MaxN ecological abundance metric (Whitmarsh et al., 2017), creating an opportunity for a much larger use of the data held within a video, such as including fishery-independent assessments of fishing pressure. Recent use of open-source image processing software to measure fisheries catch has also been successful for a wide range of fish sizes (Andrialovanirina et al., 2020).

1.3 A semi-automated and generalised method of length measurement

Here we present a semi-automated and generalised method of measuring the length of fish using DL with near-human accuracy for numerous species of fish across a wide range of habitats. Speed of analysis is therefore much increased, and demonstrates progress towards the use of stereo-BRUVS for length measurement in fisheries, aquaculture and marine ecology research applications.

2 Method

In this section, we describe the DL method used for locating the heads and tails of fish, combined with a highly precise stereo-BRUVS calibration method method (Shortis, 2015), which makes use of calibration cubes to ensure precision in calculated lengths to the nearest millimetre. Once trained and deployed, this semi-automated approach solves the problem of finding the same fish in both images; that is, the 'fish correspondence challenge', with ecologists only having to select the same fish in the left and right images by clicking anywhere on the body, eliminating the need for four very precise clicks on the head and tail in both images. The method is illustrated in Figure 1 and examples of the results in Figure 2.

2.1 Datasets

The fish length measurement data made available for this study study (Australian Institute Of Marine Science, 2020) was taken from OzFish stereo-BRUVS imagery along with annotations conducted by fish ecologists using EventMeasure. In order to develop a training dataset for the DL model, the head and tail annotations, which were initially made manually by the ecologists, were extracted by exporting the frame number and pixel location of each annotation in the frame from the data files.

The original OzFish dataset has 37695 measurements inside unique bounding boxes which indicate the location and extent of a fish and include markers which identify its head and tail. Crops from pairs of stereo images were taken from the full images to create head and tail stereo pairs. Small fish, or ones far away in the background, were excluded by filtering out any fish objects smaller than 200 pixels in either height or length. Another filter was applied to exclude fish that had been measured with a root mean square (RMS) value >20 mm. The RMS value is calculated by the photogrammetry library in EventMeasure and is an indicator of how close two corresponding points in each image are to the epipolar line calculated by the opposite point. An RMS value greater than 20 mm is considered by SeaGIS (outlined in the EventMeasure software manual) as an imprecise measurement or error in calibration and, therefore, was discarded in this study. This reduced the number of images for training to 15558 stereo pairs of cropped fish images.

2.2 Data preparation

The annotated data in OzFish did not include head or tail labels and does not store the annotations in any particular order. There was no consistent order to which the heads and tails were labelled. Head and tail labels are required to train the DL model to classify them. Therefore, a systematic review of the images was conducted to reorder many of the annotations, resulting in a dataset in which two labels, 'head' and 'tail' in consistent order, were reliably applied to all of the points for training the DL model.

The final step, before training and testing the system, was to split the data between 'in-sample' and 'out-of-sample' datasets. The videos in OzFish have had the metadata removed before publishing, although the data were given prefix letters in their filenames to indicate they were taken from different deployments and at different locations. Calibration files required for photogrammetry were only published for the images with the prefix A and E. As these calibration files are needed to do a human versus machine comparison, they were withheld from any training or validation and made up the out-of-sample data used for testing algorithm performance. Images with prefix B and G were not published with calibration files; however, these files were not needed for training the head and tail detection model and made up the in-sample training data.

After filtering the data, a total of 13555 stereo pairs of cropped fish images remained with correct head and tail labels. The available data for training and testing amounted to 59 unique family, 153 unique genus and 319 different species. The in-sample data were split 70% (5348 stereo pairs) for training, and 30% (2292 stereo pairs) for validation and hyperparameter tuning. In this study, the calibration file verification process, taken to ensure that the ground-truth length in OzFish dataset and calculated length using photogrammetry was consistent, resulted in approximately 30% of the out-of-sample data (1761 stereo pairs) being removed. The remaining out-of-sample data comprised 4154 stereo pairs.

2.3 Model training

This study used You Only Look Once (YOLO; Redmon et al., 2016) a type of DL model used in object-detection algorithms. Specifically, the YOLOv5 model, which has been pre-trained on the Common Objects in Context[2] (COCO) dataset, was chosen for its

2 https://cocodataset.org.

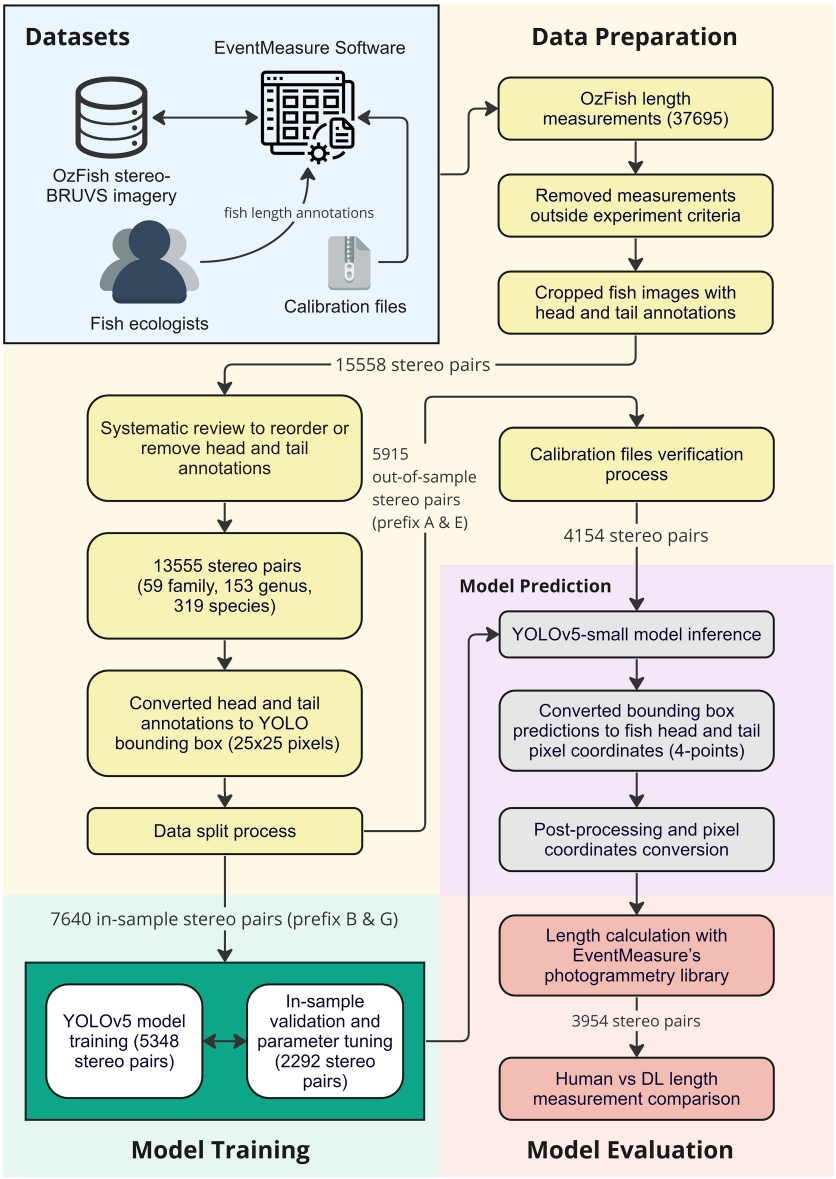


FIGURE 1
Illustrates the workflow for data preparation, model training and model evaluation.

ability to handle various sizes, numbers of classes, and computational requirements. The variant used in this study was the 'YOLOv5 small' model. To adapt the model for head and tail detection, transfer learning was employed, which built on knowledge gained from the pre-trained model while reducing the amount of training data and time needed. A subset of the in-sample dataset was used to retrain the model according to the standard procedure outlined on the YOLOv5 website[3].

The YOLOv5 model needs to be trained by defining the extent of an object of interest (heads and tails in this case) by defining a bounding box. Therefore, the head and tail points in the training data were converted to bounding boxes by defining a box of 25 × 25 pixels around the head and tail points, respectively. Finally, the in-sample training and validation fish crop images with head and tail labels were used to train the YOLOv5 small model. The early-stopping method was also implemented in this study to avoid overfitting the model.

2.4 Model prediction

The head and tail predictions from the object-detection model were converted to overall fish length by first taking the bounding box predictions from the trained DL model and converting them to points by using the centre location of the box in stereo image pairs. On occasions when the DL model failed to find one or two of either a head or a tail in both images, the location of the missing feature

3 https://github.com/ultralytics/yolov5 Access Date (Nov 22, 2022).



FIGURE 2
Presents four out-of-sample examples of automated fish length measurements using the method described in this study. The example presents fish of different sizes, habitat and distance from the camera.

was estimated by taking the reflection of one the classifier feature locations in the bounding box of the fish. On occasions when the model returned more than one candidate for a head or tail, the one with the highest confidence score was chosen. On the occasions when predicted head and tail points were inconsistent in both left and right cropped fish images; for example, if head or tail points were swapped, the predicted result was discarded as an incorrect measurement. Once the four required points were returned by the model, the camera calibration files were used along with EventMeasure's photogrammetry library to calculate the length of the fish.

2.5 Model evaluation

The out-of-sample dataset was used for evaluating the performance of the model and gives an indication of model generalisability and performance in different domains. Inference for both heads and tails was performed on the 4154 out-of-sample data (stereo pairs of cropped fish images), and heads and tails pixel coordinates were converted to the original scale of stereo-BRUVS imagery. EventMeasure's stereophotogrammetry tool was used to calculate the length of a fish from the four predicted points of head and tail pairs. Two hundred predictions were removed by the post-processing steps described in the previous section, and the remaining 3954 automated measurements were then compared to the manual measurements made by the fish ecologists. Results are presented in Figures 3, 4.

2.5.1 Recall, Precision and F_1 Score

Simplifying model performance for fish head and tail detection into a single metric can be beneficial. One such metric is the F_1 score, which is a combination of recall and precision. Recall is the likelihood of detecting all actual positive instances, while precision is the proportion of true positive (TP) predictions out of all positive predictions. False negative (FN) represent the number of predictions the model missed and false positive (FP) predictions are incorrectly predicted results. The F_1 score is calculated by taking the harmonic mean of recall and precision.

The recall, precision and F_1 score for fish head and tail detection are presented in Table 1.

$$Recall = \frac{TP}{TP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

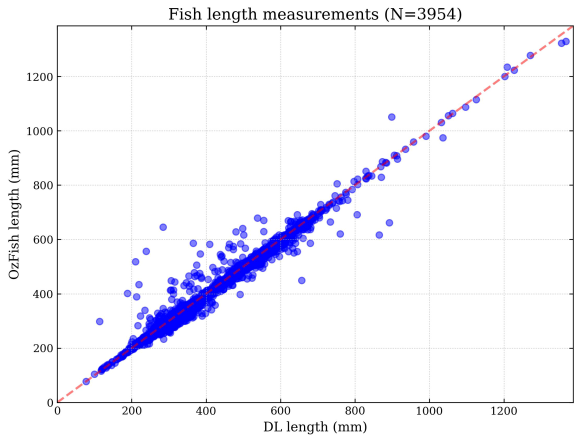


FIGURE 3
Human versus DL comparison showing how DL and photogrammetry-derived length compares with human and photogrammetry-derived length for the same fish. The Pearson correlation coefficient is 0.99 indicating that even though DL sometimes overestimates or underestimates the length compared with a manual measurement by an ecologist; with repeat measurements, the total length estimates average to be very similar.

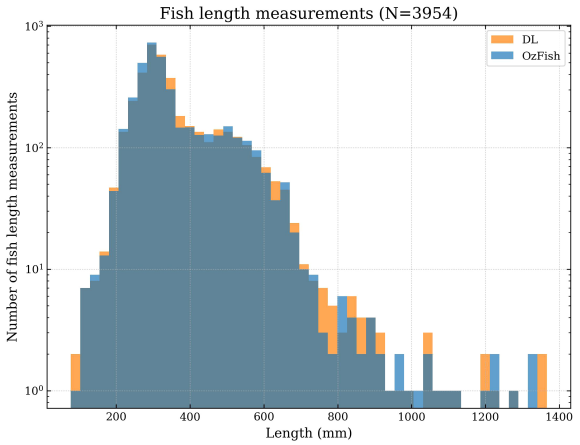


FIGURE 4
Histogram of the human versus DL comparison demonstrating the density of the number of length measurements. A higher density of points indicates the total number of measurements aggregate to close agreement.

$$F_1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (3)$$

2.5.2 Human–machine comparison

The Pearson correlation coefficient used for the human–machine comparison was calculated by:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad (4)$$

Where:

x are the individual DL inference length results
$\bar{x}$ is the average DL length
y_i are the individual human annotated length results
$\bar{y}$ is the average human annotated length

3 Results

The following section presents the results of the human–machine comparison by comparing the machine learning and photogrammetry-derived length measurements with the ecologists' manual measurements (Figure 3) and the density of those measurements and compared in Figure 4. The results presented here are calculated from the out-of-sample data. Table 1 shows the DL precision (P), recall (R) and F_1 for heads, tails and the combination of both.

4 Discussion

The semi-automated method presented in this paper demonstrates the potential to rapidly increase analysis time and decrease reporting time for assessing fish biomass. Challenges remain for a completely autonomous solution, some of which are discussed below.

4.1 Semi-automation of length measurement

The challenge of applying this model in real-world scenarios is that the model cannot currently match the fish in the corresponding left and right images. This was not a problem when building and testing the model, as the data were already analysed by experienced ecologists who had matched the stereo image pairs. To address this challenge, the DL model was adapted to communicate with Event Measure; wherein, the DL model requires an ecologist to click anywhere on the body of the fish in both images. Inference on the length is conducted after the ecologist has solved the image correspondence problem by identifying the same fish in each of the left and right images. The fish is then precisely cropped from the stereo-BRUVS image using the DL method described in Marrable et al. (2022), which places a bounding box over the fish, then parsed by the head and tail DL model. Without isolating the fish first, the model returns all of the heads and tails of all the fish it finds with no correspondence data to match them. The head and tail locations are returned to EventMeasure which automatically calculates the length of the fish using its photogrammetry library. This reduces the number of mouse clicks on the screen, from four precise clicks (i.e. left head, left tail, right head, right tail) to two. Additionally, placing clicks anywhere on the body is significantly faster and requires much less precision. This semi-automated method of length measurement has the potential to significantly increase analysis speed.

Furthermore, by requiring ecologists to choose the corresponding fish individuals, users can draw on their contextual knowledge to wait for a moment when a fish is the best pose for measurement and not occluded by other fish, seagrass, the BRUVS bait bag or other objects. This reduces false positive detection. Context is something that is not currently possible by using computer vision alone.

TABLE 1 Deep learning precision (P), recall (R) and F_1 for classification.

Feature	Images	Labels	P	R	F_1
Head	8308	8308	77.50%	70.50%	73.83%
Tail	8308	8308	77.20%	69.50%	73.15%
Both	8308	16616	77.40%	70.00%	73.51%

4.2 Sources of error

The DL model cannot correspond the head and tail of a given fish and, therefore, the largest source of error is incorrect correspondence; that is, when a head and tail pair are matched to two different fish. This is because the model searches within the bounding box for features that look like heads and tails and returns the match with the highest confidence. This works well when there is only one matching pair; however, there are occasions when there are heads and tails belonging to many fish. The model has no knowledge of correspondence and so matches them based on the highest confidence level, and sometimes pairs them incorrectly. An example of this is seen in Figure 5. This results in either the incorrect length being calculated from the photogrammetry, or the RMS value returned from EventMeasure being >20, so no length is reported.

Figure 5A shows an example where two fish tails fall within the bounding box and the model identifies the wrong tail. This false positive is seen most commonly where fish are schooling and swimming between 30° and 45° to the plane of the camera. Angles within this span produce a large bounding box with more likelihood that tails from other fish will be captured. One way to reduce this effect is to automate a rotation of the bounding box, Figure 5B, or the image in sympathy with the orientation of the fish to reduce the empty space in the bounding box. Automating this process remains a challenge, as even establishing that a false positive detection has occurred would require logic and processing beyond the capability of the current model. There are published detection models that use rotated labels (Li et al., 2018) for ship identification in satellite images; but, as yet, YOLOv5 does not have the ability to train using rotated bounding boxes. Addressing these false positive cases remains the subject of ongoing research.

4.3 Stereo calibration

Harvey and Shortis (1998) highlight the importance of precise measurement systems for accurate length. This was also the objective of this approach by using the OzFish dataset for model training and validation. The OzFish data were calibrated using the calibration cube method (Shortis, 2015) which is more accurate and precise than using 2D calibration patterns as reported by Boutros et al. (2015) in their comparison study.

4.4 Model generalisability

Previous published models capable of automating the length measurement of fish have either used a single camera out of water (Monkman et al., 2019); been limited to a single species (White et al., 2006); or used less accurate stereophotogrammetry calibration methods (Tonachella et al., 2022). The model presented in this study was trained and tested on 319 unique species of fish, making it much more generalisable than any other previously published model. The data used to train this model was restricted to the species in the OzFish dataset, which includes those mostly found along the coast of Western Australian. However, the species richness and diversity shows evidence that the model generalises across different species with varying colour, texture and morphometrics. An effort to separate in-sample and out-of-sample data was made to give some indication of model generalisability by training and testing to data collected at different dates and locations. How well the model works with species outside the OzFish data will be the focus of future work. For applications in marine environments with species not included in the OzFish data, the method described in this study should be repeated with a new training corpus that includes species in which users wish to measure.

4.5 Challenges with data quality

One reason for choosing the OzFish dataset for DL training was because the data were annotated by expert fish ecologists. However, when auditing the DL data there were still errors in the labelling. Some errors included head and tail points that seemed to be systematically shifted a few pixels away from the head and tail of the fish, which may have been caused by incorrect synchronisation of the stereo-BRUVS. There were also some instances where labels were randomly out of place, such as labels placed on a rock.

One issue that continues to be a challenge for computer-vision-based DL is that it is so far incapable of using context in the way fish ecologists do to help them label fish. For example, in the OzFish dataset, where a fish was partially occluded by an object, labels were placed where heads or tails would logically be expected, estimated by ecologists from experience and numerous previous observations of similar fish. When such an example is viewed by a computer-vision algorithm which, unlike an ecologist, cannot extrapolate from the context, the algorithm may see a label on a rock and interpret that

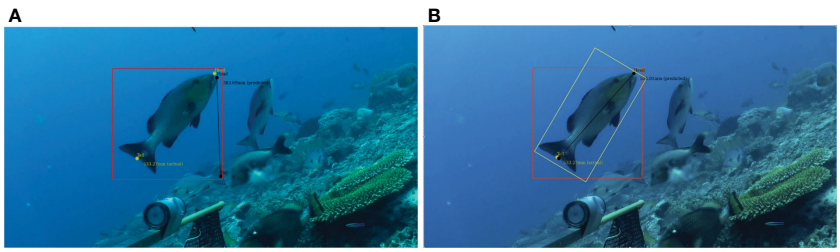


FIGURE 5
Example of a false positive detection of a tail leading to an incorrect length measurement; **(A)** two fish tails fall withinthe bounding box and the model identifies the wrong tail. **(B)** the yellow box demonstrates that rotating theobject-detection bounding box, would eliminate the second tail from the area and correct the false tail label.

rock as a fish head or tail. In such cases, those data must be removed as they would incorrectly train the DL model to detect some rocks as fish heads and tails. Additionally, there were many instances of seemingly very small fish labelled with heads and tails which were very hard to distinguish between in static images. However, upon viewing the moving video, swimming behaviours clearly indicated the direction fish were swimming in, which made head and tail identification easy to the human eye. Although there are published DL tracking algorithms (Bertinetto et al., 2016; Hu et al., 2022), YOLO-based methods only consider static images for training or inference. Combining tracking with head and tail detection will be the focus of future work so that numerous length measurements of the same fish can be made to calculate the average size, a method that is shown to be more statistically robust and less prone to measurement error (Harvey et al., 2001b). Validation experiments of measurements from stereo-BRUVS (Harvey and Shortis, 1995; Harvey et al., 2003; Harvey et al., 2010) have been conducted using three or more repeat measurements of fish. However, this is seldom done when conducting field surveys due to the extra labour required.

4.6 Combining optical and acoustic sampling methods

In recent years, size-spectrum models derived from acoustic surveys have emerged as essential tools for fish stock assessment and ecosystem-based fisheries management. Acoustic surveys possess the advantage of rapidly and efficiently covering vast spatial scales. However, stationary video platforms, such as stereo-BRUVS, are constrained by a limited field of view and can only monitor a small area around the camera. Acoustic surveys also face challenges, including difficulties in discriminating between fish species and detecting fish close to the seabed or within dense schools.

Size and shape information of fish targets is extracted from echo data by adjusting model parameters, such as growth rates, mortality rates, and species-specific traits, to match observed data (Edwards et al., 2017; Froese et al., 2019). Calibration and validation of these models often necessitate biological samples, which are invasive due to the physical capture and potential harm to fish during the process.

Assessing fishery resources in reef ecosystems, where obtaining biological samples is sometimes prohibited, remains challenging. To address these limitations, optic-acoustic methods combine video footage and acoustic measurements (Ryan and Kloser, 2016; Demer et al., 2020). Underwater cameras or video systems, either mounted on a research vessel, towed platform, or remotely operated vehicle (ROV), capture images or footage of fish, providing high-resolution information on size, shape, colour, and behaviour, which aids in species identification and refining size distribution estimates without the need for biological samples.

The automated length measurement of fish in stereo-videos using the method described in this study could be integrated with the optic-acoustic approach to capitalise on the strengths of both methods. Combining acoustic surveys with stereo-BRUVS, such as the preliminary work by Landero-Figueroa et al. (2016), or other sampling techniques can help overcome the limitations of each method and provide more accurate and comprehensive information on fish populations for stock assessment and ecosystem-based fisheries management. This non-invasive approach enables continuous monitoring of fish populations without harming the organisms or their habitats, offering a promising alternative for sustainable fishery management.

5 Conclusion

The semi-automated length measurement method presented here builds on and advances previously published DL-based fish detection from stereo-BRUVS imagery (Marrable et al., 2022). This new method combines that fish detection approach to isolate and crop individual fish from a busy scene with a new DL model for detecting the head and tail and applying photogrammetry to determine fish length measurements.

Although not completely autonomous, the machine-assisted, semi-automated labelling approach solves both the object correspondence challenge and allows for expert contextual knowledge to choose which fish (and in which pose) are sent for analysis using DL. This is expected to significantly reduce labour and analysis time by speeding up the manual process of precisely locating the head and tail of the fish in both images by carefully placing four mouse clicks on the screen, to two fast clicks anywhere on a fish while still using expert knowledge to truth and validate the result. By accelerating stereo-BRUVS analysis, more imagery can be processed; thereby, increasing the amount of data available for environmental reporting and decision making.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/open-AIMS/ozfish.

Author contributions

DM, MW, ST, and SB contributed to the development of the study design. DM, KB, ST, EH, MW, MS, and SLB contributed to the writing of the manuscript. All authors contributed to the article and approved the submitted version.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Andrialovanirina, N., Ponton, D., Behivoke, F., Mahafina, J., and Léopold, M. (2020). A powerful method for measuring fish size of small-scale fishery catches using image. *J.Fish. Res.* 223, 105425. doi: 10.1016/j.fishres.2019.105425

Atienza-Vanacloig, V., Andreu-García, G., López-García, F., Valiente-González, J. M., and Puig-Pons, V. (2016). Vision-based discrimination of tuna individuals in grow-out cages through a fish bending model. *Comput. Electron. Agric.* 130, 142–150. doi: 10.1016/j.compag.2016.10.009

Australian Institute Of Marine Science (2020). Ozfish dataset - machine learning dataset for baited remote underwater video stations. doi: 10.25845/5E28F062C5097

Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., and Torr, P. H. S. (2016). "Fully-convolutional siamese networks for object tracking," in *Computer vision – ECCV 2016 workshops* (Amsterdam, The Netherlands: Springer International Publishing), 850–865.

Boutros, N., Shortis, M. R., and Harvey, E. S. (2015). A comparison of calibration methods and system configurations of underwater stereo-video systems for applications in marine ecology. *Limnol. Oceanogr. Methods* 13, 224–236. doi: 10.1002/lom3.10020

Bradley, D., Merrifield, M., Miller, K. M., Lomonico, S., Wilson, J. R., and Gleason, M. G. (2019). Opportunities to improve fisheries management through innovative technology and advanced data systems. *Fish Fish* 20, 564–583. doi: 10.1111/faf.12361

Brock, V. E. (1954). A preliminary report on a method of estimating reef fish populations. *J. Wildl. Manage.* 18, 297–308. doi: 10.2307/3797016

Cappo, M. A., Harvey, E., Malcolm, H., and Speare, P. (2003). Potential of video techniques to monitor diversity, abundance and size of fish in studies of marine protected areas. *Aquat. Protected Areas-what works Best how do we know* 1, 455–464.

Cappo, M., Speare, P., Wassenberg, T., Harvey, E., Rees, M., Heyward, A., et al. (2001). Direct sensing of the size frequency and abundance of targt and non-target fauna in Australian fisheries-a national workshop. Pages 63–71.

Connolly, R. M., Fairclough, D. V., Jinks, E. L., Ditria, E. M., Jackson, G., Lopez-Marcano, S., et al. (2021). Improved accuracy for automated counting of a fish in baited underwater videos for stock assessment. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.658135

Cresswell, I., Janke, T., and Johnston, E. (2022). *Australia State of the environment 2021: overview* (Australia: Department of Agriculture, Water and the Environment).

Demer, D., Michaels, W., Cambronero Solano, S., Paramo, J., and Roa, C. (2020). *Integrated optic-acoustic studies of reef fish: report of the 2018 GCFI field study and workshop. NOAA Technical Memorandum NMFS-F/SPO-209* (Washington, DC: National Oceanic and Atmospheric Administration).

Dowling, N. A., Wilson, J. R., Rudd, M. B., Babcock, E. A., Caillaux, M., Cope, J., et al. (2016). FishPath: a decision support system for assessing and managing data- and capacity- limited fisheries Dowling2016-ww. doi: 10.4027/amdlfs.2016.03

Duarte, C. M., Agusti, S., Barbier, E., Britten, G. L., Castilla, J. C., Gattuso, J.-P., et al. (2020). Rebuilding marine life. *Nature* 580, 39–51. doi: 10.1038/s41586-020-2146-7

Edwards, A. M., Robinson, J. P. W., Plank, M. J., Baum, J. K., and Blanchard, J. L. (2017). Testing and recommending methods for fitting size spectra to data. *Methods Ecol. Evol.* 8, 57–67. doi: 10.1111/2041-210X.12641

Ellis, D. M., and DeMartini, E. E. (1995). Evaluation of a video camera technique for indexing abundances of juvenile pink snapper, pristipomoides filamentosus, and other hawaiian insular shelf fishes. *Oceanographic Literature Rev.* 9, 786.

French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., et al. (2019). Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES J. Mar. Sci.* 77, 1340–1353. doi: 10.1093/icesjms/fsz149

Friedlander, A. M., and DeMartini, E. E. (2002). Contrasts in density, size, and biomass of reef fishes between the northwestern and the main hawaiian islands: the effects of fishing down apex predators. *Mar. Ecol. Prog. Ser.* 230, 253–264. doi: 10.3354/meps230253

Froese, R., Winker, H., Coro, G., Demirel, N., Tsikliras, A. C., Dimarchopoulou, D., et al. (2019). On the pile-up effect and priors for linf and M/K: response to a comment by hordyk et al. on "a new approach for estimating stock status from length frequency data". *ICES J. Mar. Sci.* 76, 461–465. doi: 10.1093/icesjms/fsy199

Garcia-d'Urso, N., Galan-Cuenca, A., Pérez-Sánchez, P., Climent-Pérez, P., Fuster-?Guillo, A., Azorin-Lopez, J., et al. (2022). The DeepFish computer vision dataset for fish instance segmentation, classification, and size estimation. *Sci. Data* 9, 1–7. doi: 10.1038/s41597-022-01416-0

Goetze, J. S., Bond, T., McLean, D. L., Saunders, B. J., Langlois, T. J., Lindfield, S., et al. (2019). A field and video analysis guide for diver operated stereo-video. *Methods Ecol. Evol.* 10 (7), 1083–1090. doi: 10.1111/2041-210X.13189

Harvey, E. S., Cappo, M., Butler, J. J., Hall, N., and Kendrick, G. A. (2007). Bait attraction affects the performance of remote underwater video stations in assessment of demersal fish community structure. *Mar. Ecol. Prog. Ser.* 350, 245–254. doi: 10.3354/meps07192

Harvey, E., Cappo, M., Shortis, M., Robson, S., Buchanan, J., and Speare, P. (2003). The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (thunnus maccoyii) with a stereo–video camera system. *Fish. Res.* 63 (3), 315–326 doi: 10.1016/S0165-7836(03)00080-8

Harvey, E., Fletcher, D., and Shortis, M. (2001a). A comparison of the precision and accuracy of estimates of reef-fish lengths determined visually by divers with estimates produced by a stereo-video system. *Fish. Bull.* 99, 63.

Harvey, E., Fletcher, D., and Shortis, M. (2001b). Improving the statistical power of length estimates of reef fish: a comparison of estimates determined visually by divers with estimates produced by a stereo-video system. *Fishery bulletin-national oceanic atmospheric administration* 99, 72–80.

Harvey, E., Fletcher, D., and Shortis, M. (2002). Estimation of reef fish length by divers and by stereo-video. a first comparison of the accuracy and precision in the field on living fish under operational conditions. *Fish. Res.* 57, 255–265. doi: 10.1016/S0165-7836(01)00356-3

Harvey, E., Goetze, J., McLaren, B., Langlois, T., and Shortis, M. (2010). Influence of range, angle of view, image resolution and image compression on underwater stereo-video measurements: high-definition and broadcast-resolution video cameras compared. *Mar. Technol. Soc J.* 44, 75–85. doi: 10.4031/MTSJ.44.1.3

Harvey, E. S., McLean, D. L., Goetze, J. S., Saunders, B. J., Langlois, T. J., Monk, J., et al. (2021). The BRUVs workshop – an australia-wide synthesis of baited remote underwater video data to answer broad-scale ecological questions about fish, sharks and rays. *Mar. Policy* 127, 104430. doi: 10.1016/j.marpol.2021.104430

Harvey, E., and Shortis, M. (1995). A system for stereo-video measurement of sub-tidal organisms. *Mar. Technol. Soc J.* 29, 10–22.

Harvey, E. S., and Shortis, M. R. (1998). Calibration stability of an underwater stereo video system: implications for measurement accuracy and precision. *Mar. Technol. Soc. J.* 32 (2), 3–17.

Hellmrich, L. S., Saunders, B. J., Parker, J. R. C., Goetze, J. S., and Harvey, E. S. (2023). Stereo-ROV surveys of tropical reef fishes are comparable to stereo-DOVs with reduced behavioural biases. *Estuar. Coast. Shelf Sci.* 281, 108210. doi: 10.1016/j.ecss.2022.108210

Hu, W., Wang, Q., Zhang, L., Bertinetto, L., and Torr, P. H. S. (2022). SiamMask: a? framework for fast online object tracking and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45 (3), 3072–3089. doi: 10.1109/TPAMI.2022.3172932

Jennings, S., and Kaiser, M. J. (1998). "The effects of fishing on marine ecosystems," in *Advances in marine biology*, vol. 34 . Eds. J. H. S. Blaxter, A. J. Southward and P. A. Tyler (Academic Press), 201–352.

Jennings, S., and Polunin, N. V. C. (1997). Impacts of predator depletion by fishing on the biomass and diversity of non-target reef fish communities. *Coral Reefs* 16, 71–82. doi: 10.1007/s003380050061

Jessop, S. A., Saunders, B. J., Goetze, J. S., and Harvey, E. S. (2022). A comparison of underwater visual census, baited, diver operated and remotely operated stereo-video for sampling shallow water reef fishes. *Estuar. Coast. Shelf Sci.* 276, 108017. doi: 10.1016/j.ecss.2022.108017

Johansson, C., Stowar, M., and Cappo, M. (2008). The use of stereo BRUVS for measuring fish size. *Marine and Tropical Sciences Research Facility Report Series*; (Cape Cleveland, Australia: Australian Institute of Marine Science).

Landero-Figueroa, M. M., Parnum, I., Saunders, B. J., and Parsons, M. (2016). *Integrating echo-sounder and underwater video data for demersal fish assessment* (Brisbane, Australia: Acoustics).

Langlois, T., Goetze, J., Bond, T., Monk, J., Abesamis, R. A., Asher, J., et al. (2020). A field and video annotation guide for baited remote underwater stereo-video surveys of demersal fish assemblages. *Methods Ecol. Evol.* 11, 1401–1409. doi: 10.1111/2041-210X.13470

Lezama-Cervantes, C., Godínez-Domínguez, E., Gómez-Morales, H., Ornelas-Luna, R., Morales-Blake, A. R., Patiño-Barragán, M., et al. (2017). A suitable ichthyometer for systemic application. *Lat. Am. J. Aquat. Res.* 45, 870–878. doi: 10.3856/vol45-issue5-fulltext-1

Li, S., Zhang, Z., Li, B., and Li, C. (2018). Multiscale rotated bounding box-based deep learning method for detecting ship targets in remote sensing images. *Sensors* 18. doi: 10.3390/s18082702

Little, R., and Hill, N. (2021). 2021 state of the environment report marine chapter – expert assessment – management effectiveness – commercial fishing. doi: 10.26198/WWR3-4D52

Lopez-Marcano, S., Jinks, E. L., Buelow, C. A., Brown, C. J., Wang, D., Kusy, B., et al. (2021). Automatic detection of fish and tracking of movement for ecology. *Ecol. Evol.* 11, 8254–8263. doi: 10.1002/ece3.7656

MacNeil, M. A., Chapman, D. D., Heupel, M., Simpfendorfer, C. A., Heithaus, M., Meekan, M., et al. (2020). Global status and conservation potential of reef sharks. *Nature* 583, 801–806. doi: 10.1038/s41586-020-2519-y

Marrable, D., Barker, K., Tippaya, S., Wyatt, M., Bainbridge, S., Stowar, M., et al. (2022). Accelerating species recognition and labelling of fish from underwater video with machine-assisted deep learning. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.944582Marrable2022

McClanahan, T. R., Muthiga, N. A., Kamukuru, A. T., Machano, H., and Kiambo, R. W. (1999). The effects of marine parks and fishing on coral reefs of northern tanzania. *Biol. Conserv.* 89, 161–182. doi: 10.1016/S0006-3207(98)00123-2

Melnychuk, M. C., Kurota, H., Mace, P. M., Pons, M., Minto, C., Osio, G. C., et al. (2021). Identifying management actions that promote sustainable fisheries. *Nat. Sustainability* 4, 440–449. doi: 10.1038/s41893-020-00668-1

Miranda, J. M., and Romero, M. (2017). A prototype to measure rainbow trout's length using image processing. *Aquacult. Eng.* 76, 41–49. doi: 10.1016/j.aquaeng.2017.01.003

Monkman, G. G., Hyder, K., Kaiser, M. J., and Vidal, F. P. (2019). Using machine vision to estimate fish length from images using regional convolutional neural networks. *Methods Ecol. Evol.* 10, 2045–2056. doi: 10.1111/2041-210X.13282

Needle, C. L., Dinsdale, R., Buch, T. B., Catarino, R. M., Drewery, J., Butler, N., et al. (2015). Scottish Science applications of remote electronic monitoring. *ICES J. Mar. Sci.* 72 (4), 1214–1229. doi: 10.1093/icesjms/fsu225

Pauly, D., Christensen, V., Guénette, S., Pitcher, T. J., Sumaila, U. R., Walters, C. J., et al. (2002). Towards sustainability in world fisheries. *Nature* 418, 689–695. doi: 10.1038/nature01017

Pauly, D., and Morgan, G. R. (1987). Length-based methods in fisheries research: WorldFish, 299.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 779–788.

Roberts, C. M. (1995). Rapid build-up of fish biomass in a caribbean marine reserve. *Conserv. Biol.* 9, 815–826. doi: 10.1046/j.1523-1739.1995.09040815.x

Rochet, M.-J., and Trenkel, V. M. (2003). Which community indicators can measure the impact of fishing? a review and proposals. *Can. J. Fish. Aquat. Sci.* 60, 86–99. doi: 10.1139/f02-164

Rudd, M. B., and Thorson, J. T. (2018). Accounting for variable recruitment and fishing mortality in length-based stock assessments for data-limited fisheries. *Can. J. Fish. Aquat. Sci.* 75, 1019–1035. doi: 10.1139/cjfas-2017-0143

Ryan, T. E., and Kloser, R. J. (2016). Improved estimates of orange roughy biomass using an acoustic-optical system in commercial trawlnets. *ICES J. Mar. Sci.* 73, 2112–2124. doi: 10.1093/icesjms/fsw009

Schramm, K. D., Marnane, M. J., Elsdon, T. S., Jones, C., Saunders, B. J., Goetze, J. S., et al. (2020). A comparison of stereo-BRUVs and stereo-ROV techniques for sampling shallow water fish communities on and off pipelines. *Mar. Environ. Res.* 162, 105198. doi: 10.1016/j.marenvres.2020.105198

Seguin, R., Mouillot, D., Cinner, J. E., Stuart Smith, R. D., Maire, E., Graham, N. A. J., et al. (2022). Towards process-oriented management of tropical reefs in the anthropocene. *Nat. Sustain.* 6 (2), 148–157. doi: 10.1038/s41893-022-00981-x

Shafait, F., Harvey, E. S., Shortis, M. R., Mian, A., et al. (2017). Towards automating underwater measurement of fish length: a comparison of semi-automatic and manual stereo–video measurements. *ICES J. Mar. Sci.* 74 (6), 1690–1701. doi: 10.1093/icesjms/fsx007

Shi, C., Wang, Q., He, X., Zhang, X., and Li, D. (2020). An automatic method of fish length estimation using underwater stereo system based on LabVIEW. *Comput. Electron. Agric.* 173, 105419. doi: 10.1016/j.compag.2020.105419

Shortis, M. (2015). Calibration techniques for accurate measurements by underwater camera systems. *Sensors* 15, 30810–30826. doi: 10.3390/s151229831

Steneck, R. S., and Pauly, D. (2019). Fishing through the anthropocene. *Curr. Biol.* 29, R987–R992. doi: 10.1016/j.cub.2019.07.081

Suo, F., Huang, K., Ling, G., Li, Y., and Xiang, J. (2020). "Fish keypoints detection for ecology monitoring based on underwater visual intelligence," in *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. 542–547. doi: 10.1109/ICARCV50220.2020.9305424

Tonachella, N., Martini, A., Martinoli, M., Pulcini, D., Romano, A., and Capoccioni, F. (2022). An affordable and easy-to-use tool for automatic fish length and weight estimation in mariculture. *Sci. Rep.* 12, 15642. doi: 10.1038/s41598-022-19932-9

Upton, K. R., and Riley, L. G. (2013). Acute stress inhibits food intake and alters ghrelin signaling in the brain of tilapia (oreochromis mossambicus). *Domest. Anim. Endocrinol.* 44, 157–164. doi: 10.1016/j.domaniend.2012.10.001

Uranga, J., Arrizabalaga, H., Boyra, G., Hernandez, M. C., Goñi, N., Arregui, I., et al. (2017). Detecting the presence-absence of bluefin tuna by automated analysis of medium-range sonars on fishing vessels. *PloS One* 12, e0171382. doi: 10.1371/journal.pone.0171382

White, D. J., Svellingen, C., and Strachan, N. J. C. (2006). Automated measurement of species and length of fish by computer vision. *Fish. Res.* 80, 203–210. doi: 10.1016/j.fishres.2006.04.009

Whitmarsh, S. K., Fairweather, P. G., and Huveneers, C. (2017). What is big BRUVver up to? methods and uses of baited underwater video. *Rev. Fish Biol. Fish.* 27, 53–73. doi: 10.1007/s11160-016-9450-1

Wibisono, E., Mous, P., Firmana, E., and Humphries, A. (2022). A crew-operated data recording system for length-based stock assessment of indonesia's deep demersal fisheries. *PloS One* 17, e0263646. doi: 10.1371/journal.pone.0263646

Wilson, S. K., Graham, N. A. J., Holmes, T. H., MacNeil, M. A., and Ryan, N. M. (2018). Visual versus video methods for estimating reef fish biomass. *Ecol. Indic.* 85, 146–152. doi: 10.1016/j.ecolind.2017.10.038

Yang, L., Liu, Y., Yu, H., Fang, X., Song, L., Li, D., et al. (2021). Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: a review. *Arch. Comput. Methods Eng.* 28, 2785–2816. doi: 10.1007/s11831-020-09486-2

Zhao, S., Zhang, S., Liu, J., Wang, H., Zhu, J., Li, D., et al. (2021). Application of machine learning in intelligent fish aquaculture: a review. *Aquaculture* 540, 736724. doi: 10.1016/j.aquaculture.2021.736724

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 05 June 2023
DOI 10.3389/fmars.2023.1162064

Check for updates

OPEN ACCESS

EDITED BY
Mark C. Benfield,
Louisiana State University, United States

REVIEWED BY
Philippe Blondel,
University of Bath, United Kingdom
Peng Ren,
China University of Petroleum, China
Ning Wang,
Dalian Maritime University, China

*CORRESPONDENCE
Jianfeng Tong
jftong@shou.edu.cn

RECEIVED 09 February 2023
ACCEPTED 22 May 2023
PUBLISHED 05 June 2023

CITATION
Tong J, Wang W, Xue M, Zhu Z, Han J and Tian S (2023) Automatic single fish detection with a commercial echosounder using YOLO v5 and its application for echosounder calibration.
Front. Mar. Sci. 10:1162064.
doi: 10.3389/fmars.2023.1162064

COPYRIGHT
© 2023 Tong, Wang, Xue, Zhu, Han and Tian. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Automatic single fish detection with a commercial echosounder using YOLO v5 and its application for echosounder calibration

Jianfeng Tong[1,2,3]*, Weiqi Wang[1], Minghua Xue[1], Zhenhong Zhu[1], Jun Han[4] and Siqian Tian[1,3,5]

[1]College of Marine Sciences, Shanghai Ocean University, Shanghai, China, [2]Key Laboratory of Marine Ecological Monitoring and Restoration Technologies, Ministry of Natural Resources (MNR), Shanghai, China, [3]National Engineering Research Center for Oceanic Fisheries, Shanghai Ocean University, Shanghai, China, [4]School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney, NSW, Australia, [5]Key Laboratory of Sustainable Exploitation of Oceanic Fisheries Resources, Ministry of Education, Shanghai, China

Nowadays, most fishing vessels are equipped with high-resolution commercial echo sounders. However, many instruments cannot be calibrated and missing data occur frequently. These problems impede the collection of acoustic data by commercial fishing vessels, which are necessary for species classification and stock assessment. In this study, an automatic detection and classification model for echo traces of the Pacific saury (*Cololabis saira*) was trained based on the algorithm YOLO v5m. The *in situ* measurement value of the Pacific saury was measured using single fish echo trace. Rapid calibration of the commercial echo sounder was achieved based on the living fish calibration method. According to the results, the maximum precision, recall, and average precision values of the trained model were 0.79, 0.68, and 0.71, respectively. The maximum F1 score of the model was 0.66 at a confidence level of 0.454. The living fish calibration offset values obtained at two sites in the field were 116.30 dB and 118.19 dB. The sphere calibration offset value obtained in the laboratory using the standard sphere method was 117.65 dB. The differences between *in situ* and laboratory calibrations were 1.35 dB and 0.54 dB, both of which were within the normal range.

KEYWORDS

fishing vessel, automatic detection, commercial echosounder calibration, *Cololabis saira*, deep learning, single fish detection

1 Introduction

As an important method for fishery resource surveys, hydroacoustic technology enables fast and independent testing, that is both harmless for the resources and accurate. Moreover, underwater acoustic spatial information with time series can be obtained (Foote and Rothschild, 2009; Haris et al., 2021). Hydroacoustic detection

technology plays an important role in analyzing fish migration paths (Martignac et al., 2015; Gjøsæter et al., 2017), fish habitat distribution (Slotte et al., 2004; O'Donncha et al., 2021), and fish resource changes (Melvin et al., 2016; Aranis et al., 2022). It also allows to study zooplankton sound scattering layers (Boswell et al., 2020; Xue et al., 2021). The acoustic characteristics of a single organism or a biotic aggregate are defined as echo traces (Reid, 2000). The SHAPES theory (Coetzee, 2000) was published to provide a method of analyzing fish populations based on these echo traces. The main parameters the theory uses are the morphology and echo strength distribution of echo traces. Based on the above theory of fish echo trace analysis, the distributions of adult and juvenile sardine aggregation were found to be significantly different in the Mediterranean region (Tsagarakis et al., 2012). Swarms of anchovy (*Engraulis ringens*), common sardine (*Sardinops sagax*), and Pacific jack mackerel (*Trachurus symmetricus*) were identified by the SHAPES theory in northern and south-central Chilean waters. This analysis innovatively uses a statistical model to automate the classification of large quantities of fish echo traces (Robotham et al., 2010). The above studies demonstrate the feasibility of distinguishing species and age groups by features of fish school echo traces. However, the echo traces that emerge in response to discrete single fish situated around the school were often ignored. The echo traces of discrete individuals are usually inverted 'V'-shaped or lightning-shaped (Reid, 2000). In previous studies (Boyra et al., 2019; Julie et al., 2020; Khodabandeloo et al., 2021), single fish echo traces were the main data source for measuring the *in situ* target strength values of different fish species. These single fish echo traces are important for fish species classification. Different fish species (Sawada et al., 2009), swimming tilt angles (Fernandes et al., 2016; Tong et al., 2022), swimming speeds (Lee et al., 2010), and fish swim bladder sizes (Sobradillo et al., 2019) affect the magnitude of fish target strength values. Because of dense fish aggregation during fishing activities, there are numerous targets on the echogram, making the detection and extraction of single fish echo images more challenging because of interference of environmental and instrument noises. Thus, most current *in situ* target strength measurement applications still require rigorous equipment and environmental conditions, while having limited application scope for measured target strength values.

Previous studies predicted the categories of echo trace and large-scale automatic classification using the calculation power of computers. Initially, statistical models were used to classify morphological parameters of the acoustic image measurements and echo strength values (LeFeuvre et al., 2000). These models include supervised machine learning models, such as classification tree (Fernandes, 2009), random forest (Fallon et al., 2016), support vector machine (Robotham et al., 2010), as well as unsupervised machine learning models such as K-means (Ito et al., 2013), Gaussian mixture models (Robotham et al., 2010), and principal component analysis (Lawson et al., 2001). However, statistical models are dependent. Digital image processing techniques and related acoustic methods are required to capture and enhance the echo trace features and infer the variability between feature parameters to complete the automatic identification process. Basic hypotheses are established based on feature values and variability to guide model training, which increases the difficulty and time consumption of data processing.

Deep learning techniques have been employed to develop a number of available network frameworks (Wang et al., 2022; Wang et al., 2023a). These frameworks and the modules that are based on them have been widely applied for underwater image enhancement (Wang et al., 2023b; Wang et al., 2023c) and noise control (Wang et al., 2023d). Among them, convolutional neural network (CNN) is one of the more widely used network architectures. The advent of CNN has increased the freedom of machine self-learning (Rathi et al., 2017; Albawi et al., 2018; Gu et al., 2018) while providing more possibilities for the identification of fish echo traces. Currently, target detection algorithms based on CNN can be classified into two-stage algorithms represented by Faster R-CNN (Li et al., 2015) and one-stage algorithms represented by YOLO (You Only Look Once) (Jalal et al., 2020). The two-stage algorithms mainly include two stages of interest region extraction and image detection, and can achieve higher recognition accuracy than single-stage algorithms. The increased computational power obtained by the region of interest extraction stage also limits the speed with which the algorithm can detect the target. Compared with a two-stage algorithm, the YOLO algorithm-based single-stage algorithm implements target detection and bounding box regression operations directly on the image, thus achieving a higher target detection speed. However, its recognition accuracy is slightly lower than that of the two-stage algorithm model. In a recent study, a deep learning-based target detection algorithm was applied to the target detection of underwater fish optical images. Li et al. (2015) and Li et al., (2016) captured underwater acoustic images and achieved recognition of fish in images by the faster R-CNN algorithm. Wageeh et al. (Wageeh et al., 2021) used a YOLO model with the introduction of an image enhancement algorithm to achieve automatic detection and counting of fish at a fish farm. Wang et al. (Wang et al., 2021) established a basic line for underwater object detection based on the YOLO v5 algorithm, which facilitated subsequent research on the detection of underwater objects. Jalal et al. (Jalal et al., 2020) proposed a method for detecting and identifying fish in complex underwater environments by combining a Gaussian mixture model, an optical flow module to detect the temporal information of fish swimming in the video, and a YOLO target recognition module to improve the comprehensive accuracy of video target detection. Acoustic images are usually captured in the form of one-channel graphing, which contains less information than optical images, usually containing three channels. This is challenging for acoustic image recognition using the YOLO model. The YOLO model is still valid for small target echo target recognition in the presence of noise in acoustic images (Fang and Wang, 2021).

In this study, the acoustic data collected by commercial Pacific saury (*Cololabis saira*) fishing vessels were used as original dataset to train the YOLO model. The pre-processing module of the acoustic data was established using image processing. Based on the YOLO v5 algorithm, the automatic target detection model was constructed to complete the automatic detection and target identification of single fish and fish schools in the echograms.

Finally, echo traces extracted from the target recognition were used to identify single fish and calibrate the echosounder of the commercial fishing vessel.

2 Materials and methods

2.1 Acoustic data collection

The fishing platform is the ocean-going Pacific saury fishing vessel FV 'Ming Hua,' with a total length of 73.98 m and a draft of 5 m. The vessel entered the fishing grounds on May 13, 2021, and carried out the fishing of Pacific saury and squid (*Todarodes pacificus*). In this study, the data collected at the time of catching Pacific saury were used as the original dataset. The main area of the Pacific saury is the high seas region of the northwest Pacific Ocean (41°–48° N, 166°–172° E) (Figure 1), using a stick-held dipnet for fishing. The acoustic instrument used for acoustic data collection was a Hondex HE-1500Di (The Honda Electronics Co., Ltd., Toyohashi, Japan) single-beam commercial echo sounder. The basic parameters of the echo sounder are shown in Table 1. The commercial echo sounder was modified to save the raw echo level data collected by the transducer directly and combine it with both GPS data and time series. Then, the data were stored on a flash memory card. The detecting depth of the echo sounder was 300 m, and each memory card could collect 6.8 h of acoustic data.

2.2 Processing algorithm

2.2.1 Acoustic data pre-processing

The acoustic echograms obtained from the original acoustic dataset contain electromagnetic pulse noise from other fishing vessel equipment, environmental noise, and zooplankton reverberation. These noises can be a great obstacle for the identification and labeling of fish schools and single fish, as well as a challenge for learning single fish and fish school features during the model training process. In this study, an acoustic data pre-processing algorithm is proposed based on digital image processing technology to remove both noise and reverberation. The algorithm flow is shown in Figure 2.

The echo level value in the acoustic data was first converted to sound backscattering strength values. The conversion formula is shown in Equation (1):

$$Sv = EL + 20\log(r) + 2\alpha r - 10\log\left(\varphi \times \frac{c\tau}{2}\right) - K_0 \qquad (1)$$

where EL is the received echo level (dB re 1 μV); α is the sound absorption coefficient; r is the depth value; φ is the equivalent beam angle; c is the sound speed in water; and τ is the pulse length. K_0 is a transmitting and receiving factor, which is determined by the sphere calibration (dB) according to Equation (7). The data within 5 m of the sea surface of the acoustic data were removed according to the draft depth of the fishing vessel to avoid interference of the data by bubbles generated by the vessel and the movement of the surf. The integration threshold range of the acoustic data is set, and the part outside the integration threshold is removed to avoid the disturbance of the echo data by zooplankton and large predators. The integration threshold was set to range from − 20 dB to 64 dB according to the integration settings in previous small pelagic fish resource surveys (Axenrot et al., 2004; Trumpickas et al., 2020). The small discrete noise generated by bubbles and the high-frequency impulse noise caused by instruments were removed using the open-close operation and the 3*3 median filter, respectively. The edge detection algorithm was used to detect the edge of the echo trace. The morphology, depth, and scattering strength of the echo trace are measured using the regionprops function. To prepare the echogram data for the

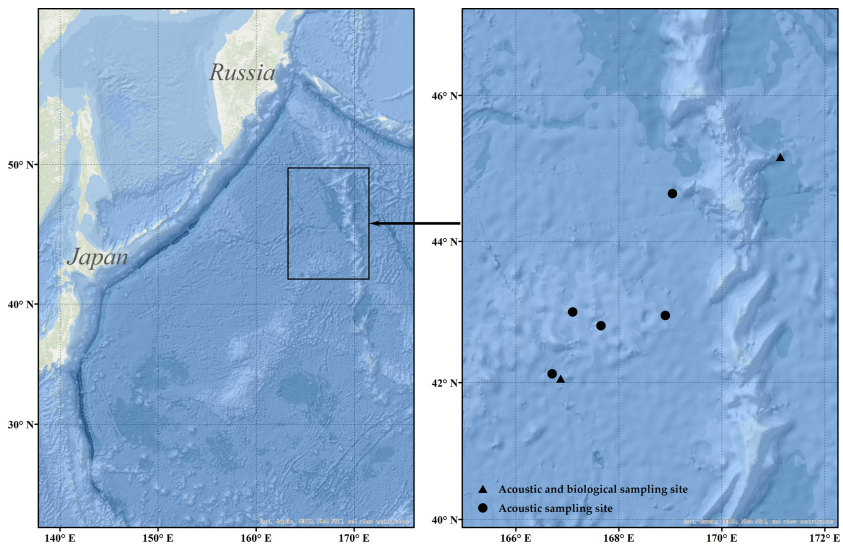


FIGURE 1
The black frame in the left figure panel indicates the range of acoustic monitoring; the black dots in the right figure panel indicate acoustic monitoring data sampling sites; the black triangles indicate acoustic and biological sampling sites.

TABLE 1 Main parameters of the Hondex HE-1500Di echo sounder.

Parameters	Values
Frequency	50 kHz
Transducer type	TD-47
Beam type	Single beam
Pulse length (ms)	1.7
Pulse interval (s)	1
Transmit power (w)	1000
Absorption coefficient (dB/m)	0.0129
Equivalent beam angle (dB re 1 Str)	-13.79

process of target detection by YOLO v5, the grayscale image was transformed by the first-order numerical matrix. Each value in the matrix is mapped to the set colormap, the colors in the colormap are all RGB colors, and each color is a double float value in the interval [0,1]. The data of the matrix is normalized to correspond with the color value, and different values represent different colors. Thus, the indexed image using RGB color is formed. At this stage, the acoustic data preprocessing is complete. The specific process of pre-processing is detailed in Appendix A.

2.2.2 Echo trace classification and labeling

After pre-processing and morphological measurements, the echo traces that remained on the 50-kHz echograms were filtered. The location of the Pacific saury school was approximately determined by comparing the time of each catch in the fishing logbook for further filtering. The method of determining whether an echo trace is a single fish by analyzing the echo trace height related to pulse length has been applied to *in situ* target strength measurements (Didrikas and Hansson, 2004; Sawada et al., 1993). In this study, the above method was used to filter and separate single fish echo traces. The fish school was filtered with reference to the SHAPES algorithm (Coetzee, 2000). Echo traces with a height larger than 1 m and a length longer than 5 m were classified as fish schools. The remaining echo traces were classified as multiple fish. The three types of echo traces were labeled as "0" for single fish, "1" for multiple fish, and "2" for fish schools.

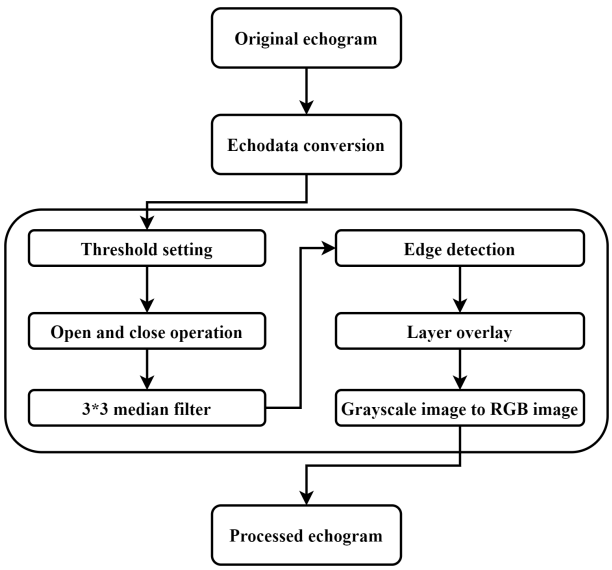


FIGURE 2
The architecture of the developed acoustic data pre-processing algorithm.

2.3 YOLO v5 model

2.3.1 Model structure

The YOLO v5 model is one of the representative models of one-stage target detection models based on deep learning. The four main versions in the existing YOLO v5 series are named YOLO v5s, YOLO v5m, YOLO v5l, and YOLO v5x. The differences between these four versions are the depth and width of the model network. Different network depths determine the number of convolutional layers, and different network widths determine the number of convolutional kernels in one convolutional layer. The network depth and width of these four versions of the model increase sequentially. An increase in the number of convolutional kernels and convolutional layers represents an enhancement in the recognition accuracy of the model, but also increases the size of model. To run the model on devices with low computing power while ensuring the detection accuracy, YOLO v5m was used as the base training model for the automatic detection experiments. YOLO v5m has a smaller model complexity compared to YOLO v5l and YOLO v5x, thus enabling model training on lower-computing devices. YOLO v5m also has a better small target detection capability compared to YOLO v5s. The main network structure of the model is shown in Figure 3. Its structure consists of four parts: Input, Backbone, Neck, and Prediction.

The size of the imported RGB images in three channels set at the input side was 640 by 640 pixels When importing the images from the dataset into the model for training, the model automatically scaled the image size to the set size using the adaptive image scaling module. The Mosaic data enhancement algorithm and adaptive anchor frame calculation method were used at the input side to enhance the generalization ability of the model.

The backbone network part of the model mainly includes the four modules of focus, CBL, CSP, and SPPF. Among them, the focus module is used for downsampling, slicing, and convolution. Adjacent pixels in the image were first sampled using the down sampling and slicing method. After this operation, an image was divided into four feature maps, thus the number of channels is expanded four times without loss of information, and the size of the obtained feature maps was 320*320*12. Then, the image was convoluted by using convolutional kernel, and the final feature maps were also 320*320*32. Compared with common down sampling, the focus module completes image down sampling without loss of information. The CBL module contains convolution (conv), batch normalization (BN), and Leaky Relu, which serve to convolve the input data. The CSP module contains the CBL module and its components, with the addition of a residual

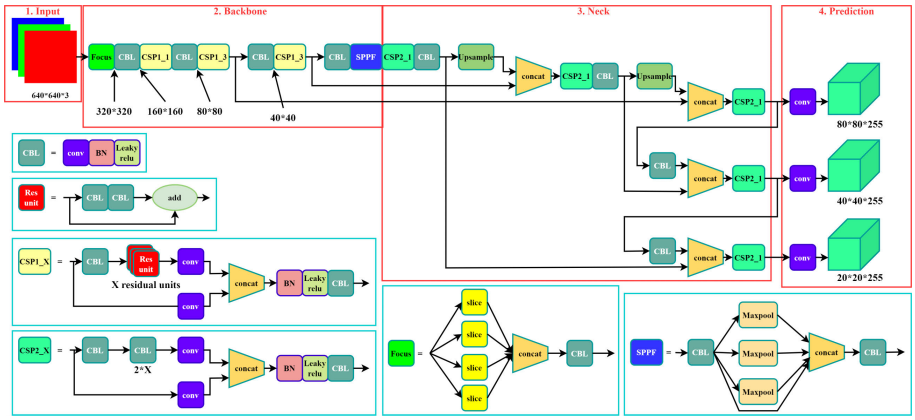


FIGURE 3
The main architecture of the YOLO v5m model.

component to avoid network degradation caused by gradient disappearance. The CSP module enables the model to learn more features. The SPPF module converts feature maps of arbitrary size into feature vectors of fixed size *via* the CBL module and maxpooling. The image was sliced and convolved into a 320*320*32 feature map by the focus module, convolved, and the residual features of the image were extracted by the CBL module. The number of network channels was expended through the SPPF module after earning the residual image features with the CSP module.

2.3.2 Model training

Model training was conducted using an Intel (R) Core (TM) i7-10875H CPU @ 2.30 GHz, GPU selected NVIDIA Geforce GTX1650 with 4 GB of video memory, using PyTorch 1.13 as the deep learning framework. The number of epochs was set to 300 in model training, and the batch size was set to 16.

2.3.3 Model evaluation indicators

Precision (P), recall (R), mean average precision (mAP), and F1-Score were used as indicators to evaluate the performance of the echo trace target detection model. P represents the precision and accuracy of the model, while R represents its recall and completeness. Formulas of P and R are shown in Equations (2) and (3):

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

where TP is truly positive, indicating that prediction and actual exist at the same time; FP is false positive, indicating that actually does not exist but prediction does; FN is a false negative, indicating that actual exists, but prediction does not. While mAP represents the average accuracy of all target categories detected by the model, the formula is obtained by averaging the average precision (AP) values of all targets. The F1-score represents the summed average of precision and recall with a maximum value of 1 and a minimum value of 0. This parameter allows for a more intuitive representation of the detection accuracy of the model. AP and mAP could be calculated using Equations (4) and (5):

$$AP = \sum_{i=1}^{n-1} (r_{i+1} - r_i) * P_{inter}(r_i + 1) \tag{4}$$

$$mAP = \frac{\sum_{i=1}^{k} AP_i}{k} \tag{5}$$

where $r_{i+1} - r_i r_i$ is the amount of change in recall and $P_{inter}(r_i + 1)$ is the precision of the interpolation segment when the recall is r_i. The F1-Score is calculated according to Equation (6):

$$F1_{score} = 2 * \frac{P * R}{P + R} \tag{6}$$

2.4 Living fish calibration for the commercial echo sounder

When acoustic surveys are conducted using fishing vessels, the lack of sufficient time for standard process instrument calibration of echo sounder indicates the need to evaluate instrument performance using a simplified method. In previous studies, certain calibration methods using objects with known physical properties have been used to calibrate the echosounder, including the calibration sphere method (Knudsen, 2009), the natural seafloor calibration method (Eleftherakis et al., 2018), and the living fish calibration method (Johannesson and Losse, 1977). Of these, the natural seafloor calibration method and the living fish calibration method (both relative calibration methods) can test the performance of the echo sounder within a short period, and are thus suitable for the calibration of acoustic instruments on commercial fishing vessels. The acoustic data collected in this study were not detected at the sea bottom because the area is located in the deep sea. Hence, the living fish calibration method was used for commercial echo sounder calibration.

The instrument calibration of the commercial echo sounder was performed in a laboratory pool before the fishing vessel was put to sea. The sphere calibration offset K_0 was obtained in a standard sphere calibration process. The formula for sphere calibration offset is shown in Equation (7):

$$K_0 = EL + 40\log(r) + 2\alpha r - TS \tag{7}$$

where r is the distance between the target and transducer; α is the hydroacoustic absorption coefficient; TS is the target strength of the calibration sphere; EL is the echo level (dB re 1μV) of the calibration sphere on the beam axis. The YOLO v5 model was used to detect single fish echo traces, and the max echo level values of the echo trace in the bounding box were extracted for calculating the on-axis measurement value (MV) (dB). MV is calculated according to Equation (8):

$$MV = EL + 40\log(r) + 2\alpha r - 2D \tag{8}$$

where D is the directivity of the transducer. This study used a single-beam transducer to measure the target echo level value. When the target is directly below the transducer, D is 0, and the target echo level value reaches the maximum at this time. The prolate spheroidal model (PSM) was used to simulate the target strength of the Pacific saury, and the catches caught during the acoustic monitoring were sampled to obtain 100 fish from two sampling sites. The total length and fork length of the Pacific saury were measured on board. The correlation coefficient A_{soft} was calculated based on the swim bladder fish model, as shown in Equation (9):

$$A_{soft} = 20\log\left(\frac{F}{2a}\right) + 20\log\left(\frac{L_b}{L}\right) - 40 \tag{9}$$

where F is defined as the absolute value of the backscattering amplitude from the fish in the far field region; a is half of the fork length; L_b is the length of the swim bladder, and L is the fork length of the fish. For the ratio of the length of the swim bladder to the fork length of the fish in Equation (9), a typical value of 0.34 is assumed based on the research of Furukawa (Furusawa, 1988). The TS_{model} is calculated based on Equation (9), as shown in Equation (10):

$$TS_{model} = A_{soft} + 20\log(L) \tag{10}$$

The living fish calibration offset K is obtained by subtracting the *in situ* MV from the TS_{model}, as shown in Equation (11):

$$K = MV - TS_{model} \tag{11}$$

3 Results

3.1 Pre-processing algorithm experiment

The raw acoustic data were collected over 7 d of fishing. During the catching process, the number of Pacific sauries in the total catch was highest, which shows that when fishing with the collector light, the fish that rise to the sea surface are mainly saury; furthermore, the fish that are attracted by the beam emitted by the transducer are saury. An example original acoustic echogram obtained during the fishing process is shown in Figure 4. The fish gradually concentrated within the water layer about 30 m from the sea surface when the fish trap light was turned on. The echo data within 30 m intercepted from Figure 4 are shown in Figure 5, where Figure 5A shows the fish underwater during the search process. Figures 5B, C show the underwater fish when the fish trap light is turned on. The fish gradually gathered in the water layer around 20 m and formed a dense cluster. Figure 5D shows the fish underwater during the fishing process. The fish were mainly concentrated in the water layer of 20–30 m depth, while the fish within 20 m were relatively discrete. Many bubbles and noise signals were generated by the fishing vessel in the above images, and reverberant signals were generated by plankton, which is the main prey of the Pacific saury.

The acoustic echogram after pre-processing using the algorithm and labeling is shown in Figure 6. The echograms of the echo trace of Pacific saury were separated, and the noise and reverberation generated by the plankton were removed. The depth and morphology of fish could be seen more clearly in the echograms. The isolated echo traces were boxed out using the red bounding box. The parameters obtained from the measurements were used to classify the echo trace as "0" for single fish, "1" for multiple fish, and "2" for schools. The labeled results are located in the upper left corner of the red bounding box.

3.2 Dataset construction

The pre-processed echograms are used as automatic recognition model training dataset. The duration of each pre-processed echogram was 30 min, and the depth of the echogram was 30 m. According to the size of imported images (640*640*3), the resolution in the vertical direction was 4.6 cm and the resolution in the horizontal direction was 2.81 sec. Because the speed of the fishing vessel was not constant during the fishing process, the horizontal resolution of each data is different. See Appendix B for details. A total of 91 echograms were finally available in the dataset. A total of 10,710 echo traces were extracted from the echograms, including 7,725 single-fish echo traces, 2,346 multiple-fish echo traces, and 639 echo traces of the school. The dataset was randomly divided into a training set (85%), a validation set (5%), and a test set (15%).

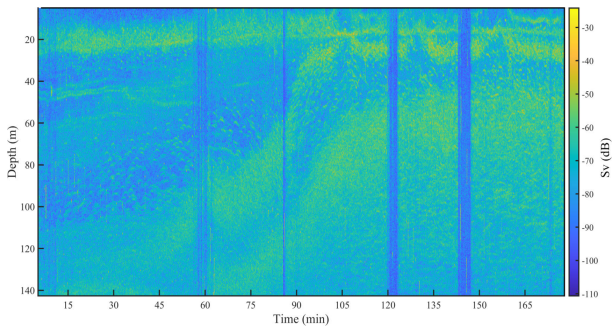


FIGURE 4
Example of an original acoustic echogram associated with Pacific saury during the search and catch period.

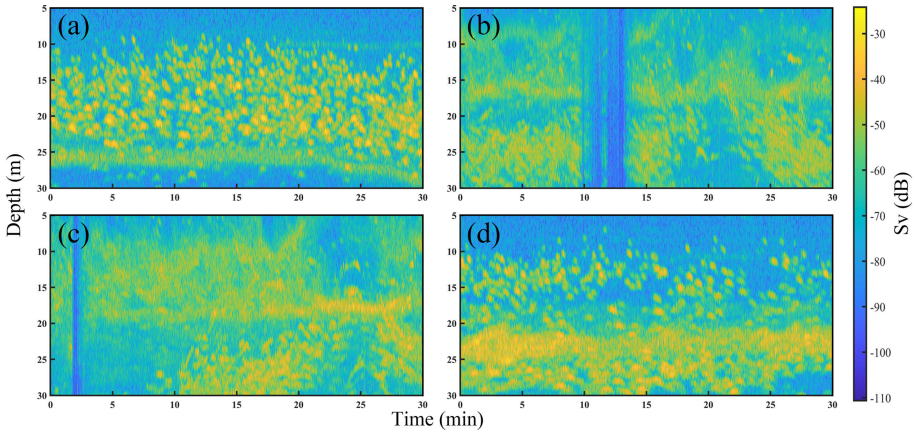


FIGURE 5
Acoustic echograms associated with Pacific saury in the surface layer (5–30 m) during the searching the catching periods. **(A)** The swarm during the searching process. **(B)** The swarm when the fish collector light was turned on. **(C)** The swarm after a period of illumination of the water surface by the collector light. **(D)** The swarm during the catching process.

3.3 Model training results

Table 2 presents the evaluation metrics of the observation used to test the effectiveness of the trained model. The recall of the model reached a maximum of 0.68 when the number of training epochs was 211. The precision and mAP_0.5 reached maximum values of 0.79 and 0.71, respectively, when the number of epochs was 281. The mAP_0.5:0.95 reached a maximum of 0.43 when the number of epochs was 300. The curve of the F1-score related to the confidence level is shown in Figure 7. The F1-score for all classes at a confidence level of 45.4% reached a maximum value of 0.66. At a confidence level of about 55%, the F1-score remained above 0.6, then decreased rapidly until it reached zero. The echograms from the test set were imported into the trained model. Detection results are shown in Figure 8.

3.4 Calibration of the commercial echo sounder

Figure 9 shows the *in situ MV* histograms for two sampling sites with biological sampling. The maximum value of the *in situ MV* observed on June 4, 2021, was 94.35 dB, and the minimum value was 54.93 dB. The maximum value observed on July 5 was 95.58 dB, and the minimum value was 55.13 dB. The difference between the two maximum values was 1.23 dB, and the difference between the two minimum values was 0.2 dB.

The average, standard deviation, maximum, and minimum values of the measured body lengths of the Pacific saury samples collected at the two stations are presented in Table 3. The histogram of the TS_{model} calculated from the measured body lengths is shown in Figure 10. The *in situ MV* and TS_{model} measurements are averaged and differenced to obtain the value of living fish calibration offset K. The calculated living fish calibration parameters are shown in Table 4.

The data in Table 4 show that the mean *in situ MV* measured on Jun 4, 2021, was 70.48 dB and 73.85 dB on Jul 5, 2021. The mean values of the correlation coefficients A_{soft} for the two sites measured by the model method were -95.29 and -92.47, respectively, and the calculated TS_{model} values were -45.82 dB and -44.34 dB, respectively. The living fish calibration offset K values were 116.30 dB and 118.19 dB, respectively. Compared with the K_0 measured from the standard sphere method calibration, the differences between K and K_0 were 1.35 dB and 0.54 dB, respectively.

4 Discussion

4.1 Automatic echo trace detection

In this study, no training set of echo traces was available to pre-train the model. Therefore, a training set was created to train the automatic detection model for subsequent automatic detection of echo traces. The training set was created using the integral threshold setting method, median filter, and open-close operation to remove noise and reverberation from images. In the actual experiment, the noise and reverberation that were present in the original echograms (Figure 5) were removed. At the same time, the echo traces of single fish, multiple fish, and schools of fish were retained more completely (Figure 6). The method used in this study is simpler than denoising using the dB difference method (Fernandes, 2009; Brautaset et al., 2020). The reason for its simplicity is the overwhelming dominance of Pacific saury in the detected echograms and the fact that the used instrument is a single-beam with a single-frequency echo sounder.

The adopted YOLO v5 deep learning automatic detection model has a maximum value of 0.71 for mAP at intersection over

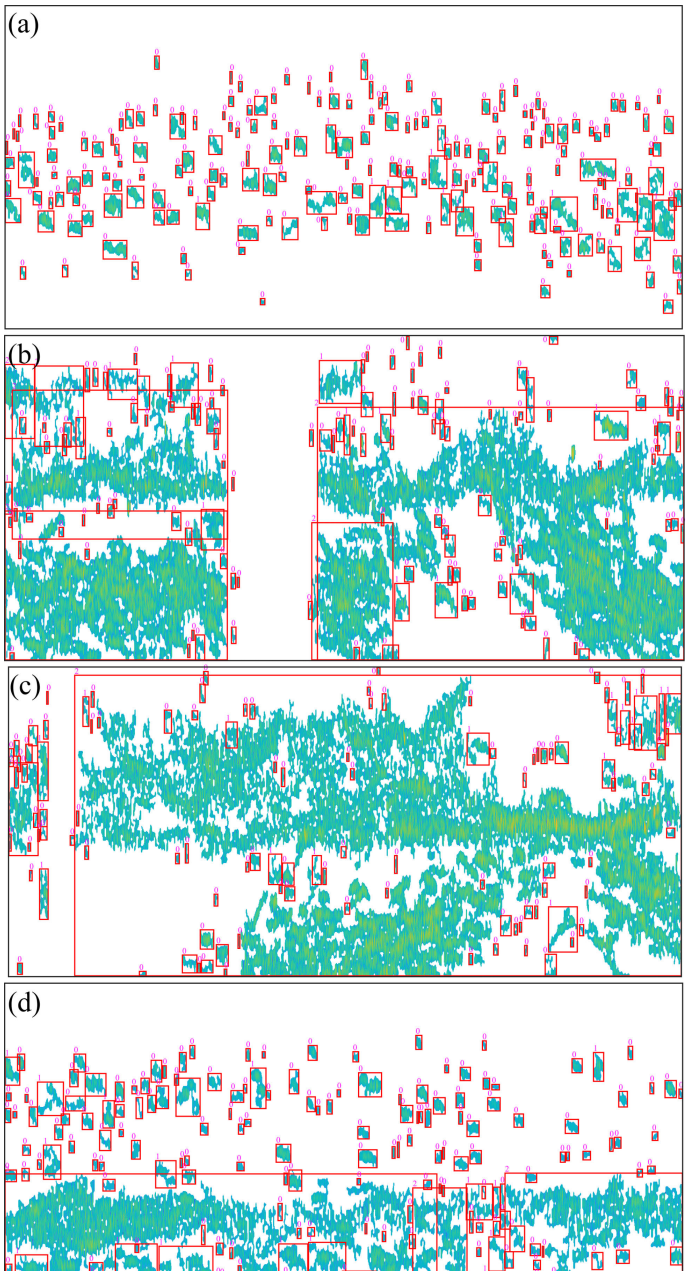


FIGURE 6
Results of echogram pre-processing and echo trace labeling. The figure panels of **(A–D)** correspond to the original echo images in Figure 5. The small pink numbers represent single fish ("0"), multiple fish ("1"), and fish groups ("2").

union (IOU) thresholds (Redmon and Farhadi, 2018) of 0.5 and 0.43 at an IOU threshold of 0.5:0.95 after 300 rounds of training. These values indicate that the prediction accuracy is low when the set prediction box and the actual box have an overlap of 50–95%, and most targets at the set prediction box and the actual box at 50% overlap are accurately predicted. The identification accuracy of the model is higher for larger objects in the echogram and lower for smaller objects in the echogram, which is also consistent with the F1 score curve for evaluating model performance (Figure 7). The maximum F1 score of the trained automatic detection model is

TABLE 2 The main results of model training.

Parameter	Precision (P)	Recall (R)	mAP_0.5	mAP_0.5:0.95
Result	0.79	0.68	0.71	0.43
Epoch	280	210	280	299

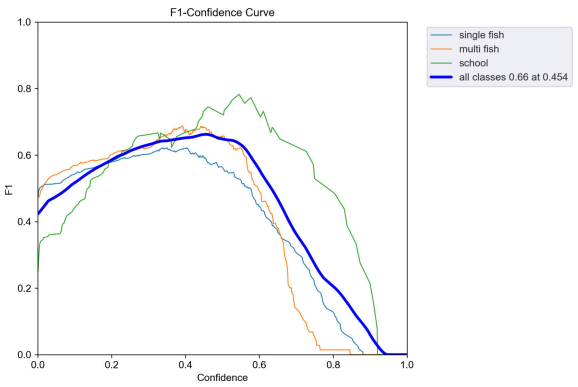


FIGURE 7
Curves of F1 scores related to the confidence level. The thinner three lines indicate the F1 scores for each class. The thick line indicates the F1 score for all classes.

0.66, which still represents a large advantage. The number of images in the training set and the resolution of the images are essential factors affecting the F1 score of the model (Chicco and Jurman, 2020; Jalal et al., 2020; Fourure et al., 2021). The single-beam echograms are sparse and contain less information in one echo trace. Therefore, more samples are needed to improve the F1 score of the trained model.

4.2 Calibration of the commercial echo sounder

The measured *in situ MV* histogram curves were similar to those obtained by Sawada et al. (Sawada et al., 2011) when measuring the *in situ* target strength of *Diaphus theta*, in which the distribution of the value at site Jul 5, 2021, had a larger interval and a higher mean value than that at site Jun 04, 2021, and the distribution at site Jun 04, 2021, was more concentrated. The distribution of TS_{model} measured from the fork length at the site sampled by the PSM method was similar to the distribution of measurement values obtained *in situ*. The distribution of TS_{model} on Jun 4, 2021, was mainly concentrated between -45 dB and -46.5 dB, while the distribution on Jul 5, 2021, was in the range of -42 dB to -47 dB, which is largely different from the *in situ MV* distribution characteristics. The mean TS_{model} at the two sites were -45.82 dB and -44.34 dB, respectively, while the mean target strength of the Pacific

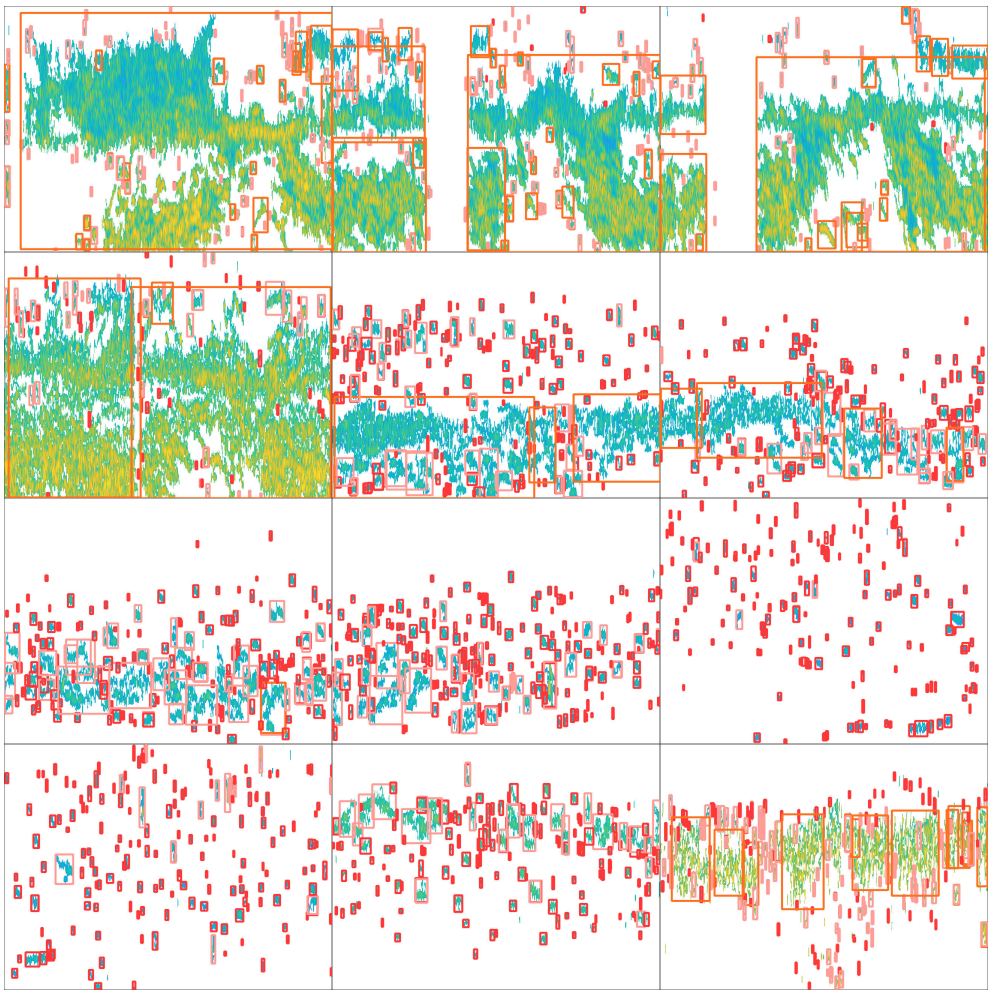


FIGURE 8
Automatic annotation of test set echograms.

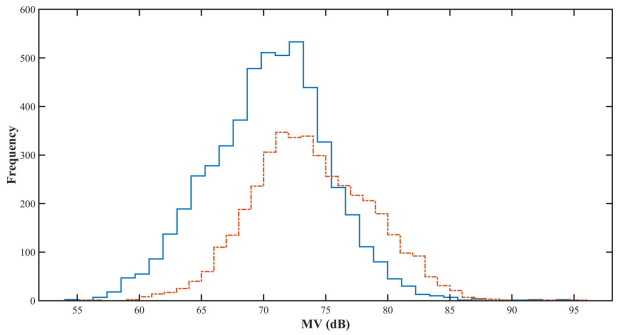


FIGURE 9
Observed *in situ MV* histograms of Pacific saury from two sampling sites (Jun 4, 2021, and Jul 5, 2021.) The solid blue line was sampled on Jun 4, 2021, and the dashed orange line was sampled on Jul 5, 2021.

TABLE 3 Average, standard deviation, maximum, and minimum fork length of Pacific saury at two sampling sites, which had synchronized acoustic data and biological sampling data.

	Sampling sites	
	Jun 4, 2021	Jul 5, 2021
Number	50	50
Avg. (mm)	282.84	242.04
S.D. (mm)	8.01	30.92
Max (mm)	305	301
Min (mm)	267	192

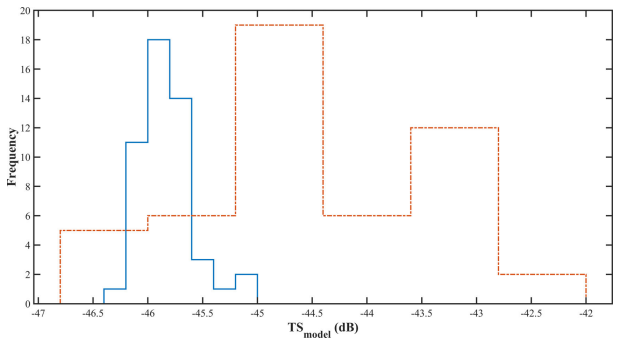


FIGURE 10
The TS_{model} of Pacific saury measured using the prolate spheroidal model. The solid blue line was sampled on Jun 4, and the dashed orange line was sampled on Jul 5.

saury calculated by PSM by Sawada et al. (Sawada et al., 2009) was -39.9 dB. A gap exists between the target strength calculated by the developed model and that calculated by Sawada et al. This may be caused by the following reasons: First, Sawada et al. used fewer samples for their calculations, all of which were based on "bird sampled". This makes the target strength value selective and leads to a smaller interval distribution. Second, the frequency they used was 70 kHz, and the frequency used in the present study was 50 kHz. The target strength values of fish were different at different frequencies. In PSM calculations, the angle of inclination of the swim bladder is another important factor that affects the target strength value of fish. In this study, the typical swim bladder length to fork length ratio was substituted into the PSM model for calculations. The size of the swim bladder tilt angle was not adequately considered. Measurements of the tilt angle distribution of swim bladder are necessary in further studies.

The living fish calibration offset K calculated by *in situ MV* and TS_{model} for the two sites differed by 1.35 dB and 0.54 dB, respectively, compared to the K_0 calibrated in the laboratory using the standard sphere method. According to the standard deviation threshold of 2 dB given by the Biosonics instrument calibration manual (Biosonics, 2004), the values obtained in the present study were within the standard range. The shipboard commercial echo sounder can carry out scientific acoustic survey work. As a rapid acoustic instrument performance testing method, the living fish calibration method is also feasible to a certain degree. The calibration method for rapid instrument performance testing can efficiently obtain more accurate acoustic survey data to expand the coverage area of fish resources. However, compared to the calibration of acoustic instruments using the standard calibration sphere method, there are still certain deviations, which mostly originate from the swimming behavior of fish and physical changes in the marine environment (Simmonds and Maclennan, 2008).

4.3 Fishing vessel acoustic monitoring

Commercial fishing vessels worldwide are commonly equipped with echo sounders for vertical detection of underwater information. However, current acoustic monitoring of fishery resources still relies on research vessels. In most cases, the underwater information detected by commercial echo sounders is not collected and analyzed. The main reasons for this situation include the absence of information such as geographic information location and time series associated with the echo intensity level; moreover, the echo sounders are often not calibrated when using fishing vessels for acoustic monitoring (Haris et al., 2021). These reasons result in the acoustic data collected by commercial fishing vessels remaining unutilized, as these data cannot be applied to classify fish species and assess resources.

TABLE 4 Mean values of *in situ MV*, TS_{model}, A_{soft}, and K of the Pacific saury measured at two sampling sites on Jun 4 and Jul 5; the value of K_0 is measured from the standard sphere calibration process.

Sampling site	*MV* (dB)	TS_{model}(dB)	A_{soft}	K(dB)	K_0(dB)
Jun 4	70.48	-45.82	-95.29	116.30	117.65
Jul 5	73.85	-44.34	-92.47	118.19	

The most important work of this study was the combination of the automatic detection model and the living fish calibration method to propose an echo sounder calibration method that is suitable for commercial fishing vessels. The developed method uses a deep learning target recognition method (YOLO v5) to quickly identify single fish echo traces in the echogram without the need to extract feature parameters by a manual operation before identification. Identification is based on the absolute dominance of the target fish species in the fishing process. The ease of access to target biological samples during fishing operations enables the measurement of model target strength values in a short period of time using the PSM method. The performance of the shipboard echo sounder is tested by comparing it with the *in situ* measurement value and deriving the offset of the acoustic data. The method can be used without impacting fishing operations. The offset is removed in a subsequent pre-processing step to make the data available for scientific research. The single-beam acoustic data used in this study are commonly available on commercial fishing vessels. The sparse nature of the single-beam data enables the acquisition of more acoustic detection areas with less storage space. The species classification results obtained by identifying single-beam data can be used for resource assessments and can aid fishing staff. For multi-species mixed fisheries, the method still needs further verification. With the development of fish detection technology, echo sounders equipped with multi-beam and broadband transducers are gradually used on fishing vessels. Of these, the broadband acoustic technique can obtain continuous echo features over the entire frequency band range, obtain a spectrogram of target echo intensity with frequency, and increase the amount of information on an individual echo trace (Xue et al., 2021). When using deep learning methods for target recognition, the developed method increases the training accuracy of the model and improves the success rate of target detection. Applying this method to broadband acoustic data is an important direction for future research.

5 Conclusions

Fishing vessels equipped with echosounders provide unique opportunities for the monitoring and assessment of fishery resources. A key challenge in the use of echo data collected from commercial echosounders is data calibration. This paper presents a deep learning method for the automatic detection of single fish echo traces. The results demonstrated that by combining the detected single fish echo traces with fishing samples, the echo data could be calibrated to a level similar to that of scientific echosounders, which aids scientific interpretation of these data. However, the current calibration method is still at a relatively moderate level, and traditional calibration with a standard sphere should be conducted whenever an opportunity arises.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

ST and JT designed the study. JH and JT provided the methodology and developed the data recording equipment. WW conducted the investigation. WW, MX, and ZZ analyzed the data. WW wrote the original draft. JT reviewed and edited the draft. All authors contributed to the article and approved the submitted version.

Funding

This research was funded by the National Key R&D Program of China (2019YFD0901401) and the Key Laboratory of Marine Ecological Monitoring and Restoration Technologies (MEMRT202202). We also acknowledge funds provided by the Ministry of Agriculture and Rural Affairs of China, through the project on the Survey and Monitor-Evaluation of Global Fishery Resources.

Acknowledgments

The authors of this research would like to thank the captain and all the crews of the FV Ming Hua for providing the investigation platform and related facilities of data collection in this study. The authors also thank the China Aquatic Products Zhoushan Marine Fisheries Corporation for help with implementating the research project. The help of Taoxi Xue, a member of staff of the Yellow Sea Fisheries Research Institute, is particularly appreciated for his assistance in the survey work.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1162064/full#supplementary-material

References

Albawi, S., Bayat, O., Al-Azawi, S., and Ucan, O. N. (2018). Social touch gesture recognition using convolutional neural network. *Comput. Intell. Neurosci.* 2018, 6973103. doi: 10.1155/2018/6973103

Aranis, A., de la Cruz, R., Montenegro, C., Ramírez, M., Caballero, L., Gómez, A., et al. (2022). Meta-estimation of araucanian herring, *Strangomera bentincki* (Norman 1936), biological indicators in the central-south zone of Chile (32°–47° LS). *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.886321

Axenrot, T., Didrikas, T., Danielsson, C., and Hansson, S. (2004). Diel patterns in pelagic fish behaviour and distribution observed from a stationary, bottom-mounted, and upward-facing transducer. *ICES J. Mar. Sci.* 61, 1100–1104. doi: 10.1016/j.icesjms.2004.07.006

Biosonics, I. (2004). Calibration of BioSonics digital scientific echosounder using T/C calibration spheres. (Seattle, WD, USA: Biosonics Inc.), 1–11. Available at: http://www.biosonicsinc.com/doc_library/docs/DTXcalibration2e.pdf.

Boswell, K. M., D'Elia, M., Johnston, M. W., Mohan, J. A., Warren, J. D., Wells, R. J. D., et al. (2020). Oceanographic structure and light levels drive patterns of sound scattering layers in a low-latitude oceanic system. *Front. Mar. Sci.* 7. doi: 10.3389/fmars.2020.00051

Boyra, G., Moreno, G., Orue, B., Sobradillo, B., and Sancristobal, I. (2019). *In situ* target strength of bigeye tuna (Thunnus obesus) associated with fish aggregating devices. *ICES J. Mar. Sci.* 76, 2446–2458. doi: 10.1093/icesjms/fsz131

Brautaset, O., Waldeland, A. U., Johnsen, E., Malde, K., Eikvil, L., Salberg, A.-B., et al. (2020). Acoustic classification in multifrequency echosounder data using deep convolutional neural networks. *ICES J. Mar. Sci.* 77 (4), 1391–1400. doi: 10.1093/icesjms/fsz235

Chicco, D., and Jurman, G. (2020). The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics* 21, 6. doi: 10.1186/s12864-019-6413-7

Coetzee, J. (2000). Use of a shoal analysis and patch estimation system (SHAPES) to characterise sardine schools. *Aquat. Living Resour.* 13, 1–10. doi: 10.1016/S0990-7440(00)00139-X

Didrikas, T., and Hansson, S. (2004). *In situ* target strength of the Baltic Sea herring and sprat. *ICES J. Mar. Sci.* 61, 378–382. doi: 10.1016/j.icesjms.2003.08.003

Eleftherakis, D., Berger, L., Le Bouffant, N., Pacault, A., Augustin, J.-M., and Lurton, X. (2018). Backscatter calibration of high-frequency multibeam echosounder using a reference single-beam system, on natural seafloor. *Mar. Geophys. Res.* 39, 55–73. doi: 10.1007/s11001-018-9348-5

Fallon, N. G., Fielding, S., and Fernandes, P. G. (2016). Classification of southern ocean krill and icefish echoes using random forests. *ICES J. Mar. Sci.* 73, 1998–2008. doi: 10.1093/icesjms/fsw057

Fang, J., and Wang, P. (2021). Application of improved YOLO V3 algorithm for target detection in echo image of sonar under reverb. *J. Phys.: Conf. Ser.* 1748, 42048. doi: 10.1088/1742-6596/1748/4/042048

Fernandes, P. G. (2009). Classification trees for species identification of fish-school echotraces. *ICES J. Mar. Sci.* 66, 1073–1080. doi: 10.1093/icesjms/fsp060

Fernandes, P. G., Copland, P., Garcia, R., Nicosevici, T., and Scoulding, B. (2016). Additional evidence for fisheries acoustics: small cameras and angling gear provide tilt angle distributions and other relevant data for mackerel surveys. *ICES J. Mar. Sci.* 73, 2009–2019. doi: 10.1093/icesjms/fsw091

Foote, K. G., and Rothschild, B. J. (2009). "Acoustic methods: brief review and prospects for advancing fisheries research," in *The future of fisheries science in north america. fish & fisheries series*, vol. 31 . Ed. R. J. Beamish (Dordrecht: Springer), 313–343. doi: 10.1007/978-1-4020-9210-7_18

Fourure, D., Javaid, M. U., Posocco, N., and Tihon, S. (2021). "Anomaly detection: how to artificially increase your F1-score with a biased evaluation protocol," in *Machine learning and knowledge discovery in databases. applied data science track. ECML PKDD 2021. lecture notes in computer science*. Eds. Y. Dong, N. Kourtellis, B. Hammer and J. A. Lozano (Cham: Springer), 12978. doi: 10.1007/978-3-030-86514-6_1

Furusawa, M. (1988). Prolate spheroidal models for predicting general trends of fish target strength. *J. Acoust. Soc. Japan (E)* 9, 13–24. doi: 10.1250/ast.9.13

Gjøsæter, H., Wiebe, P. H., Knutsen, T., and Ingvaldsen, R. B. (2017). Evidence of diel vertical migration of mesopelagic sound-scattering organisms in the Arctic. *Front. Mar. Sci.* 4. doi: 10.3389/fmars.2017.00332

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2018). Recent advances in convolutional neural networks. *Pattern recognit.* 77, 354–377. doi: 10.1016/j.patcog.2017.10.013

Haris, K., Kloser, R. J., Ryan, T. E., Downie, R. A., Keith, G., and Nau, A. W. (2021). Sounding out life in the deep using acoustic data from ships of opportunity. *Sci. Data* 8, 1–23. doi: 10.6084/m9.figshare.13172516

Ito, M., Matsuo, I., Imaizumi, T., Akamatsu, T., Wang, Y., and Nishimori, Y. (2013). "Classification of fish schools based on acoustic features associated with tilt angle," in *2013 IEEE International Underwater Technology Symposium (UT)*, Tokyo, Japan. 2013, 1–4. doi: 10.1109/UT.2013.6519865

Jalal, A., Salman, A., Mian, A., Shortis, M., and Shafait, F. (2020). Fish detection and species classification in underwater environments using deep learning with temporal information. *Ecol. Inf.* 57, 101088. doi: 10.1016/j.ecoinf.2020.101088

Johannesson, K., and Losse, G. (1977). Methodology of acoustic estimations of fish abundance in some UNDP/FAO resource survey projects. *Rapports Proces-Verbaux Des. Reunions (ICES)* 170, 296–318.

Julie, S., Anne, L. D., Paulo, T., Sven, G., Gildas, R., Gary, V., et al. (2020). *In situ* target strength measurement of the black triggerfish melichthys niger and the ocean triggerfish canthidermis sufflamen. *Mar. Freshw. Res.* 71, 1118–1127. doi: 10.1071/MF19153

Khodabandeloo, B., Agersted, M. D., Klevjer, T., Macaulay, G. J., and Melle, W. (2021). Estimating target strength and physical characteristics of gas-bearing mesopelagic fish from wideband *in situ* echoes using a viscous-elastic scattering model. *J. Acoust. Soc. America* 149, 673–691. doi: 10.1121/10.0003341

Knudsen, H. P. (2009). Long-term evaluation of scientific-echosounder performance. *ICES J. Mar. Sci.* 66, 1335–1340. doi: 10.1093/icesjms/fsp025

Lawson, G. L., Barange, M., and Fréon, P. (2001). Species identification of pelagic fish schools on the south African continental shelf using acoustic descriptors and ancillary information. *ICES J. Mar. Sci.* 58, 275–287. doi: 10.1006/jmsc.2000.1009

Lee, K. H., Lee, D. J., Kim, H. S., and Park, S. W. (2010). Swimming speed measurement of pacific saury (Cololabis saira) using acoustic Doppler current profiler. *J. Korean Soc. Fish. Ocean Technol.* 46 (2), 165–172. doi: 10.3796/ksft.2010.46.2.165

LeFeuvre, P., Rose, G., Gosine, R., Hale, R., Pearson, W., and Khan, R. (2000). Acoustic species identification in the Northwest Atlantic using digital image processing. *Fish. Res.* 47, 137–147. doi: 10.1016/S0165-7836(00)00165-X

Li, X., Shang, M., Hao, J., and Yang, Z. (2016). Accelerating fish detection and recognition by sharing CNNs with objectness learning. *OCEANS 2016 - Shanghai Shanghai China* 2016, 1–5. doi: 10.1109/OCEANSAP.2016.7485476

Li, X., Shang, M., Qin, H., and Chen, L. (2015). *Fast accurate fish detection and recognition of underwater images with fast r-cnn. OCEANS 2015 - MTS/IEEE Washington* (Washington, DC: IEEE) 2015, 1–5. doi: 10.23919/OCEANS.2015.7404464

Martignac, F., Daroux, A., Bagliniere, J. L., Ombredane, D., and Guillard, J. (2015). The use of acoustic cameras in shallow waters: new hydroacoustic tools for monitoring migratory fish population. a review of DIDSON technology. *Fish fish.* 16, 486–510. doi: 10.1111/faf.12071

Melvin, G. D., Kloser, R., and Honkalehto, T. (2016). The adaptation of acoustic data from commercial fishing vessels in resource assessment and ecosystem monitoring. *Fish. Res.* 178, 13–25. doi: 10.1016/j.fishres.2015.09.010

O'Donncha, F., Stockwell, C. L., Planellas, S. R., Micallef, G., Palmes, P., Webb, C., et al. (2021). Data driven insight into fish behaviour and their use for precision aquaculture. *Front. Anim. Sci.* 2. doi: 10.3389/fanim.2021.695054

Rathi, D., Jain, S., and Indu, S. (2017). "Underwater fish species classification using convolutional neural network and deep learning," in *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*. 2017, 1–6 (Bangalore, India: IEEE). doi: 10.1109/ICAPR.2017.8593044

Redmon, J., and Farhadi, A. (2018). Yolov3: an incremental improvement. *Comput. Vision Pattern Recognit.* 1804, 2767. doi: 10.48550/arXiv.1804.02767

Reid, D. G. (2000). Report on echo trace classification. *ICES Coop. Res. Rep.* 238, 1–115. doi: 10.17895/ices.pub.5371

Robotham, H., Bosch, P., Gutiérrez-Estrada, J. C., Castillo, J., and Pulido-Calvo, I. (2010). Acoustic identification of small pelagic fish species in Chile using support vector machines and neural networks. *Fish. Res.* 102, 115–122. doi: 10.3135/jmasj.20.73

Sawada, K., Furusawa, M., and Williamson, N. J. (1993). Conditions for the precise measurement of fish target strength *in situ*. *J. Mar. Acoust. Soc. Japan* 20, 73–79. doi: 10.3135/jmasj.20.73

Sawada, K., Takahashi, H., Abe, K., Ichii, T., Watanabe, K., and Takao, Y. (2009). Target-strength, length, and tilt-angle measurements of pacific saury (Cololabis saira) and Japanese anchovy (Engraulis japonicus) using an acoustic-optical system. *ICES J. Mar. Sci.* 66, 1212–1218. doi: 10.1093/icesjms/fsp079

Sawada, K., Uchikawa, K., Matsuura, T., Sugisaki, H., Amakasu, K., and Abe, K. (2011). *In situ* and ex situ target strength measurement of mesopelagic lanternfish, diaphus theta (Family myctophidae). *J. Mar. Sci. Technol.* 19, 10. doi: 10.51400/2709-6998.2196

Simmonds, J., and Maclennan, D. N. (2008). *Fisheries acoustics: theory and practice* (New York: John Wiley & Sons).

Slotte, A., Hansen, K., Dalen, J., and Ona, E. (2004). Acoustic mapping of pelagic fish distribution and abundance in relation to a seismic shooting area off the Norwegian west coast. *Fish. Res.* 67, 143–150. doi: 10.1016/j.fishres.2003.09.046

Sobradillo, B., Boyra, G., Martinez, U., Carrera, P., Peña, M., and Irigoien, X. (2019). Target strength and swimbladder morphology of mueller's pearlside (Maurolicus muelleri). *Sci. Rep.* 9, 17311. doi: 10.1038/s41598-019-53819-6

Tong, J., Xue, M., Zhu, Z., Wang, W., and Tian, S. (2022). Impacts of morphological characteristics on target strength of chub mackerel (Scomber japonicus) in the Northwest pacific ocean. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.856483

Trumpickas, J., Pinder, M., and Dunlop, E. S. (2020). Effects of vessel size and trawling on estimates of pelagic fish backscatter in lake Huron. *Fish. Res.* 224, 105430. doi: 10.1016/j.fishres.2019.105430

Tsagarakis, K., Pyrounaki, M., Giannoulaki, M., Somarakis, S., and Machias, A. (2012). Ontogenetic shift in the schooling behaviour of sardines, sardina pilchardus. *Anim. Behav.* 84, 437–443. doi: 10.1016/j.anbehav.2012.05.018

Wageeh, Y., Mohamed, H. E.-D., Fadl, A., Anas, O., Elmasry, N., Nabil, A., et al. (2021). YOLO fish detection with euclidean tracking in fish farms. *J. Ambient Intell. Humanized Comput.* 12, 5–12. doi: 10.1007/s12652-020-02847-6

Wang, N., Chen, T., Kong, X., Chen, Y., Wang, R., Gong, Y., et al. (2023d). Underwater attentional generative adversarial networks for image enhancement. *IEEE Trans. Human-Machine Syst.* 1-, 11. doi: 10.1109/THMS.2023.3261341

Wang, N., Chen, T., Liu, S., Wang, R., Karimi, H. R., and Lin, Y. (2023a). Deep learning-based visual detection of marine organisms: a survey. *Neurocomputing* 532, 1–32. doi: 10.1016/j.neucom.2023.02.018

Wang, H., Sun, S., Bai, X., Wang, J., and Ren, P. (2023b). A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes. *IEEE J. Oceanic Eng.* 48, 2: 443–2: 461. doi: 10.1109/JOE.2022.3226202

Wang, H., Sun, S., and Ren, P. (2023c). Meta underwater camera: a smart protocol for underwater image enhancement. *ISPRS J. Photogrammetry Remote Sens.* 195, 462–481. doi: 10.1016/j.isprsjprs.2022.12.007

Wang, H., Sun, S., Wu, X., Li, L., Zhang, H., Li, M., et al. (2021). "A yolov5 baseline for underwater object detection," in *OCEANS 2021* (San Diego–Porto: IEEE), 1–4. doi: 10.23919/OCEANS44145.2021.9705896

Wang, N., Wang, Y., and Er, M. J. (2022). Review on deep learning techniques for marine object recognition: architectures and algorithms. *Control Eng. Pract.* 118, 104458. doi: 10.1016/j.conengprac.2020.104458

Xue, M., Tong, J., Tian, S., and Wang, X. (2021). Broadband characteristics of zooplankton sound scattering layer in the kuroshio–oyashio confluence region of the Northwest pacific ocean in summer of 2019. *J. Mar. Sci. Eng.* 9, 938. doi: 10.3390/jmse9090938

frontiers | Frontiers in Marine Science

TYPE Review
PUBLISHED 05 June 2023
DOI 10.3389/fmars.2023.1157370

Check for updates

OPEN ACCESS

EDITED BY
Mark C. Benfield,
Louisiana State University,
United States

REVIEWED BY
Nils Piechaud,
Norwegian Institute of Marine Research (IMR), Norway
Lina Zhou,
Hong Kong Polytechnic University,
Hong Kong SAR, China

*CORRESPONDENCE
Andrew M. Hein
andrew.hein@cornell.edu

RECEIVED 08 February 2023
ACCEPTED 19 May 2023
PUBLISHED 05 June 2023

CITATION
Belcher BT, Bower EH, Burford B, Celis MR, Fahimipour AK, Guevara IL, Katija K, Khokhar Z, Manjunath A, Nelson S, Olivetti S, Orenstein E, Saleh MH, Vaca B, Valladares S, Hein SA and Hein AM (2023) Demystifying image-based machine learning: a practical guide to automated analysis of field imagery using modern machine learning tools.
Front. Mar. Sci. 10:1157370.
doi: 10.3389/fmars.2023.1157370

COPYRIGHT
© 2023 Belcher, Bower, Burford, Celis, Fahimipour, Guevara, Katija, Khokhar, Manjunath, Nelson, Olivetti, Orenstein, Saleh, Vaca, Valladares, Hein and Hein. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Demystifying image-based machine learning: a practical guide to automated analysis of field imagery using modern machine learning tools

Byron T. Belcher[1], Eliana H. Bower[1], Benjamin Burford[1,2], Maria Rosa Celis[1,2], Ashkaan K. Fahimipour[1,2,3], Isabela L. Guevara[1], Kakani Katija[4], Zulekha Khokhar[1], Anjana Manjunath[1], Samuel Nelson[1,2], Simone Olivetti[1,2], Eric Orenstein[4], Mohamad H. Saleh[1], Brayan Vaca[1], Salma Valladares[1], Stella A. Hein[1,5] and Andrew M. Hein[1,6]*

[1]AI for the Ocean program, University of California Santa Cruz, Santa Cruz, CA, United States, [2]Institute of Marine Sciences, University of California Santa Cruz, Santa Cruz, CA, United States, [3]Florida Atlantic University, Department of Biology, Boca Raton, FL, United States, [4]Monterey Bay Aquarium Research Institute, Research and Development, Moss Landing, CA, United States, [5]Cornell University, College of Agriculture and Life Sciences, Ithaca, NY, United States, [6]Department of Computational Biology, Cornell University, Ithaca, NY, United States

Image-based machine learning methods are becoming among the most widely-used forms of data analysis across science, technology, engineering, and industry. These methods are powerful because they can rapidly and automatically extract rich contextual and spatial information from images, a process that has historically required a large amount of human labor. A wide range of recent scientific applications have demonstrated the potential of these methods to change how researchers study the ocean. However, despite their promise, machine learning tools are still under-exploited in many domains including species and environmental monitoring, biodiversity surveys, fisheries abundance and size estimation, rare event and species detection, the study of animal behavior, and citizen science. Our objective in this article is to provide an approachable, end-to-end guide to help researchers apply image-based machine learning methods effectively to their own research problems. Using a case study, we describe how to prepare data, train and deploy models, and overcome common issues that can cause models to underperform. Importantly, we discuss how to diagnose problems that can cause poor model performance on new imagery to build robust tools that can vastly accelerate data acquisition in the marine realm. Code to perform analyses is provided at https://github.com/heinsense2/AIO_CaseStudy.

KEYWORDS

machine learning, image analysis, deep neural network, underwater imagery, computer vision, artificial intelligence, distribution shift

1 Introduction

Imagery from the ocean has long been used to survey marine environments, quantify physical conditions, and monitor the inhabitants of marine ecosystems (Longley and Martin, 1927; Drew, 1977; Beijbom et al., 2015; Lombard et al., 2019; Marochov et al., 2021). This reliance on imagery as a means of extracting data from marine systems has only grown with the increasing accessibility of satellite imagery and the decreasing cost and increasing quality of imaging systems that can be deployed directly in the field (Durden et al., 2016; Williams et al., 2019; Bamford et al., 2020; Rodriguez-Ramirez et al., 2020). Yet visual data bring with them some unique challenges. Images and video are expensive to process due in part to the fact that imagery is inherently high-dimensional; for example, a single grayscale image of one-megapixel resolution, a coarse image by modern standards, is a 2^{20}-dimensional data object. Researchers who collect imagery in the course of their work often return from field campaigns with terabytes to petabytes of such high-dimensional imagery that must then be processed (Schoening et al., 2018).

The role of ***image analysis*** (see Table 1 for glossary of bolded terms) is to compress high-dimensional visual data into much lower-dimensional summaries relevant to a particular task or study objective. As humans, we perform this type of visual data compression naturally (Marr, 1982). We look at an image and with proper training, can classify what is present in the image, localize and count distinct objects, and partition the image into regions of one type or another. The objective of image-based ***machine learning*** (ML), a subfield of ***computer vision***, is to train computer algorithms to perform these same tasks with a high level of accuracy. Doing so can tremendously accelerate image processing and greatly reduce its cost (Norouzzadeh et al., 2018), while also providing an explicit, standardized, and reproducible workflow that can be shared easily among researchers and applied to new problems (Goodwin et al., 2021; Katija et al., 2022). Despite the promise of these methods, the expertise required to apply, adapt, and troubleshoot ML methods using the kinds of image datasets marine scientists collect still creates a high barrier to entry (Crosby et al., 2023).

A number of recent articles provide overviews of how modern image-based machine learning methods work and how these methods have been applied to problems in marine science (*e.g.*, Michaels et al., 2019; Goodwin et al., 2021; Li et al., 2022). Here, we focus on the practical problem of how to implement image-based ML pipelines on real imagery from the field. The remainder of this paper is structured as a sequence of steps involved in defining an analytical task to be solved, preparing training data, training and evaluating models, deploying models on new data, and diagnosing and fixing performance issues. To provide concreteness, we present a running case study: object detection of marine species using imagery and software tools from the open source *FathomNet* database and interface (Katija et al., 2022). We use this case study to demonstrate each phase of constructing and troubleshooting a ML pipeline, and we provide code and guidelines needed to reproduce each step in a github repository: https://github.com/heinsense2/AIO_CaseStudy.

1.1 Building and using a machine learning pipeline

Researchers often have a clear idea of how they want to use the data extracted from imagery. This idea forms the starting point for designing a ***machine learning pipeline*** to automatically extract data from imagery. Building a machine learning pipeline to solve image analysis tasks involves a series of steps:

1.1.1 Define an analytical task

This step requires working to define the objective of image analysis and the target metrics to be extracted from imagery. The type of imagery to be analyzed should be specified. This step may also involve defining performance criteria and setting benchmarks for acceptable performance.

1.1.2 Generate and organize training and testing datasets

This step involves developing and organizing image libraries for training, testing, and deploying models. This involves both organizing imagery with appropriate file structure and, very often, hand-labeling ***ground truth*** data to be used to train and test models. This step requires software tools that allow a researcher to organize images and to label, or ***annotate***, imagery so it can be later used to train and test machine learning models.

1.1.3 Select and train appropriate machine learning models

This step requires identifying a machine learning model architecture capable of performing the desired image analysis task, as well as software and hardware implementations capable of training and deploying the model to perform inference on new imagery.

1.1.4 Evaluate model performance

This step involves summarizing and visualizing model predictions and performance measures, and often comparing these measures across alternative model architectures or training schedules.

1.1.5 Diagnose performance issues and apply interventions to improve performance

This step involves applying a trained model to new imagery and re-evaluating its performance. If performance is below target levels, it may be necessary to modify training methods, datasets, or model architecture to improve performance.

In the following sections, we walk through each of these steps to illustrate how each is accomplished, and how the steps combine to produce an adaptable pipeline with robust performance.

TABLE 1 Glossary of terms relevant to image-based machine learning.

Term	Definition
Image analysis	The process of extracting task-relevant information from imagery.
Machine learning	A body of mathematical and computational methods for extracting information from data to make predictions.
Machine learning pipeline	A computer program or set of programs that reads in training data, specifies and trains a ML model, produces model predictions, and provides performance metrics.
Image annotation	Process of generating ground truth labels for images, which are typically used to train ML models or evaluate performance.
Ground truth	A verified record, often produced by a human annotator, that describes what is contained within the image. Sometimes also called an annotation, or label.
Image classification	A task in which a whole image is assigned a class from a list of valid classes.
Object detection	A task in which objects within a set of classes of interest are detected and localized within an image, typically either within a bounding box, or polygon region. Many object detection methods also classify objects.
Instance segmentation	A task in which individual instances of objects in a class or classes of interest are localized within an image. Sometimes used synonymously with object detection, when objects are localized within polygons rather than bounding boxes.
Semantic segmentation	A task in which all individual pixels in an image are assigned to a class, but individual instances of objects are not specified.
Supervised learning	A type of machine learning that involves training a model with example input-output pairs.
Panoptic labels	A type of annotation that assigns a class to each pixel in an image and delineates the borders of instances of distinct objects of interest.
Few-shot learning	Machine learning methods designed to achieve good performance by training on few examples.
Deep neural network (DNN)	A machine learning method based on networks of interconnected computing nodes called "neurons." DNNs take data as input, process the data through one or more sequential layers of processing known as "hidden layers," and return predictions about the image.
Classification accuracy	Fraction of class predictions that are correct: (true positives + true negatives)/total number of predictions.
Precision	The fraction of positive class predictions that are correct: true positives/total predicted positives.
Recall	The fraction of positives present in the dataset that are correctly predicted by a model: true positives/total positives present in dataset. Sometimes referred to as *sensitivity*.
F1 score	A performance measure that incorporates both precision and recall: 2 (precision x recall)/(precision + recall).
Intersection-over-union (IoU)	A measure of spatial localization performance used in object detection and instance segmentation. IoU measures the number of pixels contained within both the predicted instance location and the ground truth ("intersection" between the two areas), divided by the total number of unique pixels contained within the predicted instance location, and ground truth ("union" of the two areas).
Mean average precision (mAP)	An average measure of classifier performance when bounding boxes or object instances are classified. Incorporates precision, recall, and IoU.
k-fold cross validation	A type of model evaluation in which training, validation, and test data are partitioned into k different splits, and performance measures are evaluated on each split.
Distribution shift	Systematic differences in image statistics, scene complexity, class identities and distributions, and other relevant features between a training set and a new dataset to which a model is to be applied.
Image augmentation	The process of applying random digital alterations to training imagery during the training process to improve model generalization.
Image resolution	The resolution of the image in pixels. Many ML pipelines reduce image resolution by default to save memory and reduce training and deployment times.

(Continued)

TABLE 1 Continued

Term	Definition
Background imagery	Images that do not contain classes of interest.
Class coarsening	The process of lowering the resolution of classes by grouping several fine classes (e.g., species A, B, C, and D) into coarser classes (e.g., genus 1, genus 2).

2 Defining an image analysis task

2.1 Overview

Defining the image analysis task to be solved is the first step in any machine learning pipeline. Is the goal to assign an image to one class or another – for example, to decide whether a particular species is or is not present or a particular environmental condition is or is not met? Or is the aim instead to identify and count objects of interest – for example, to find all crustaceans in an image and identify them to genus? Or is the objective to divide regions of the image into distinct types and quantify the prevalence of those types – for example, to partition the fraction of a benthic image occupied by different algae or coral morphotypes? The answers to these questions determine how one proceeds with gathering appropriate labeled data, selecting and training a model, and deploying that model on new data.

2.2 Technical considerations

Many of the traditional problems marine scientists currently use imagery to address fall into one of three categories: *image classification, object detection, or semantic segmentation*. More complex tasks such as tracking (Katija et al., 2021; Irisson et al., 2022), functional trait analysis (Orenstein et al., 2022), pose estimation (Graving et al., 2019), and automated measurements (Fernandes et al., 2020) often rely on these more basic tasks as building blocks.

In ***image classification*** problems, a computer program is presented with an image and asked to assign the image to one of a set of classes. Classes could be defined based on the presence or absence of particular objects (e.g., shark present or shark absent; Sharma et al., 2018), or represent a set of categories to which the image must be assigned, for example on the basis of what kind of animal is present in the image (Piechaud et al., 2019) or what type of habitat is represented in the image (Jackett et al., 2023). An important distinction between whole-image classification and other common image analysis tasks is that in image classification, classes are assigned at the scale of the entire image (Chapelle et al., 1999; Fei-Fei et al., 2004). Thus, objects of interest are not spatially localized within the image, nor does the model provide information on the properties of individual pixels or spatial regions within the image. Whole image classification is appropriate for some tasks, such as simply detecting the presence or absence of a particular species of interest or environmental condition, but is less appropriate for others, for example, counting individuals of a particular species when multiple individuals can occur within a single image (Beery et al., 2021). Nevertheless, this task remains relevant in many automated image analysis problems (Qin et al., 2016; Villon et al., 2021; Kyathanahally et al., 2022) and is the approach of choice for certain types of marine microscopy data where images are typically stored as extracted region of interest (e.g., Luo et al., 2018; Ellen et al., 2019).

A second common task involves detecting and spatially localizing objects of interest within images, a task known as ***object detection*** or ***instance segmentation***. Separating instances of the same type of object (e.g., there are nine fish identified as Atlantic cod in this image) in a given image is often crucial if imagery is being used to estimate abundances (Moeller et al., 2018), and most object detection pipelines can also be trained to detect objects of many different classes, which is valuable for analyzing images that contain multiple objects of interest that belong to different classes (see Scoulding et al., 2022 for a discussion of limitations at high density).

A third task, known as ***semantic segmentation***, involves assigning a class to each pixel in an image. Semantic segmentation differs from object detection in that one is not interested in detecting and discriminating instances of a particular class, but rather in determining the class membership of each pixel in an image. This can be useful for tasks such as estimating the percent cover of algae, corals, or other benthic substrate types (e.g., Beijbom et al., 2015; Williams et al., 2019). If images are collected in a controlled and standardized way, the percentage of each image occupied by different species or classes of object can be estimated by the relative abundance of pixels assigned to each class.

Image-based ML tools have also been used for a variety of applications beyond the three tasks described above. Examples include "structure-from-motion" studies, in which the three-dimensional structure of objects are inferred and reconstructed from a sequence of images taken from different locations in the environment (Francisco et al., 2020), animal tracking and visual field reconstruction (Hein et al., 2018; Fahimipour et al., 2023), quantitative measurement and size estimation (Fernandes et al., 2020), animal postural analysis (Graving et al., 2019), and re-identification of individual animals in new images based on a set of previous observations (Nepovinnykh et al., 2020).

2.3 Case study: species detection and classification from benthic and midwater imagery

To provide a concrete example, we consider an object detection and classification task that seeks to localize and identify marine animals in deep-sea imagery collected from the Eastern Pacific within the Monterey Bay and surrounding regions. Images were collected by the Monterey Bay Aquarium Research Institute (MBARI) during Remotely Operated Vehicle (ROV) surveys conducted between 1989 and 2021 (Robison et al., 2017), and are housed in the open-source *FathomNet* database (FathomNet.org; Katija et al., 2022). We focus on six common biological taxa that are observed broadly across the sampling domain, at a range of depths, and over several decades of sampling (Figure 1. shows iconic image of each class): the fish genera *Sebastes* (Rockfish) and *Sebastolobus* (Thornyheads), and the squid species *Dosidicus gigas* (Humboldt squid), *Chiroteuthis calyx* (swordtail squid), *Gonatus onyx* (black-eyed squid), and the siphonophore, *Nanomia bijuga*. Although classes of interest are sometimes clearly visible in images as shown in Figure 1, *FathomNet* contains many images with small subjects, complex visual backgrounds, heterogeneous lighting, and a host of other challenging visual conditions (Figure 2) that are ubiquitous in marine science applications.

We selected the six classes shown in Figures 1, 2 from the much larger set of classes available in *FathomNet* based on three criteria: (i) hundreds to thousands of human-generated labels were available for each class providing us with a sufficient number of labeled instances to explore performance of ML models under different partitions of the data, (ii) images of these classes were collected over a relatively broad spatial region and/or depth range compared to many other classes in *FathomNet*, allowing us to compare performance across spatial partitions of the data, and (iii) images of these classes were collected over many years, allowing us to partition the dataset temporally. Because searchable metadata, including depth and collection date, are included with the images in *FathomNet*, we were able to quickly create these partitions. As described in "*Diagnosing and Improving Model Performance on New Data*" below, we use these spatial and temporal partitions of the data to illustrate how ML models can fail when applied to new data, and how to diagnose and address such performance issues. We will return to this case study at the end of each section to provide a concrete example of each step involved in constructing and evaluating a machine learning pipeline.

3 Labeled imagery for training and evaluating models

3.1 Overview

The image-based ML methods that are currently most widely applied for marine science applications are based on ***supervised learning*** (Cunningham et al., 2008; Goodfellow et al., 2016). In supervised learning problems, the user provides a training dataset in which the desired output corresponding to a given input is specified for a set of examples. For object detection and classification problems, training data typically consist of a set of images (the image set) in which objects of interest are localized and identified by a human annotator. Labels (also sometimes referred to as "ground truths" or "annotations") are standardized records of identity and, in some cases, spatial information describing what is contained within the image.

To train a supervised ML pipeline to perform image analysis automatically, one needs a suitable training dataset consisting of images and corresponding labels. A researcher has two choices for acquiring labeled data: manually create a set of labels to be used for training, or use images and labels from a pre-existing database (Table 2). At present, the number of publicly available annotated datasets containing marine imagery is relatively small, and the size and spatial, temporal, and taxonomic coverage of these datasets is still rather limited. In practice, this means that researchers typically need to create a new training dataset of annotated imagery *de novo*. This custom training set can then be used as a stand-alone training set or combined with images and labels from existing databases to fully train a ML model to carry out a specified task (Knausgård et al., 2021).

3.2 Technical considerations

When building and working with training datasets, there are several issues a researcher should consider that can help determine which software tools are most useful, and how to best structure the labeling process to solve the desired image analysis task.

3.2 1 Label types

The most common method for creating new labels involves manual labeling of imagery (Mahajan et al., 2018; see Ji et al., 2019 for discussion of unsupervised methods). The type of label used depends on several considerations. The first consideration is the type of image analysis task that will allow the researcher to access the information they want to extract from the imagery (see "*Defining the image analysis task*" above).

If the objective is image classification, then labels consist of a class label assigned to each image in the training set (Figure 3A). For example, suppose the objective is to take in new images and to determine which images contain a target species and which do not. An appropriate training dataset would consist of a set of representative images from sampling cameras, each of which would be labeled by a human annotator as containing or not containing the target species.

If the objective of image analysis is to localize and classify objects within an image, then manually generated labels must contain information about the locations and classes of objects of interest within an image. The most commonly used labeling formats for object detection are bounding box labels and polygon labels (Figures 3B, C). Bounding boxes are rectangular regions that enclose each object of interest and carry the appropriate class ID for the object (Figure 3B). Polygon labels, sometimes also referred to

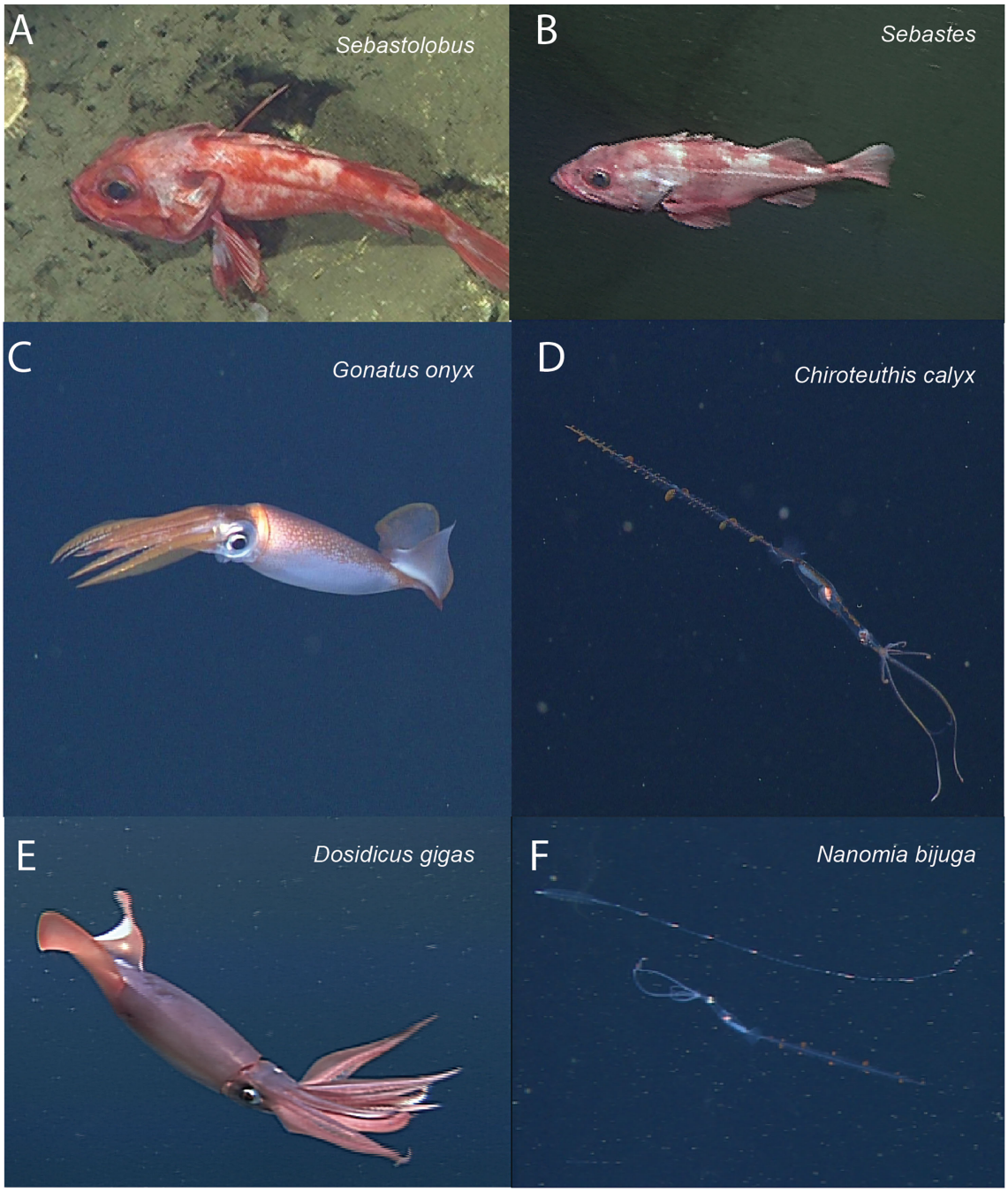


FIGURE 1
Focal species included in case study (iconic images). Focal species included fish in the genera Sebastolobus **(A)** and Sebastes **(B)**, squid species Gonatus onyx **(C)** Chiroteuthis calyx **(D)**, and Dosidicus gigas **(E)**. Panel **(F)** shows an image of the siphonophore, Nanomia bijuga, alongside a juvenile *C. calyx* (F, lower organism in image), which are believed to visually and behaviorally mimic N. bijuga (Burford et al., 2015). Images in panels **(A-F)** were selected for clarity and subjects are enlarged for visualization. (Figure 2) shows focal species in images that are more representative of typical images in FathomNet.

as "masks," are enclosing polygons that outline an object of interest (Figure 3C). These too are associated with the class label of the object.

If the objective of image analysis is to assign the pixels in an image to distinct classes (*i.e.*, semantic segmentation), for example to compute the fraction of the region captured in an image composed of different types of benthic cover, then labels must assign the pixels in an image to distinct classes (Figure 3D). This is typically done within labeling software by manually selecting the borders of local regions within the image and assigning a class to these regions. Some semi-automated "assisted methods" have been developed to aid in semantic labeling of images (e.g., Uijlings et al., 2020, "magic wand" tool in BIIGLE, Langenkämper et al., 2017).

Machine learning-based computer vision libraries such as Detectron 2 (Wu et al., 2019) and Deeplab v3+ (Chen et al., 2018)

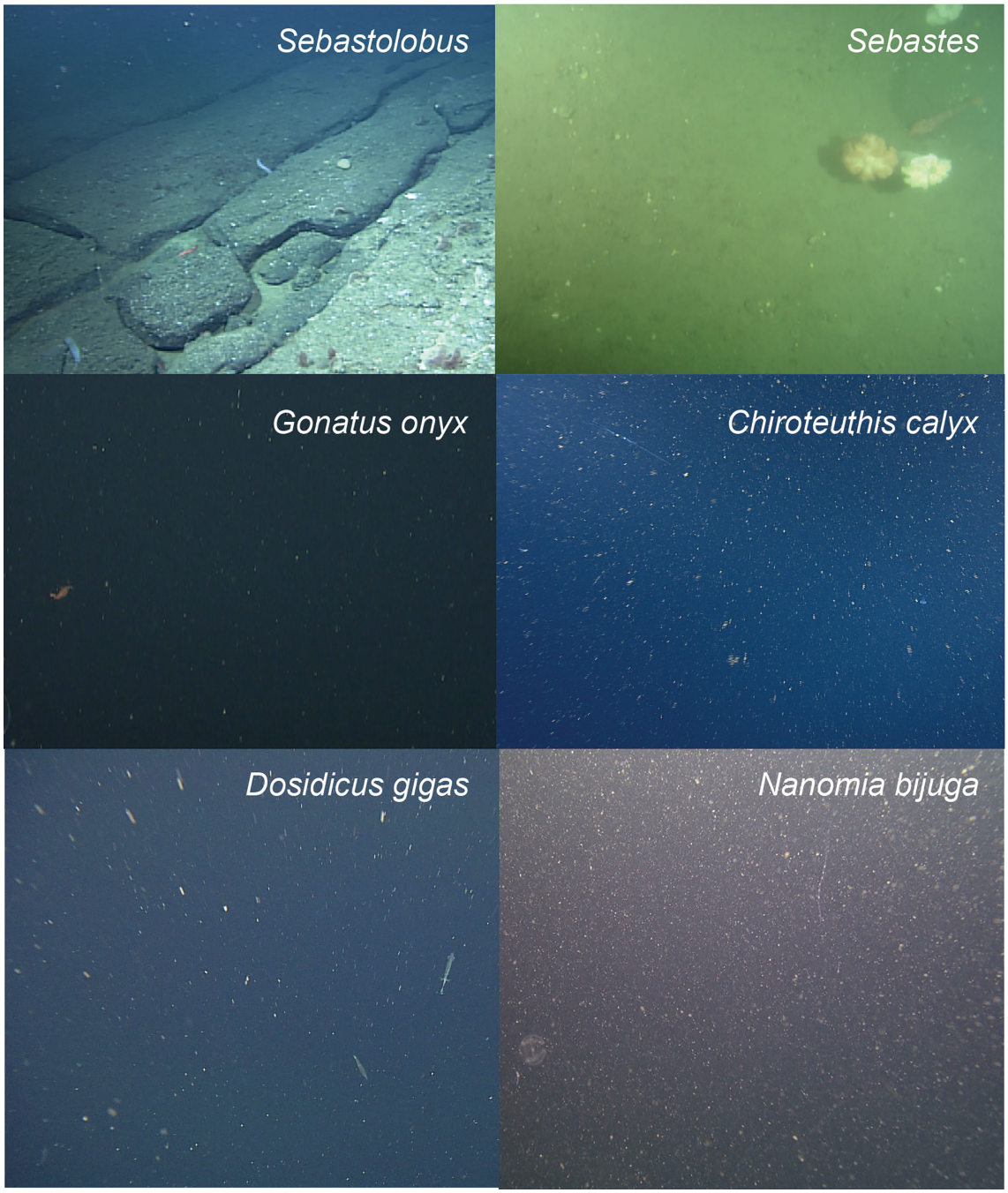


FIGURE 2
Typical images from *FathomNet* containing focal species. Focal species from Figure 1 shown in the context of more typical images from *FathomNet*. The focal class present in each image is noted in the upper right corner. Note complex and variable visual conditions, small size of objects of interest, clutter, and complex backgrounds. These conditions are typical in marine imagery collected for scientific sampling purposes.

contain models that operate on an additional type of label referred to as a ***panoptic label***. Panoptic labels include both class assignments for each pixel within an image and instance labels, so that the distinct pixels belonging to an individual instance of an object, for example, an individual squid, are grouped together (Figure 3D). We are not aware of past studies in marine science that have made use of panoptic labels, however, this type of labeling and segmentation could be useful in cases where a researcher wants to simultaneously characterize foreground objects of interest and background or substrate conditions.

3.2.2 Labeled data file formats

A variety of formats exist for storing manually generated labels. Unfortunately, there has been little standardization of the file formats used to encode labels of marine imagery, nor have researchers included consistent metadata within these files (Howell et al., 2019; Schoening et al., 2022). When creating new labels, we recommend choosing from among several formats that are most widely used in the computer vision community. These include YOLO text files, Pascal VOC XML files, and COCO ("common objects in context", https://

cocodataset.org/) Java Script Object Notation (JSON) formats. Pascal VOC and COCO formats both allow for convenient storage of metadata, making them attractive options.

3.2.3 Software for manually labeling imagery

A web search for the term "image labeling" will return many graphical user interface-based software tools designed to help users perform manual image labeling. In our experience, many of these tools work reliably, and are easy for human annotators to learn to use. Some widely-used, free labeling tools are CVAT (https://cvat.org), VGG Image Annotator (https://www.robots.ox.ac.uk/~vgg/software/via/), and Annotator J (https://biii.eu/annotatorj). Tools developed specifically for use in marine environments include BIIGLE (Langenkämper et al., 2017), VIAME (Richards et al., 2019), and EcoTaxa (Picheral et al., 2017; see Gomes-Pereira et al., 2016 for a review). These software tools are typically intuitive to use, but different tools have different capabilities that are important to understand when deciding which package to use for a given project. When selecting a software tool, there are four issues we suggest considering: (i) the speed and ease with which images can be loaded, labeled, and the labels exported; (ii) features the labeling tool offers such as convenient batch loading of images, zooming in and out, rotating images, assisted labeling, etc.; (iii) the

TABLE 2 Publicly available databases containing annotated images from marine environments.

Dataset Name	Subject	Approx. label count	Label type	Label file type	Geographic location	Published reference	URL
Save the Turtles	Turtles	2,000	Bounding box	.txt	Global	NA	1
OzFish	Fish	45,000	Bounding box	.json	Australia	doi: 10.25845/5e28f062c5097	2
Labeled fish in the wild	Fish	1,000	Bounding box	.dat	California	doi: 10.1109/WACVW.2015.11	3
Fathomnet	Marine organisms and objects	75,000	Bounding box	.json	Global	doi: 10.1038/s41598-022-19939-2	4
SUIM (Semantic Segmentation of Underwater Imagery)	Marine organisms and objects	1,500	Semantic segmentation	.bmp	Global	arXiv: 2004.01241	5
Fish-Pak	Fish	900	Whole image	NA	Pakistan	doi: 10.17632/n3ydw29sbz.3	6
Nature Conservancy Fisheries Monitoring	Fish aboard boats	8,000	Whole image	NA	Global	NA	7
CoralNet	Coral	94,000,000	Semantic segmentation	NA	Global	NA	8
LifeCLEF-16 Fish Dataset	Fish	9,000	Bounding box	.xml	Global	doi: 10.1007/978-3-319-44564-9_26	9
Trash-ICRA19: A Bounding Box Labeled Dataset of Underwater Trash	Marine robotics, debris, fauna	5,500	Bounding box	.json	Sea of Japan	doi: 10.1109/ICRA.2019.8793975	10
TrashCan 1.0: An Instance-Segmentation Labeled Dataset of Trash Observations	Marine robotics, debris, fauna	7,000	Instance segmentation	.json	Sea of Japan	arXiv: 2007.08097	11
Woods Hole Plankton Dataset	Marine plankton	3,500,000	Whole image	NA	Woods Hole Harbor	doi: 10.4319/lom.2007.5.204	12
Moorea labeled corals (MCL)	Corals and non-corals	400,000	Semantic segmentation	NA	Mo'orea	doi: 10.1109/CVPR.2012.6247798	13
RSMAS + EILAT	Corals	2,000	Whole image	NA	Red Sea	doi: 10.17632/86y667257h.2	14
ZooScan	Marine zooplankton	19,000	Whole image	NA	France	doi: 10.1093/plankt/fbp124	15
Kaggle Plankton Data	Marine plankton	NA	Whole image	NA	Hatfield Marine Science Center	NA	16
Wildfish	Fish	55,000	Whole image	NA	Global	doi: 10.1145/3240508.3240616	17

(Continued)

TABLE 2 Continued

Dataset Name	Subject	Approx. label count	Label type	Label file type	Geographic location	Published reference	URL
Labeled fishes in the wild	Fish	1,000	Bounding box	NA	Southern California Bight	doi: 10.1109/WACVW.2015.11	18
DIDSON Imaging Sonar fish dataset	Fish	1,500	Whole image	NA	Ocqueoc River, Michigan, USA	doi: 10.1038/sdata.2018.190	19
OBSEA EMSO	Fish, underwater scenes	1,200	Whole image	NA	OBSEA-EMSO testing-site	doi: 10.1038/s41598-018-32089-8	20
FishCLEF-2015	Fish	14,000	Semantic segmentation	.xml	NA	doi: 10.1007/978-3-319-24027-5_46	21
UNICT Underwater Background	Underwater scenes	3,500	Semantic segmentation	.xml	NA	doi: 10.1016/j.cviu.2013.12.003	22
SeaCLEF-17 Dataset	Fish, marine animals	NA	Whole image	.xml	Taiwan	NA	23
Japan E-Library of Deep Sea Images	Organisms, geologic features, debris	NA	Whole image	NA	Deep-sea environments	NA	24

1. https://www.kaggle.com/datasets/smaranjitghose/sea-turtle-face-detection?msclkid=2540da87b6dd11eca46690336c5e94aa
2. https://github.com/open-AIMS/ozfish
3. https://swfscdata.nmfs.noaa.gov/labeled-fishes-in-the-wild/
4. https://fathomnet.org/
5. https://github.com/xahidbuffon/SUIM
6. https://data.mendeley.com/datasets/n3ydw29sbz/3
7. https://www.kaggle.com/competitions/the-nature-conservancy-fisheries-monitoring/data
8. https://coralnet.ucsd.edu
9. https://www.imageclef.org/lifeclef/2015/fish
10. https://doi.org/10.13020/x0qn-y082
11. https://doi.org/10.13020/g1gx-y834
12. https://hdl.handle.net/10.1575/1912/7341, https://doi.org/10.4319/lom.2007.5.204
13. https://doi.org/10.1109/CVPR.2012.6247798
14. https://doi.org./10.17632/86y667257h.2
15. https://www.seanoe.org/data/00446/55741/
16. https://www.kaggle.com/c/datasciencebowl
17. https://github.com/PeiqinZhuang/WildFish
18. https://www.st.nmfs.noaa.gov/aiasi/DataSets.html
19. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6176783/
20. https://www.nature.com/articles/s41598-018-32089-8
21. https://link.springer.com/chapter/10.1007/978-3-319-24027-5_46
22. https://tinyurl.com/UNICT-Underwater-Bkg-Modeling
23. https://www.imageclef.org/lifeclef/2017/sea
24. https://www.godac.jamstec.go.jp/jedi/e/index.html

Cells labeled "NA" (not applicable) are not applicable to the corresponding dataset.Note that some databases are actively curated and updated over time. Image and label counts are approximate and current as of October, 2022.

label types the software allows (i.e., whole image labeling, bounding box labels, polygon labels, semantic labels, panoptic labels); and (iv) and labeled data file formats the software is capable of importing and exporting (e.g., Pascal VOC XML, COCO JSON).

3.2.4 Publicly available databases of annotated imagery from the field

In the computer vision literature, large, publicly available labeled image datasets such as ImageNet (14.2 million images; Russakovsky et al., 2015) and COCO (over 320,000 images; Lin et al., 2014) have been pivotal in driving the development of image-based ML methods. These datasets provide researchers with a source of data for quickly testing new model architectures, and for benchmarking and comparing new models using the same data sources. However, perhaps not surprisingly, these datasets contain relatively few images and label classes that are directly relevant to the use cases of interest to most marine scientists (Qin et al., 2016). Over the past decade, a number of curated open source databases containing labeled imagery from marine environments have begun to come online. The largest and most thoroughly curated of these are listed in Table 2. Depending on the specific problem a researcher is interested in addressing, these datasets may provide useful resources for model pre-training (Salman et al., 2016; Orenstein and Beijbom, 2017; Knausgård et al., 2021; Li et al., 2022), or if classes of interest are contained within one or more of these datasets, they may contain sufficient examples to train an initial model that can be deployed on new imagery and fine-tuned with new labels if needed.

3.2.5 Size of training set and balance among classes

An obvious question that arises when creating a training dataset is the question of how many images are required to achieve a desired level of performance. Several recent studies have sought to address this question for the tasks of instance segmentation (Ditria et al., 2020) and whole image classification (Piechaud et al., 2019;

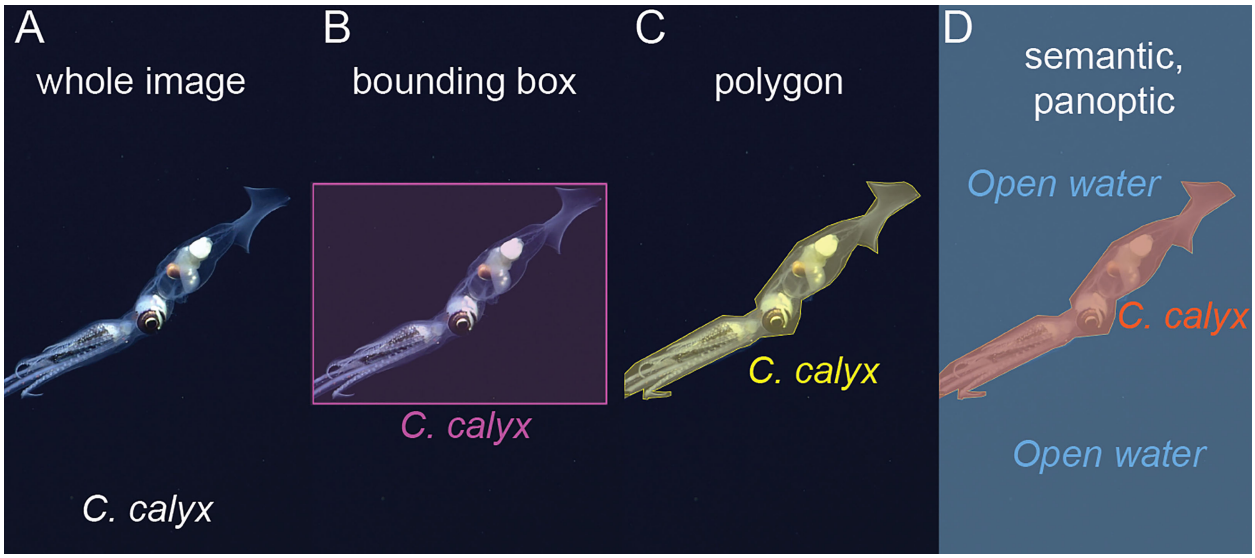


FIGURE 3
Examples of different label types. **(A)** Whole image labels assign a class to the entire image, in this case the squid species that occurs in the image. **(B)** Bounding box labels bound objects of interest within boxes and assign a class to each box. **(C)** Polygon labels bound each object of interest with a polygon and assign a class to each polygon. **(D)** Semantic segmentation assigns a class to each pixel in an image, in this case, pixels are labeled either *"C. calyx"* or *"Open water"* classes. In panoptic segmentation, pixels are assigned a class, and pixels belonging to the same instance of a given class are grouped together. Here, pixels assigned to the *C. calyx* class would be grouped into a single instance.

Villon et al., 2021). In these studies, performance metrics often begin to saturate at around 1,000 and 2,000 labels of a given class, beyond which point adding additional labeled data results in diminishing gains in performance. This saturation of performance around roughly 1,000 labeled instances per class is also consistent with other analyses of ML model performance on field imagery (e.g., Schneider et al., 2020 but see Durden et al., 2021). While the precise number of labels required to provide a desired level of performance is unlikely to follow a hard and fast rule, such numbers do provide ballpark estimates of the number of labeled instances per class one ought to have before expecting high performance from a ML model. It is worth noting, however, that many studies that report saturating performance as label count increases compute these metrics on test sets selected at random from the overall set of images used to train, test, and validate models (e.g., Ditria et al., 2020; Villon et al., 2021). As we will show later, the method of test set construction can have a major impact on measurements of model performance.

In practice, when constructing training sets, several factors are likely to influence the number of training labels available for each class. The first is the time and cost required to manually generate labels. Whole image classification by humans can be reasonably fast (e.g., 5 seconds per image, Villon et al., 2018); instance segmentation tends to be slower (e.g., 13.5 sec per image, Ditria et al., 2020); and more elaborate labeling such as panoptic is slower still (e.g., up to 20 minutes per image, Uijlings et al., 2020). How much time and money might it cost to create a labeled dataset? Assuming the per-image human instance labeling rate reported by Ditria et al. (2020), it would take 3.75 hours to label 1,000 images of a single class, which is not insignificant if objects of many different classes must be labeled. Katija et al. (2022) performed a more detailed valuation of the data contained within the initial release of *FathomNet* and estimated the initially uploaded dataset consisting of approximately 66,000 images to have taken over 2,000 hours of expert annotation time at a cost of roughly $165,000 for the labeling effort alone.

A second factor that influences the size of image datasets has to do with limited availability of images of rare classes. Even relatively large annotated image databases from the field typically contain many classes that are represented by far fewer than 1,000 instances (Schneider et al., 2020). Given the highly skewed distribution of species abundances documented in ecosystems around the world (McGill et al., 2007), it is simply expected that few species will be common, and most species will be far rarer. This distribution of species abundances is likely to result in image sets that contain relatively few training images of most species (Villon et al., 2021). When this is the case, using training routines (e.g., weighted penalization of errors, Schneider et al., 2020; hard negative mining, Walker and Orenstein, 2021) and ML pipelines that enhance performance on rare classes may be the only option. As an example of the latter, Villon et al. (2021) recently showed that, ***few-shot learning*** models can begin to saturate performance with tens of training examples per class rather than the thousand or more required by more conventional ML models. For this reason, development of few-shot learning methods is likely to be an important area of research in the coming years.

3.2.6 Scope of training imagery versus deployment imagery

One common source of underperformance of machine learning methods on new imagery can be traced to the range of conditions and class distributions present in the training set relative to new

datasets on which the model is to be used. A good rule of thumb is to try to create a training dataset that spans the range of conditions you expect to sample when deploying the model on new imagery. For example, if you plan to use an image classifier on shallow water imagery collected from 30 distinct sampling locations across the daylight cycle, to the extent possible, train on imagery that contains the spatial and temporal variation inherent in this target use case. This does not necessarily mean labeling more imagery, but rather, labeling images that span the range of conditions expected when the model is applied to new image data. González et al. (2017) provide a detailed discussion of strategies for building a training and validation routines that yields reliable estimates of the future performance of a trained ML pipeline.

3.3 Case study: bounding box data from the *FathomNet* database with species- and genus-level class labels

As described above, our case study focused on six biological taxa detected in imagery collected in the Monterey Bay and surrounding regions of the coastal eastern Pacific. Images and corresponding labels for the classes used in our case study can be downloaded programmatically from *FathomNet*. Labels are downloadable from *FathomNet* in the widely-used COCO JSON format, which includes object bounding box instances corresponding to each image, along with their classes, and metadata associated with each image. Because we wished to apply a ML model called YOLO that does not accept COCO JSON as an input format, we had to convert labeled data to an admissible input format and create the necessary directory structure. Code to download and convert images and organize directories is provided at https://github.com/heinsense2/AIO_CaseStudy.

4 Selecting and training a machine learning model

4.1 Overview

After specifying an image analysis task and building a training dataset, the next step is identifying a particular machine learning model to train and test. Here, we are focused primarily on modern computer vision methods for automated analysis, many of which rely on deep learning – learning algorithms that involve the use of ***deep neural networks*** (DNNs). Deep learning is a form of representation learning, in which the objective is not only to use input data (e.g., an image) to make predictions (e.g., the class to which the image belongs), but also to discover efficient ways to represent the input data that make it easier to make accurate predictions (Bengio et al., 2013). Deep learning models are representation learning algorithms that teach themselves which features of an image are important for making predictions about the image. By training on a set of labeled images, these algorithms learn a mapping between raw pixel values and the desired output based on these features.

4.2 Technical considerations

Foundational work in deep learning demonstrated that networks that are good at representing features useful for prediction often share common structural features (LeCun et al., 2015), and this idea has fueled the use of deep neural networks with convolutional structure (Convolutional Neural Networks or CNNs), network pre-training, and other practices that help ensure that networks can quickly be trained to perform a target task on a new dataset, rather than having to be fully re-designed and trained *de novo* for each new application.

4.2.1 Selecting a machine learning model

The field of DNN-based models capable of performing image classification, object detection, and semantic segmentation is enormous, and expanding by the day. Table 3 provides a list of models that have shown promising results on imagery collected from either marine environments, or terrestrial environments that present similar challenges to those frequently encountered in marine environments (e.g., complex backgrounds, heterogeneous lighting, variable image quality, *etc.*) that is up to date as of this publication. Benchmarking sites (e.g., https://paperswithcode.com/sota/object-detection-on-coco) are another useful resources for tracking the most recent high-performing models on standard computer vision tasks.

In a practical sense, choosing which ML model to use in any particular setting involves first determining which models can perform the target task (e.g., whole image classification vs. semantic segmentation). For any given target task, there will be many available models to choose from. We recommend researchers consider three things when choosing from among these models: (i) have previous studies evaluated and compared model performance? Has any study been done that applied a particular model in a similar setting with favorable performance? (ii) Is open-source code or a GUI-based implementation of the model available? If so, how easy does it appear to be to implement? Is it compatible with the computational hardware you have available? (iii) How many additional packages, software updates, and other back-end steps are required to be able to train and deploy a given model using new data? In our experience, perhaps *the* major hurdle associated with applying any given ML model to a new dataset is the time required to configure the software and system specifications necessary to run the model code. This "implementation effort" may ultimately dictate which model an end user ultimately selects. If a given ML model has been shown to exhibit good performance, but implementing that model requires significant knowledge of command-line interfaces, software package installers or dependencies, virtual environment management, hardware compatibility, or GPU programming, it may simply require too much invested time at the outset to be a viable option for most researchers.

4.2.2 Hardware implementation: CPU vs. GPU, local vs. cloud

Another decision a user must make when implementing ML pipelines is whether to run the computations involved in training, testing, and deploying the model on a computer's central processing unit (CPU) or on the computer's graphics processing unit (GPU). Among the technological developments that enabled widespread use of DNN models is software and hardware innovations that allow these models to be trained rapidly and in parallel using GPUs. The technical details of ML implementations on these two distinct types of hardware are discussed in Goodfellow et al. (2016) and Buber and Diri (2018). The advantage of training using a CPU is that any computer can, in principle, be used to perform training without the need for specialized hardware that some computers have and others lack. The disadvantage is that, in the absence of custom parallelization, training a DNN model of any depth using CPUs can be prohibitively slow. Fortunately, many consumer-grade workstations now ship with GPUs that are compatible with deep learning frameworks like PyTorch (Paszke et al., 2019) and Tensorflow (Abadi et al., 2016), and many universities and research institutes are investing in shared GPU clusters. Another option for accessing machines capable of training ML models is through cloud computing services such as Google Colab, Amazon Web Services, Microsoft Azure, and others. Free cloud services maybe a good option for researchers seeking to perform small pilot studies of ML model performance on their own datasets. Paid cloud services may be a particularly good option for researchers who wish to have access to many GPUs or powerful GPUs for relatively short periods of time, but who do not need or wish to manage their own local computing hardware.

4.3 Case study: object detection and classification with YOLO

Our case study task involves detecting objects of interest, along with a bounding box and class label for each object. We selected one of the most widely used object detection and classification pipelines, YOLO ("You-Only-Look-Once", Redmon et al., 2016). YOLO is heavily used in industry and research applications, has fast deployment times relative to other deep architectures, and is relatively easy to use. Moreover, various versions of YOLO have

TABLE 3 Machine learning models applied to analysis of field imagery.

Model	Study	Task type	Application	Performance measures	Test set construction (in-domain vs. out-of-domain)	Code provided?
Mask R-CNN	Ditria et al., 2020	instance segmentation, classification	Identify and segment single fish species in seagrass meadows	F1 scores, mAP50	in-domain, out-of-domain	no
DeepMac	Beery et al., 2021	instance segmentation, classification	Instance segmentation from terrestrial camera traps	mAP, mean RMSE, RSSE	not reported	no
SOLO (v1, v2)	Lv et al., 2021	instance segmentation, classification, panoptic segmentation	Instance segmentation of camouflaged animals (terrestrial and aquatic).	Mean absolute error, root mean absolute error	not reported	yes, 1
R-CNN	Salman et al., 2020	bounding box detection, classification	Fish detection in a variety of field settings (e.g. crowded, dynamic background)	Average F1 score	in-domain	yes, 2
Fast R-CNN	Chegini et al., 2022	bounding box detection, classification	Detection and instance segmentation of weeds.	mAP, precision, recall, F1 score	in-domain	no, some pseudocode provided
YOLO	Jalal et al., 2020; Yusup et al., 2020	bounding box detection, classification, instance segmentation	Fish detection and classification in images and video	Accuracy	in-domain	yes, 3
Megadetector	Beery et al., 2021	bounding box detection, coarse classification	Object detection in terrestrial camera traps	mAP, RMSE, RSSE	not reported	yes, 4
Ensemble Vision Transformer	Kyathanahally et al., 2022	whole image classification	Whole image classification in several field imagery datasets, compared several DNNs/ ensembles	Reduction in error relative to other	Varies by dataset, mostly in-domain or k-fold in-domain	no

(Continued)

TABLE 3 Continued

Model	Study	Task type	Application	Performance measures	Test set construction (in-domain vs. out-of-domain)	Code provided?
				classification methods		
Densenet 169 Convnet Ensemble	Wyatt et al., 2022	whole image classification	Whole image classification from coral thumbnails	Data-shifting accuracy using Expected Calibration Error	in-domain, out-of-domain	yes, 5
RetinaNet, YOLO v5	Katija et al., 2022	bounding box detection, classification	Object detection, classification of many class types in diverse benthic imagery	Accuracy, confusion matrix	in-domain, out-of-domain	yes, 6
Inception v3	Allken et al., 2019	whole image classification	Species classification for trawl surveys	Accuracy	in-domain	no
AlexNet	Jaüger et al., 2015.	whole image classification	Spcecies identification of fish from thumbnails	Accuracy, mAP	in-domain	no
GoogLeNet	Villon et al., 2018	whole image classification	Fish species classification from underwater thumbnail images	Accuracy	in-domain	no
CNN-SENet	Knausgård et al., 2021	bounding box detection, classification	Temperate fish detection, classification, compared several DNNs	Accuracy	in-domain	no
Conv. GANs	Zhao et al., 2018	whole image classification	Live fish identification in aquaculture	Accuracy	in-domain	yes, 7

1. https://github.com/aim-uofa/AdelaiDet/
2. https://github.com/ahsan856jalal/Fish-Abundance
3. https://github.com/ahsan856jalal/Fish-detection-and-classification-using-HOGY.git
4. https://github.com/microsoft/CameraTraps/blob/main/megadetector.md
5. https://doi.org/10.5281/zenodo.6317553
6. https://github.com/fathomnet/models
7. https://github.com/Zhaojian123/Transactions-of-the-ASABE

A selection of past models used to perform image analysis tasks on field imagery. Performance measures reported describes which performance measures were reported for test sets in each study. Test set construction describes whether the statistics reported were computed using a test set derived from the same overall dataset used to train the model ("in-domain"), or whether the test set was deliberately constructed using data from new spatial or temporal regions ("out-of-domain"). The Code provided column indicates whether the study provided the code used in their analyses.

been incorporated into more complex detection and classification pipelines that have shown promising results on marine imagery (e.g., Knausgård et al., 2021; Peña et al., 2021). For all analyses, we used initial weights provided in YOLO v5 from pre-training on the COCO dataset (https://github.com/ultralytics/yolov5). We selected the "small" network size as a compromise between network flexibility and the number of network weights that need to be estimated during training. Prior to training and testing, we reduced the resolution of images to 640 x 640 px (the impact of changing resolution is evaluated below). We included the five classes of squid and fish in our primary analysis, and reserved images of the siphonophore, *N. bijuga*, for a later analysis (see "*Distractor classes*" below).

We benchmarked training and deployment of YOLO v5 using both in-house hardware (a single workstation with four GPUs), and a cloud-based implementation. For the local hardware implementation, we used a Lambda Labs Quad workstation running Ubuntu 18.04.5 LTS and equipped with four NVIDIA GeForce RTX 2080 Ti/PCIe/SSE2 GPUs, each with 11,264 MB of memory. The machine also had a 24 Intel Core i9-7920X CPUs @2.90GHz with 125.5GiB of memory. Our cloud implementation used Google Colab (https://colab.research.google.com), a cloud-based platform for organizing and executing Python programs using code notebooks (termed "Colab Notebooks"). Our cloud implementation made use of these resources using a Google Compute Engine backend with a single NVIDIA K80/Tesla T4 GPU with 16 GB of memory. In both local and cloud implementations, all models were trained for 300 epochs (or for fewer epochs when early stopping conditions were met) using all available GPUs. Run times on our local and cloud implementations were comparable, with the 4 GPU local machine performing slightly faster (mean of 19.1 sec per training epoch; 1.59 hours to complete 300 epochs) than the single GPU cloud implementation (mean of 26.8 sec per training epoch; 2.23 hours to complete 300 epochs). System specifications, software versions, training settings and all other details required to repeat our analyses are described in the accompanying code tutorial at https://github.com/heinsense2/AIO_CaseStudy.

5 Evaluating model performance

5.1 Overview

After training models, a final step in the model building process is to evaluate model performance. Many metrics are available for

measuring the performance of ML models, and the most appropriate metric in any given application will depend both on the task the model is trained to execute (e.g., image classification vs. semantic segmentation), and the relative importance of different kinds of errors the model can make (e.g., false positives vs. false negatives), which must, of course, be determined by the researcher. Goodwin et al (2021) and Li et al (2022) provide approachable discussions of common metrics, along with formulae for computing them and the logic that underlies them. Tharwat (2020) provides a more technical account of classification metrics and their strengths and weaknesses. In very general terms, one typically wishes to evaluate the ability of the ML model to predict the correct class of an object, image, or subregion of the image, and, if the method provides spatial predictions about objects or semantic classes located in different parts of the image, one would like to know how accurate these spatial predictions are.

5.2 Technical considerations

For whole image classification, performance metrics seek to express the tendency of the model to make different kinds of errors when predicting classes. For example, suppose a researcher has 300 sea surface satellite images, and a model is trained to determine which images contain harmful algal blooms (HABs) and which do not (Henrichs et al., 2021). The ***classification accuracy*** of the model is the ratio of images that were assigned the correct class (HAB present vs. HAB absent) over the total number of images classified: (true positives + true negatives)/(total images classified). If the model correctly predicted 100 images that contained HABs, and correctly predicted 100 images that did not contain HABs, the accuracy is 200/300 = 0.67. Accuracy is an appealing measure because of its simplicity but it can be misleading, particularly when the dataset contains multiple classes and the relative frequency of classes differs (see discussion in Tharwat, 2020). Other widely-used metrics including precision, recall, and F1 score, were designed to capture other aspects of model performance, while avoiding some of the biases of classification accuracy. The ***precision*** of a classifier measures the fraction of positive class predictions that are correct. If the model classifies 130 images as containing HABs and 100 of these images actually contain HABs, the precision of the classifier is 100/130 = 0.77. ***Recall,*** sometimes also referred to as "sensitivity," measures the ability of a model to detect all images or instances of a given class that are present in the dataset, thereby expressing how sensitive the model is to the presence of a class. If the classifier correctly classifies 100 images containing HABs but the dataset contains 160 images that contain HABs, the model's recall is 100/160 = 0.63. The ***F1 score*** provides a composite performance measure that incorporates both precision and recall: *F1 = 2 (precision x recall)/(precision + recall)*.

For methods that make spatial predictions, there is an additional question of whether the model's spatial predictions are located in the right place. Among the most widely-used methods for measuring the spatial overlap between predictions and data this involves computing the spatial overlap between a prediction from the model and objects in the labeled image. This is often measured using the ***intersection-over-union*** (IoU): the intersection area of the predicted borders or bounding box of an object and the borders or bounding box of the label, divided by the total number of unique pixels covered by the bounding box and the label. Pairs in which the predictions precisely overlap labels will have equal intersection and union, giving an IoU value of one. Complete mismatches, partial spatial matches, or cases where the predicted and labeled bounding boxes differ in size will result in a union that exceeds the intersection and an IoU value less than one, with a minimum of zero when there is no overlap between predicted and observed bounding boxes.

In object detection and classification tasks, the added complication of predictions being spatial raises some questions about how one ought to compute the accuracy of class predictions. A standard practice is to consider a given bounding box a valid "prediction" if its IoU value exceeds some pre-specified threshold, which is often set arbitrarily at 0.5. For bounding box-ground truth pairs exceeding this threshold, one then evaluates performance using one or more of the same metrics applied in whole image classification (e.g., accuracy, precision, recall, F1 score, etc.). A widely-used metric is the ***mean average precision*** (mAP), which is most commonly calculated from the precision-recall curve as the average precision of model predictions over a set of evenly spaced recall values (Everingham et al, 2010), where the precision-recall curve represents model precision as a function of model recall across a range of values of a threshold parameter. The thresholds most often used are the box or instance confidence score and the IoU of predicted and labeled object detections. By default, YOLO v5 produces two measures of mean average precision: mAP@0.5, which is the mean average precision of the model assuming matches constitute all prediction-ground truth pairs with IoU >= 0.5, and a second measure, mAP@0.5:0.95, which is the arithmetic mean of average precision of the model computed across a range of threshold IoU values in the set, {0.50, 0.55,0.60,…,0.90, 0.95}. Different studies and machine learning software implementations compute mAP slightly differently, so ensuring that you understand how it is being computed is important when comparing predictions across studies or ML methods.

5.2.1 Cross validation and performance evaluation

When evaluating the performance of a model on test images held out during training, the exact values of performance metrics will depend on the particular subset of images used during testing. Because training, validation, and testing image sets are typically selected at random from the overall image set, random variability in exactly which images end up in training, validation, and test sets will invariably introduce stochasticity in performance estimates. One way to address this is to create several or even many random subsets of the overall image dataset into training, validation, and test sets. This is sometimes referred to as ***k-fold cross validation***, where *k* denotes the number of training/validation/test splits included in the analysis. The objective of this type of cross validation is to provide more robust measures of performance by averaging over multiple

random partitions of the data into training, validation, and testing sets.

5.2.2 Non-random partitioning and "out-of-domain" performance

In addition to cross validation using random partitions of the data, it is also becoming more common to evaluate model performance on non-random partitions of data into training/validation and test datasets (Schneider et al., 2020; Taori et al., 2020). Typically, this is done to produce test sets that are more representative of new data on which the ML pipeline is intended to be used. For example, if one wishes to train an image classifier to classify coral species from images (Wyatt et al., 2022), and this classifier is intended to be used at new locations in the future, one way to test its performance would be to divide the annotated imagery available into distinct spatial locations, and to construct the training and validation set from a subset of those locations, while holding out other locations that the model never sees during training. This type of model evaluation seeks to determine whether models are capable of performing well on images that may have very different statistics than the images on which they were trained. We will come back to this issue in the following section.

5.3 Case study: performance on object detection and classification of underwater imagery

In-domain performance on test imagery. Images of our target classes in *FathomNet* were collected at many different physical locations, and over decades of sampling (32 years spanning 1989-2021) using remotely operated vehicles equipped with a range of different types of imaging equipment. This led to an image set with complex and diverse backgrounds, highly variable visual conditions, and a wide range of image statistics (Figure 2) – characteristics that we expect will also be typical of medium- to long-term image datasets collected from other locations. Despite this variability, after training YOLO v5, we were able to achieve high object detection and classification performance on test imagery selected at random from the same spatial region or temporal period used to build the training set (Table 4 "in domain"). Mean average precision (mAP) of model predictions ranged from 0.67-0.95, and three classes had mAP values of 0.88 or above. Model F1 scores had an average value of 0.77, and three classes had F1 scores of 0.81-0.92. To put these performance metrics in context, Ditria et al. (2020) quantified the ability of citizen scientists and human experts to detect and classify a fish species (*Girella tricuspidata*) in images taken from shallow-water seagrass beds in Queensland, Australia. Citizen scientists and experts had mean F1 scores of 0.82 and 0.88, respectively. Comparing performance of YOLO v5 on our dataset to these benchmarks implies that our detection and classification results are in the same range as those of human annotators on a similar task.

5.3.1 Out-of-domain performance: evidence for distribution shifts

As noted above, many researchers who wish to use machine learning pipelines to analyze imagery from the field often intend to use trained pipelines to analyze *new* imagery taken at later dates or different physical locations, rather than focusing solely on images taken from the same database used to construct the training set (Beery et al., 2018; Wyatt et al., 2022). To simulate this scenario, we performed a nonrandom, four-fold cross-validation procedure on the overall set of annotated imagery available on our classes of interest in *FathomNet*. This involved two different kinds of nonrandom partitioning of the dataset. The first was a temporal partition, in which we divided all annotated images of our focal classes into images collected prior to 2012, and images collected from 2012 through the present. This partitioning resulted in *pre-2012* and *post-2012* (2012 onward) image subsets. Splitting the data

TABLE 4 Average performance of YOLO v5 object detection and classification on images selected at random from the same spatial or temporal partition used to build the training set ("in-domain"), or the partition held out ("out-of-domain").

	in domain				out of domain			
Class	p	r	mAP	F1	p	r	mAP	F1
Average	0.76	0.79	**0.81**	**0.77**	0.64	0.61	**0.62**	**0.62**
C. calyx (1)	0.90	0.94	0.95	0.92	0.81	0.87	0.89	0.84
D. gigas (2)	0.60	0.64	0.67	0.62	0.62	0.59	0.64	0.60
G onyx (3)	0.80	0.90	***0.89***	**0.85**	0.51	0.37	***0.38***	**0.43**
Sebastes (4)	0.71	0.63	***0.67***	**0.67**	0.53	0.53	***0.49***	**0.53**
Sebastolobus (5)	0.81	0.82	***0.88***	**0.81**	0.70	0.67	***0.71***	**0.69**
Squid (1-3)	0.88	0.92	0.94	0.90	0.78	0.85	0.85	0.81
Fishes (4-5)	0.80	0.77	0.84	0.78	0.77	0.72	0.77	0.74

Note near universal decrease in all performance measures in out-of-domain data consistent with distribution shifts across spatial and temporal partitions. Metrics reported are precision (p), recall (r), mean average precision (mAP), and F1 score (F1). Drops in mAP and F1 between in-domain and out-of-domain sets of greater than 0.10 are bolded. "Squid" and "Fish" rows give results for class coarsening experiment (see "*Class Coarsening*" in text), where species and genus-level classes are aggregated into coarser classes, fishes (*Sebastes* and *Sebastolobus*) and squid (*C. calyx, D. gigas*, and *G. onyx*). Note mAP and F1 scores on "Fishes" class in out-of-domain data exceeds performance on either of the individual fish genera, indicating an overall enhancement in performance through class aggregation.

at 2012 yielded a similar number of labeled instances for most classes before and after the split. The second partition we performed was a spatial partition. Images from all sampling dates were pooled together. But for each class, we divided images either by depth or by latitude and longitude to ensure that images of each class were divided into distinct spatial "regions," defined arbitrarily as *region 1* and *region 2*. This temporal and spatial partitioning resulted in a four-fold partition of the data: two temporal sampling periods, and two spatial regions. We measured average performance over the four data partitions by training on one of the partitions and testing on the other (e.g., training on *pre-2012* images and testing on *post-2012* images).

Figure 4 and Table 4 shows the results of this analysis. Performance metrics were generally lower in the out-of-domain partition than in the partition from which training data were drawn (general trend of decreasing performance evident in Figure 4). This decrease in performance was particularly extreme for certain classes. For example, average mAP and F1 scores for the black-eyed squid, *Gonatus onyx*, were cut approximately in half – from 0.89 and 0.85, respectively, to 0.38 and 0.43 – when a model trained on one partition was deployed on the other. As previously suggested (Katija et al., 2022), these findings imply that ***distribution shift*** occur in the *FathomNet* dataset, and that these shifts can significantly degrade performance when models are trained on data from one set of locations or time periods and deployed on imagery from new locations or time periods. This phenomenon appears to be widespread in imagery collected from the field (Schneider et al., 2020; Wyatt et al., 2022).

6 Diagnosing and improving model performance on new imagery

6.1 Overview

Although ML-based frameworks have shown impressive classification performance on imagery from marine systems (e.g., Kyathanahally et al., 2022), inevitably, all models make errors. Moreover, the degree to which a previously trained model makes errors when applied to new image datasets can change over time as new imagery changes relative to the original dataset used to perform training. Therefore, one key step in building and maintaining a ML

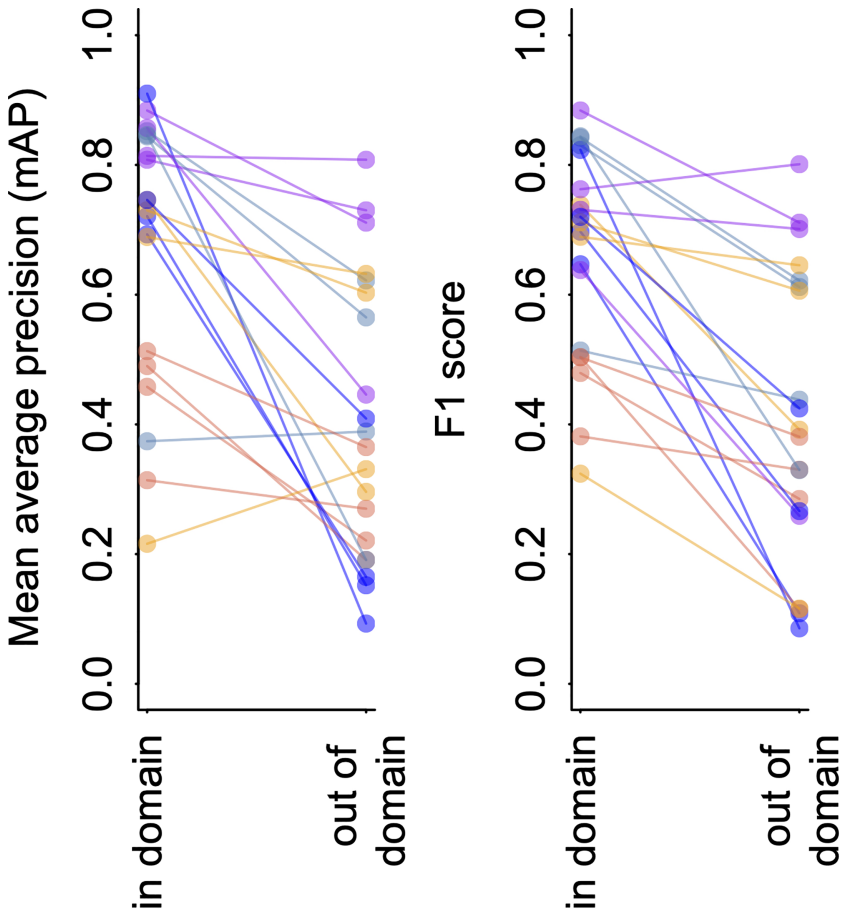


FIGURE 4
YOLO v5 model performance on imagery from *FathomNet*. Change in mean average precision (left) and F1 score (right) when a model is tested using out of sample data from the same spatial or temporal partition from which training data was selected ("in-domain"), and when the same model is tested using data from a different spatial or temporal partition ("out-of-domain"). Colors indicate different classes, and lines connect points from the same spatial or temporal partitioning of the data to indicate trends.

pipeline for automated image analysis is diagnosing performance problems and finding ways to fix them (Norouzzadeh et al., 2018). In this section, we address issues that can degrade performance of a ML pipeline, and suggest approaches for remedying these issues. Many such issues can be traced back to the problem of distribution shift (Beery et al., 2018; Schneider et al., 2020; Taori et al., 2020). A distribution shift occurs when the imagery on which a ML pipeline is trained differs in some systematic way from the imagery on which the pipeline is deployed – that is, the new imagery the ML method is being used to analyze. The term "distribution shift" refers to a generic set of differences that may occur between one set of images (the "in-domain" set) and another (the "out-of-domain" set), including things like differences in lighting, camera attributes, image scene statistics, background clutter, turbidity, and the relative abundances and appearances of different classes of objects (Taori et al., 2020; Scoulding et al., 2022; Wyatt et al., 2022).

Many existing ML methods perform poorly under distribution shifts without careful training interventions (Beery et al., 2018; Schneider et al., 2020; Taori et al., 2020). Despite this, human labelers exhibit similar performance on original and distribution shifted datasets (Shankar et al., 2020), suggesting that distribution shifts do not reduce the information needed to accurately identify objects *per se*, but rather that the structure and training of ML models cause them to fail on distribution shifted imagery (Taori et al., 2020). Given that distribution shifts are documented here (Figure 4, Table 4), and in past studies of imagery from the field (e.g., Beery et al., 2018; Schneider et al., 2020; Katija et al., 2022), a natural question is whether there are steps that can be taken to reduce their effects on model performance.

6.2 Technical considerations

A wide array of methods have been proposed to improve the performance of ML models on new imagery that is distribution shifted relative to training images. These range from training interventions like digitally altering (*i.e.*, *"augmenting"*) training imagery to destroy irrelevant features that can result in overtraining (Bloice et al., 2019; Buslaev et al, 2020; Zoph et al., 2020), to the use of more robust inference frameworks such as ensemble models, which combine predictions of multiple machine learning models (Wyatt et al., 2022). To provide a sense for how some of these methods work, we applied a suite of training interventions to our case study dataset.

Image augmentation is a widely used method for improving model performance on out-of-sample and out-of-domain imagery (Bloice et al., 2019; Buslaev et al, 2020; Zoph et al., 2020). Image augmentation involves applying random digital alterations of training imagery during the training process to help avoid over-fitting ML models to specific nuances of training imagery that are not useful for identifying objects of interest in general. Augmentation of training images is used by default in many ML pipelines (including YOLO v5) as part of the training process, but augmentation parameters are often tunable, so having some understanding of how different types of augmentation affect performance on field imagery is useful.

Increasing ***image resolution*** is another straightforward training intervention. Due to the computational and memory demands of training DNN-based ML models, it is common to reduce image resolution during training, testing, and deployment (e.g., Kyathanahally et al., 2022). However, if objects of interest constitute relatively small regions of the overall image (e.g., Figure 2), reducing resolution can coarsen or destroy object features that can be important for detection and classification. The loss or degradation of these features during training and deployment, mean that they cannot be used to accurately detect and classify objects in new imagery that is distribution shifted relative to the training set. It may, therefore, be beneficial in some applications to maintain higher image resolutions during training and deployment.

Training using ***background imagery*** is another intervention that is relatively easy to implement. While it can be costly to label new imagery for the reasons discussed above, it can be relatively cheap to identify "background images," defined simply as images that do not contain objects of interest. Training a ML model by deliberately including background imagery in the training set has been proposed as one method for helping models to better generalize to new image sets (Villon et al., 2018).

A fourth type of intervention is known as ***class coarsening***. Intuitively, objects that are visually similar are likely to be harder to discriminate than are objects that look very different. Given this, one potential solution to improve model predictions under distribution shifts is to coarsen class labels in a way that results in similar looking classes being aggregated into a single super-class (Williams et al., 2019; Katija et al., 2022). In biological applications, this may result in aggregating classes with finer phylogenetic resolution (e.g., species or genus-level classes) into classes with coarser resolution (e.g., family or order-level classes or coarse species groups). For instance, rather than requesting individual species of sea fan and corals, one might simply specify "sea fans" and "corals" as classes (but see Howell et al., 2019 for a discussion of the need to aggregate with care). Whether this is a suitable training intervention obviously depends on the ultimate goal of the image analysis and whether coarser class labels are acceptable.

A final intervention we consider is training on images that include objects in distractor classes. The definition of the term "distractor" in the computer vision literature has varied (e.g., see Das et al., 2021 vs. Zhu et al., 2018). Here, we define a distractor class as a class of object that shares visual characteristics with a target class and could reasonably be confused with the target class during classification. This working definition is consistent with the way the term "distractor" is used in the visual neuroscience literature (e.g., Bichot and Schall, 1999). When training ML models to detect a certain class or small set of classes, it is common to train models using labels of only the class or classes of interest. However, if distractor classes are regularly present in new imagery, they can degrade model performance. Deliberately including images of distractor classes in the training set is a form of adversarial training that may improve model performance when distractor classes occur in new imagery.

In addition to these simple training interventions, a variety of other solutions to improve model robustness on new imagery have

been proposed. These include the use of ensemble models, where predictions are derived not from just one deep neural network, but from many networks whose predictions are combined to make an overall class prediction (Wyatt et al., 2022), adversarial training, sometimes also called "active learning," in which models are re-trained with images on which they previously made errors (Mathis et al., 2020), training on synthetic data (Schneider et al., 2020), and stratified training in which the relative abundance of classes in the training set are modified by excluding or including extra examples of one class or another (Schneider et al., 2020). There are related methods that seek instead to analyze the output of automated systems at the sample level, rather than the individual level, to correct errors and detect changes in new domains (González et al., 2019; Walker and Orenstein, 2021). We refer the reader to the research cited in this section, and to Taori et al. (2020); Schneider et al (2020) and Koh et al. (2021) for further reading on methods for improving performance under distribution shifts.

6.3 Case study: training interventions and performance on out-of-domain imagery

6.3.1 Image augmentation

To test whether and how augmentations might improve model performance on new imagery, we applied three kinds of augmentation to images during training: orientation augmentations, in which the training image and corresponding bounding box is scaled or flipped by a random amount, color space augmentations, in which the color attributes of the training image are randomly perturbed during training, and mosaic augmentation, in which sets of training images from the training set are randomly selected, cropped, and recombined to form a new composite "mosaic" image used in training. We tested the impact of each of these augmentation types by starting with all of them active, then dropping one augmentation type at a time. For each of these augmentation "treatments," we computed performance metrics on the out-of-domain testing set, averaging over all four partitions of the data. Augmentation parameters and parameter values are defined in the case study code accompanying this manuscript.

Applying no augmentations at all resulted in the poorest performance (Table 5), whereas the best performance occurred when all augmentations were applied. However, the effects of augmentations were highly variable among different partitions of the data, and among classes. These results suggest that augmentation may indeed be a way to improve generalization on new imagery, but that effect of augmentation may differ from one class to another. We did not observe a systematic decrease in performance under any augmentation scheme. However, Tan et al. (2022) recently reported such decreases in performance in the context of marine benthic imagery, emphasizing that it is important to choose augmentation routines with care.

6.3.2 Image resolution

To explore the impact of changing image resolution in our case study, we modified the default resolution specified in YOLO v5 (640 px x 640 px) to a higher resolution (1280 px x 1280 px). Between 91% and 98% of images available in *FathomNet* for each class have a resolution equal to or greater than 640 pixels along at least one axis. Images with resolution lower than 1280 x 1280 were loaded at full resolution and padded at the borders to reach the desired training resolution. Effects of increased image resolution were not large. For example, the average change in mean average precision on out-of-domain data across the four partitions was 0.04, and the largest performance increase was only 0.05 (for *G. onyx*), while performance on *C. calyx* actually dropped slightly when we used higher resolution imagery. It is worth noting that differences in resolution between training and testing data can cause degraded performance (Recht et al., 2019), which may have contributed to a lack of improvement in performance in some of the partitions (e.g., *pre-2012* vs. *post-2012* splits, for which image resolution systematically differed).

6.3.3 Training on background imagery

To test whether training on background imagery could improve out-of-domain performance, we re-trained YOLO v5 using the *post-2012* partition as a training set, but we also included background images from the *pre-2012* and *post-2012* partitions in the training imagery. Including background imagery improved performance on all classes (Table 6), with the largest increases in performance for *Dosidicus gigas* and *Gonatus onyx*, the classes with the fewest labels in the training set (n = 42, and n = 84 labeled instances, respectively).

6.3.4 Class coarsening

To explore whether class coarsening improved performance under distribution shifts, we coarsened class labels from the species

TABLE 5 Effect of image augmentation on performance of YOLO v5 on out of domain set.

Augmentation type	p	r	mAP	F1
No augmentation	0.38-0.62	0.17-0.72	0.20-0.67	0.22-0.62
No mosaic	0.47-0.79	0.37-0.84	0.34-0.85	0.36-0.81
No orientation	0.51-0.77	0.34-0.84	0.36-0.84	0.39-0.8
No color space	0.54-0.84	0.36-0.86	0.41-0.89	0.42-0.85
All augmentations	0.54-0.83	0.41-0.88	0.45-0.90	0.47-0.85

Metrics reported are precision (p), recall (r), mean average precision (mAP), and F1 score (F1). Each cell reports the range of values across classes after averaging performance of each class over spatial and temporal partitions. "No augmentation" used only raw training images to train model. "No mosaic" used orientation augmentations and color space augmentations only. "No orientation" used color space and mosaic augmentations only. "No color space" used mosaic and orientation augmentations only, and "All" used mosaic, orientation, and color space augmentations. A description of these augmentation types is given in the text, and specifics of implementation in YOLO v5 and parameter values are provided in the case study code: https://github.com/heinsense2/AIO_CaseStudy.

(*Gonatus onyx*, *Chiroteuthis calyx*, and *Dosidicus gigas*) and genus level (*Sebastes* and *Sebastolobus*) to the coarse categories of *squids* and *fishes*. Table 4 shows performance of YOLO v5 when trained and tested on these coarser classes. As expected, coarsening classes resulted in a smaller average drop in model performance when models were applied to out-of-domain data. For the *squid* class, out-of-domain performance was higher than for any individual class in the fine class model except for *C. calyx* (class for which the model had the highest performance). Out-of-domain performance for the *fish* class was higher than performance on either of the individual fish genera in the analysis where genera were treated as separate classes.

6.3.5 Training with distractor classes

To quantify the impact of training with distractor classes on model performance, we restricted our analysis to two classes: the swordtail squid, *Chiroteuthis calyx*, and the siphonophore, *Nanomia bijuga*. In particular, we sought to determine whether a trained ML model could discriminate images of juvenile swordtail squid in an image set containing images of juvenile *C. calyx* and imagery of *N. bijuga*, a distractor class that is a mimicked both morphologically and behaviorally by juvenile *C. calyx* (Burford et al., 2015). Because the spatial distributions and habitat use of these two species overlap, a researcher interested in *C. calyx* would likely need to contend with images containing *N. bijuga*, either by itself or in the same image as the target class *C. calyx* (e.g. as in Figure 1F). A naïve approach for training a ML model to detect juvenile *C. calyx*, would be to train only on images of this target class, and then to deploy the model on new images containing one or both of the two classes.

Table 7 shows performance of YOLO v5 trained to detect juvenile *C. calyx* using this naïve approach. Mean average precision is relatively poor as are precision and recall scores (e.g., mAP = 0.54). Moreover, 22% of the instances of *N. bijuga* in test data were erroneously classified as *C. calyx*, indicating that the model often mistook the distractor class for the target class. To determine whether training on both the target and distractor class could help remedy this issue, we re-trained YOLO v5 with a training set containing labeled imagery of both juvenile *C. calyx* and *N. bijuga*. We then applied this model to test imagery. Training on both classes resulted in a pronounced increase in all performance metrics to levels that match or exceed reported performance of human labelers in similar tasks (precision = 0.86, recall = 0.9, mAP = 0.87). Moreover, despite the strong morphological resemblance between *N. bijuga* and juvenile *C. calyx* (Figure 1F), the model trained on both classes never classified new images of *C. calyx as N. bijuga* or vice versa (0% misclassification rate). A third approach to improving model performance in the presence of distractor classes that is less costly than manually labeling distractor classes is to include in the training set images that contain the distractor, but to treat these as unlabeled "background" imagery. That is, if an image contains only the distractor class, it would be included in the training set with no instance labels. To test this approach, we used the same images of *C. calyx* and *N. bijuga* used to train the two-class model, but we included no labels for the *N. bijuga* class. Performance using this approach was only slightly lower than performance of the model trained on labels of the distractor (Table 7), indicating that such training be a viable alternative to building a full dataset containing labels for distractors as well as the target class.

6.3.6 Summary of training interventions and their effects on performance on new imagery

The image set and number of classes used in our case study was intentionally limited, so our findings should also be taken with this in mind. Overall, we found that image augmentation (improvement in mAP of 0.18-0.25, F1 of 0.04-0.25), and class coarsening (average improvement in mAP of 0.21-0.25, F1 of 0.13-0.19) provided improvements in performance on new imagery in all or most classes in the dataset. Training distractors also resulted in large improvements in performance for the target class used in the distractor analysis (improvement in mAP of 0.25-0.33, F1 of 0.22-0.33). The impact of training on background imagery was more variable, but still resulted in overall improvements in performance for most classes (improvement in mAP of 0-0.08, F1 of 0.01-0.11). Training and deploying the model on high-resolution imagery (as

TABLE 6 Effect of including background imagery on performance of YOLO v5 on out of domain imagery (temporal partitions).

		No BG images				BG images			
class	num. labels in training set	p	r	mAP	F1	p	r	mAP	F1
C. calyx	351	0.84	0.88	0.90	0.86	0.87	0.88	0.90	**0.88**
D. Gigas	229	0.48	0.46	0.51	0.47	0.57	0.60	**0.59**	**0.58**
G. onyx	94	0.66	0.53	0.57	0.59	0.76	0.51	**0.60**	**0.61**
Sebastes	445	0.67	0.54	0.61	0.60	0.65	0.60	**0.63**	**0.62**
Sebastolobus	1178	0.74	0.82	0.83	0.78	0.76	0.82	**0.84**	**0.79**

"No BG images" shows performance of standard training in which no background images are included in the training set. "BG images" shows statistics for training runs in which background images from pre-2012 and post-2012 periods that did not contain any classes of interest were included in the training set. Metrics reported are precision (p), recall (r), mean average precision (mAP), and F1 score (F1). Bolded mAP and F1 score values in "BG images" show cases where these statistics improved relative to training without background images. Results from the two temporal partitions are averaged.

TABLE 7 Effects of distractor class on model performance.

	p	r	mAP	F1	Misclassification frequency
Train without distractor	0.49	0.68	0.54	0.56	0.22
Train with distractor as background	0.88	0.70	0.79	0.78	0.008
Train with distractor labels	0.86	0.90	0.87	0.89	0

Precision (p), recall (r), mean average precision (mAP), and F1 score are shown along with misclassification frequency, the fraction of labeled instances of the distractor class, *N. bijuga*, that were erroneously classified as the target class, *C. calyx*.
Detection of an object class of interest – in this case, juveniles of the swordtail squid, *C. calyx* – in imagery containing the class of interest and a distractor class, *N. bijuga*, that closely resembles the target class (see Figure 1F).

opposed to images with reduced resolution) had the smallest and most variable effect on performance (e.g., change in mAP of -0.04-0.05), but this should be taken with the caveat that our image set consisted of a mix of high- and low-resolution imagery, and that resolution mismatches between training and testing data can sometimes result in poor performance (Hendrycks and Dietterich, 2019; Recht et al., 2019).

7 Recommendations and conclusions

Image-based machine learning methods hold tremendous promise for marine science, and for the study of natural systems more generally. These methods can vastly accelerate image processing, while also greatly lowering its costs (Gaston & O'Neill, 2004; MacLeod et al., 2010; Norouzzadeh et al., 2018; Katija et al., 2022). In doing so, they could fundamentally change the spatial coverage and frequency of sampling achieved by field research and monitoring efforts. Our objective in this work has been to provide a guide for researchers who may be new to these methods, but wish to apply them to their own data. If image-based machine learning methods are to be more widely adopted and fully exploited, the current high barrier to entry associated with these methods must be lowered (Crosby et al., 2023). We therefore conclude with four suggestions for the research community that we believe could help expand the use of, and access to image-based machine learning tools across marine science.

7.1 Open sharing of labeled image datasets from the field

At present, the ability of researchers to test and engineer ML methods relevant to the tasks marine scientists want to perform on imagery is constrained by the limited publicly available data for training and testing these methods. Thus, among the most important steps that can be taken to improve ML models for use in the marine domain, is to increase the availability, coverage, quality, and size of domain-relevant labeled image datasets, as well as the standardization of label formats and class naming conventions across those datasets. As Table 2 shows, available datasets focus rather heavily on tropical fishes, benthic habitats, coral, and marine phytoplankton, whereas imagery of other kinds of objects of interest and imagery from other habitats is not as well represented. Researchers who generate manually labeled image datasets in the course of their work would contribute much to the community by making those datasets available in a form that is easily readable by ML pipelines. The issue of readability extends beyond using standard file formats and labeling methods, it also means using class naming conventions that are interpretable and useable by other researchers in the future (Schoening et al., 2022). Idiosyncratic class definitions – for example the use of project- or institution-specific operational taxonomic units – are one major factor that limits the utility of many existing image datasets (Howell et al., 2019). The more standardized and interoperable such datasets become, the more tractable it will be to fully exploit the tremendous volume of ocean imagery currently being collected (Schoening et al., 2022).

Good methods for releasing and publicizing datasets include stand-alone publications (e.g., Saleh et al., 2020; Ditria et al., 2021), publication of datasets as part of standard research publications (e.g., Sosik & Olson, 2007), or contributing datasets to existing open image repositories such as *FathomNet* (Katija et al., 2022) and *CoralNet* (Williams et al., 2019). Of course, constructing labeled image datasets requires funding, domain expertise, and a significant commitment of personnel time. It is therefore crucial that researchers who generate such datasets and the funding sources that support them receive credit. This will involve a shift in perspective from viewing labeled imagery as simply a means to an end, to viewing these kinds of datasets as valid research products in their own right (Qin et al., 2016; Ditria et al., 2021; Koh et al., 2021). Fortunately, this shift in perspective is already beginning to occur, and we expect funding agencies, tenure and promotion committees, and the broader research community will continue to move in the direction of recognizing the value of producing and sharing high-quality labeled image datasets.

7.2 Sharing of open source code for repeating analyses

A second recommendation is aimed at researchers who are developing and testing ML methods for analyzing imagery from the field. It is now commonplace among the larger computer vision community for preprints, conference publications, and journal publications to include links to code repositories that contain the code necessary to repeat analyses. We encourage researchers who are developing ML methods to solve problems

in marine science to follow this same practice. Providing the code that accompanies work described in publications can accelerate research. While newer studies are beginning to follow this practice, it is still not as widespread among researchers working in marine science as it is in the broader machine learning community (Table 3). Code can be efficiently shared, for example, through GitHub repositories or through "model zoo" features of existing image repositories (e.g., https://github.com/fathomnet/models). The machine learning community is adopting standards to further enable model sharing *via* model and dataset cards, resources that allow users to understand at a glance what they are downloading (Mitchell et al., 2019). Applying similar standards in the marine science community would help ensure that code and accompanying data is structured and benchmarked consistently across studies.

7.3 Develop and adopt standards for model evaluation that accurately capture performance in common use-cases

At present, there has been little standardization of model performance metrics reported in papers that apply image-based machine learning to problems in marine science. Different papers report different metrics that often include just one or a few of the performance measures described in "*Evaluating model performance*" above. The most commonly reported metric across studies is classification accuracy (Table 3), but, as noted above, this metric is subject to biases that inherently make comparisons across studies problematic (Tharwat, 2020). Another less obvious issue is that different studies compute performance metrics from test data sets that are built in very different ways. For example, some studies compute performance from a single random partition of the data into training, validation, and test sets. Others perform several random partitions of the overall dataset using a k-fold cross-validation procedure. Others still report true out-of-domain statistics computed on test data from specific locations or time periods that were held out during training (see Table 3). The manner in which test imagery is selected (e.g., at random from in-domain data vs. from out-of-domain data) can have a major impact on performance measures, and any fair comparison between methods clearly requires that performance statistics of competing methods be computed in the same way.

In the end, the most appropriate performance measures will be the ones that best reflect how a model will perform at the task for which it is ultimately intended to be used (González et al., 2017). At the same time, adopting a standard will likely be necessary if performance is to be compared among studies. To achieve this compromise, it will be productive for the community of developers and users of image-based machine learning methods to begin a conversation about the most appropriate standards for evaluating models and comparing model performance among studies, with the goal of identifying metrics that meet the needs of researchers. One potentially fruitful question that could guide this conversation is how the standard performance measures used to evaluate machine learning models (e.g., precision, recall, F1) relate to widely-used statistics scientists often want to compute using image data (e.g., abundance, species richness, measures of ecological community composition, Durden et al., 2021).

7.4 Develop open source, GUI-based applications that implement full image analysis pipelines

A full pipeline for applying image-based ML models in a versatile way requires software to carry out tasks ranging from image labeling and curation to visualizing results of ML model predictions. The transition from largely manual analysis of imagery to ML-based automated analysis is already taking place in other fields, and the availability of free, GUI-based, and actively maintained software packages that integrate all of these tasks has helped facilitate this transition. We point to the *DeepLabCut* package (Mathis and Mathis, 2020; Mathis et al., 2020, https://github.com/DeepLabCut/DeepLabCut) developed for the study of neuroscience and quantitative behavior from laboratory videos as a potent example of how easy-to-use software can rapidly increase use of ML methods within a field. Although some efforts are underway to produce similar "all-in-one" packages for analyzing imagery from marine environments (e.g., the VIAME project; Richards et al., 2019), and several application-specific packages are already in use (e.g. CoralNet, Lozada-Misa et al., 2017; ReefCloud, ReefCloud, 2021), most research groups that apply image-based ML models to data from the field still use custom software pipelines that often combine many packages and software modules (see references in Table 3). We believe that creating software architectures that allow users to easily build their own annotated image libraries and to quickly test and evaluate performance of a suite of widely used ML methods may be the single biggest step that can be taken to encourage broader adoption of these methods in marine science.

8 Conclusions

The evolving research needs of the marine science community will undoubtedly lead to new priorities, and we do not intend these suggestions to be exhaustive. Yet, we believe these steps would go a long way toward making image-based machine learning easier to use, more reliable, and more accessible. As we move toward these goals, it will be crucial to create an open dialogue between researchers who are developing and testing image-based ML methods and researchers who are collecting, labeling, and analyzing imagery from the field. Such a dialogue will help fuel the development of novel methods that empower marine scientists to use machine learning to study the ocean in ways that were never before possible.

Author contributions

All authors designed research. All authors performed research. All authors wrote manuscript. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by National Science Foundation grant IOS-1855956 and REU supplement to AMH, and National Science Foundation grant EF-2222478 to AKF and AMH. This work was produced as a result of the AI for the Ocean undergraduate interdisciplinary training program in science and technology.

Acknowledgments

We thank B. Schlining, B. Martin, and M. Gil for useful discussions, advice, and technical support that improved this manuscript.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "{TensorFlow}: a system for {Large-scale} machine learning," in *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (Savannah, GA, USA: USENIX Association). 265–283.

Allken, V., Handegard, N. O., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. (2019). Fish species identification using a convolutional neural network trained on synthetic data. *ICES J. Mar. Sci.* 76, 342–349. doi: 10.1093/icesjms/fsy147

Bamford, C. C. G., Kelly, N., Dalla Rosa, L., Cade, D. E., Fretwell, P. T., Trathan, P. N., et al. (2020). A comparison of baleen whale density estimates derived from overlapping satellite imagery and a shipborne survey. *Sci. Rep.* 10 (1), 1–12. doi: 10.1038/s41598-020-69887-y

Beery, S., Agarwal, A., Cole, E., and Birodkar, V. (2021). The iwildcam 2021 competition dataset. *arXiv preprint arXiv* 2105.03494. doi: 10.48550/arXiv.2105.03494

Beery, S., van Horn, G., and Perona, P. (2018). "Recognition in terra incognita," in *European Conference on Computer Vision (ECCV)*. Lecture Notes in Computer Science (Cham: Springer) 456–473. doi: 10.48550/arXiv.1807.04975

Beijbom, O., Edmunds, P. J., Roelfsema, C., Smith, J., Kline, D. I., Neal, B. P., et al. (2015). Towards automated annotation of benthic survey images: variability of human experts and operational modes of automation. *PloS One* 10 (7), e0130312. doi: 10.1371/journal.pone.0130312

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (8), 1798–1828. doi: 10.1109/TPAMI.2013.50

Bichot, N. P., and Schall, J. D. (1999). Saccade target selection in macaque during feature and conjunction visual search. *Visual Neurosci.* 16 (1), 81–89. doi: 10.1017/S0952523899161042

Bloice, M. D., Roth, P. M., and Holzinger, A. (2019). Biomedical image augmentation using augmentor. *Bioinformatics* 35 (21), 4522–4524. doi: 10.1093/bioinformatics/btz259

Buber, E., and Diri, B. (2018). "Performance analysis and CPU vs GPU comparison for deep learning," in *6th International Conference on Control Engineering & Information Technology (CEIT)*, Istanbul, Turkey. 1–6. doi: 10.1109/CEIT.2018.8751930

Burford, B. P., Robison, B. H., and Sherlock, R. E. (2015). Behaviour and mimicry in the juvenile and subadult life stages of the mesopelagic squid chiroteuthis calyx. *J. Mar. Biol. Assoc. United Kingdom* 95 (6), 1221–1235. doi: 10.1017/S0025315414001763

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., and Kalinin, A. A. (2020). Albumentations: fast and flexible image augmentations. *Information* 11 (2), 125. doi: 10.3390/info11020125

Chapelle, O., Haffner, P., and Vapnik, V. N. (1999). Support vector machines for histogram-based image classification. *IEEE Trans. Neural Networks* 10 (5), 1055–1064. doi: 10.1109/72.788646

Chegini, H., Beltran, F., and Mahanti, A. (2022). Designing and developing a weed detection model for California thistle(TOIT). *ACM Trans. Internet Technol.*

Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*. Lecture Notes in Computer Science (Cham: Springer) 801–818.

Crosby, A., Orenstein, E. C., Poulton, S. E., Bell, K. L., Woodward, B., Ruhl, H., et al. (2023). "Designing ocean vision AI: an investigation of community needs for imaging-based ocean conservation," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. (New York, NY, USA: Association for Computing Machinery) 1–16.

Cunningham, P., Cord, M., and Delany, S. J. (2008). "Supervised learning," in *Machine learning techniques for multimedia. cognitive technologies*. Eds. M. Cord and P. Cunningham (Berlin, Heidelberg: Springer). doi: 10.1007/978-3-540-75171-7_2

Das, R., Wang, Y. X., and Moura, J. M. (2021). "On the importance of distractors for few-shot classification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Montreal, QC, Canada. 9030–9040. doi: 10.1109/ICCV48922.2021.00890

Ditria, E. M., Connolly, R. M., Jinks, E. L., and Lopez-Marcano, S. (2021). Annotated video footage for automated identification and counting of fish in unconstrained seagrass habitats. *Front. Mar. Sci.* 8, 629485. doi: 10.3389/fmars.2021.629485

Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., and Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: optimizing animal ecology with deep learning. *Front. Mar. Sci.* 429. doi: 10.3389/fmars.2020.00429

Drew, E. A. (1977). A photographic survey down the seaward reef-front of aldabra atoll. *Atoll Res. Bull.* 193, 1–7. doi: 10.5479/si.00775630.193.1

Durden, J. M., Hosking, B., Bett, B. J., Cline, D., and Ruhl, H. A. (2021). Automated classification of fauna in seabed photographs: the impact of training and validation dataset size, with considerations for the class imbalance. *Prog. Oceanography* 196, 102612. doi: 10.1016/j.pocean.2021.102612

Durden, J. M., Schoening, T., Althaus, F., Friedman, A., Garcia, R., Glover, A. G., et al. (2016). "Perspectives in visual imaging for marine biology and ecology: from acquisition to understanding," in *Oceanography and Marine Biology*. 9–80 (CRC Press).

Ellen, J. S., Graff, C. A., and Ohman, M. D. (2019). Improving plankton image classification using context metadata. *Limnology Oceanography: Methods* 17 (8), 439–461. doi: 10.1002/lom3.10324

Everingham, M., Van Gool, L., Williams, C.K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comp. Vision* 88, 303–338.

Fahimipour, A. K., Gil, M. A., Celis, M. R., Hein, G. F., Martin, B. T., and Hein, A. M. (2023). Wild animals suppress the spread of socially transmitted misinformation. *Proc. Natl. Acad. Sci.* 120 (14), e2215428120. doi: 10.1073/pnas.2215428120

Fei-Fei, R., Fergus, L., and Perona, P. (2004). "Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories," *Conference on Computer Vision and Pattern Recognition Workshop*, Washington, DC, USA in CVPR workshop on generative-model based vision. pp. 178–178. doi: 10.1109/CVPR.2004.383

Fernandes, A. F. A., Dórea, J. R. R., and Rosa, G. J. D. M. (2020). Image analysis and computer vision applications in animal sciences: an overview. *Front. Veterinary Sci.* 7, 551269. doi: 10.3389/fvets.2020.551269

Francisco, F. A., Nührenberg, P., and Jordan, A. (2020). High-resolution, non-invasive animal tracking and reconstruction of local environment in aquatic ecosystems. *Movement Ecol.* 8 (1), 1–12. doi: 10.1186/s40462-020-00214-w

Gaston, K. J., and O'Neill, M. A. (2004) Automated species identification: why not? philosophical transactions of the royal society of London *Ser. B: Biol. Sci.* 359(1444) 655–667 doi: 10.1098/rstb.2003.1442

Gomes-Pereira, J. N., Auger, V., Beisiegel, K., Benjamin, R., Bergmann, M., Bowden, D., et al. (2016). Current and future trends in marine image annotation software. *Prog. Oceanography* 149, 106–120. doi: 10.1016/j.pocean.2016.07.005

González, P., Álvarez, E., Díez, J., López-Urrutia, Á., and del Coz, J. J. (2017). Validation methods for plankton image classification systems. *Limnology Oceanography: Methods* 15 (3), 221–237. doi: 10.1002/lom3.10151

González, P., Castano, A., Peacock, E. E., Díez, J., Del Coz, J. J., and Sosik, H. M. (2019). Automatic plankton quantification using deep features. *J. Plankton Res.* 41 (4), 449–463. doi: 10.1093/plankt/fbz023

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning* (Cambridge, MA USA: MIT press).

Goodwin, M., Halvorsen, K., Jiao, L., Knausgård, K., Martin, A., Moyano, M., et al. (2021). Unlocking the potential of deep learning for marine ecology: overview, applications, and outlook. *ICES J. of Mar. Sci.* 79 (2), 319–336. doi: 10.48550/arXiv.2109.14737

Graving, J. M., Chae, D., Naik, H., Li, L., Koger, B., Costelloe, B. R., et al. (2019). DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning. *eLife* 8, e47994. doi: 10.7554/eLife.47994.sa2

Hein, A. M., Gil, M. A., Twomey, C. R., Couzin, I. D., and Levin, S. A. (2018). Conserved behavioral circuits govern high-speed decision-making in wild fish shoals. *Proc. Natl. Acad. Sci.* 115 (48), 12224–12228. doi: 10.1073/pnas.1809140115

Hendrycks, D., and Dietterich, T. G. (2019). Benchmarking neural network robustness to common corruptions and surface variations. *arXiv preprint arXiv* 1807.01697. doi: 10.48550/arXiv.1807.01697

Henrichs, D. W., Anglès, S., Gaonkar, C. C., and Campbell, L. (2021). Application of a convolutional neural network to improve automated early warning of harmful algal blooms. *Environ. Sci. pollut. Res.* 28 (22), 28544–28555. doi: 10.1007/s11356-021-12471-2

Howell, K. L., Davies, J. S., Allcock, A. L., Braga-Henriques, A., Buhl-Mortensen, P., Carreiro-Silva, M., et al. (2019). A framework for the development of a global standardised marine taxon reference image database (SMarTaR-ID) to support image-based analyses. *PloS One* 14 (12), e0218904. doi: 10.1371/journal.pone.0218904

Irisson, J. O., Ayata, S. D., Lindsay, D. J., Karp-Boss, L., and Stemmann, L. (2022). Machine learning for the study of plankton and marine snow from images. *Ann. Rev. Mar. Sci.* 14, 277–301. doi: 10.1146/annurev-marine-041921-013023

Jaüger, J., Simon, M., Denzler, J., and Wolff, V. (2015). "Croatian Fish dataset: fine-grained classification of fish species in their natural habitat," in *Proceedings of the Machine Vision of Animals and their Behaviour Workshop*. Proceedings of the British Machine Vision Conference. (Swansea, UK: British Machine Vision Association)

Jackett, C., Althaus, F., Maguire, K., Farazi, M., Scoulding, B., Untiedt, C., et al. (2023). A benthic substrate classification method for seabed images using deep learning: application to management of deep-sea coral reefs. *J. Appl. Ecol.* 00, 1–20 doi: 10.1111/1365-2664.14408

Jalal, A., Salman, A., Mian, A., Shortis, M., and Shafait, F. (2020). Fish de- tection and species classification in underwater environments using deep learning with temporal information. *Ecol. Inf.* 57, 101088. doi: 10.1016/j.ecoinf.2020.101088

Ji, X., Henriques, J. F., and Vedaldi, A. (2019). "Invariant information clustering for unsupervised image classification and segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, Korea (South). 9865–9874. doi: 10.1109/ICCV.2019.00996

Katija, K., Orenstein, E., Schlining, B., Lundsten, L., Barnard, K., Sainz, G., et al. (2022). FathomNet: a global image database for enabling artificial intelligence in the ocean. *Sci. Rep.* 12, 15914. doi: 10.1038/s41598-022-19939-2

Katija, K., Roberts, P. L., Daniels, J., Lapides, A., Barnard, K., Risi, M., et al. (2021). "Visual tracking of deepwater animals using machine learning-controlled robotic underwater vehicles," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, Waikoloa, HI, USA. 860–869. doi: 10.1109/WACV48630.2021.00090

Knausgård, K. M., Wiklund, A., Sørdalen, T. K., Halvorsen, K. T., Kleiven, A. R., Jiao, L., et al. (2021). Temperate fish de- tection and classification: a deep learning based approach. *Appl. Intell.* 52, 1–14.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., et al. (2021). "WILDS: a benchmark of in-the-Wild distribution shifts," in *Proceedings of the 38th International Conference on Machine Learning, in Proceedings of Machine Learning Research*, Vol. 139. 5637–5664. Available at: https://proceedings.mlr.press/v139/koh21a.html

Kyathanahally, S., Hardeman, T., Reyes, M., Merz, E., Bulas, T., Pomati, F., et al. (2022). Ensembles of vision transformers as a new paradigm for automated classification in ecology. doi: 10.48550/arXiv.2203.01726

Langenkämper, D., Zurowietz, M., Schoening, T., and Nattkemper, T. W. (2017). Biigle 2.0-browsing and annotating large marine image collections. *Front. Mar. Sci.* 4, 83. doi: 10.3389/fmars.2017.00083

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature* 521 (7553), 436–444. doi: 10.1038/nature14539

Li, D., Wang, Q., Li, X., Niu, M., Wang, H., and Liu, C. (2022). Recent advances of machine vision technology in fish classification. *ICES J. Mar. Sci.* 79 (2), 263–284. doi: 10.1093/icesjms/fsab264

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "September. Microsoft coco: common objects in context," in *European conference on computer vision – ECCV 2014*. Lecture Notes in Computer Science (Cham: Springer) 8693, 740–755. doi: 10.1007/978-3-319-10602-1_48

Lombard, F., Boss, E., Waite, A. M., Vogt, M., Uitz, J., Stemmann, L., et al. (2019). Globally consistent quantitative observations of planktonic ecosystems. *Front. Mar. Sci.* 6, 196. doi: 10.3389/fmars.2019.00196

Longley, W. H., and Martin, C. (1927). The first autochromes from the ocean bottom. *Nat. Geog. Mag.* 51, 56–60.

Lozada-Misa, P., Schumacher, B. D., and Vargas-Angel, B. (2017). *Analysis of benthic survey images via coralnet: a summary of standard operating procedures and guidelines. administrative report no. h-17-02* (Honolulu, HI: Joint Institute for Marine and Atmospheric Research University).

Luo, J. Y., Irisson, J. O., Graham, B., Guigand, C., Sarafraz, A., Mader, C., et al. (2018). Automated plankton image analysis using convolutional neural networks. *Limnology Oceanography: Methods* 16 (12), 814–827. doi: 10.1002/lom3.10285

Lv, Y., Zhang, J., Dai, Y., Li, A., Liu, B., Barnes, N., et al. (2021). "Simultaneously localize, segment and rank the camouflaged objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA. 11591–11601. doi: 10.1109/CVPR46437.2021.01142

MacLeod, N., Benfield, M., and Culverhouse, P. (2010). Time to automate identification. *Nature* 467 (7312), 154–155. doi: 10.1038/467154a

Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., et al. (2018). Exploring the limits of weakly supervised pretraining. In: *Proceedings of the European conference on computer vision (ECCV)*. pp. 181–196. doi: 10.1007/978-3-030-01216-8_12

Marochov, M., Stokes, C. R., and Carbonneau, P. E. (2021). Image classification of marine-terminating outlet glaciers in Greenland using deep learning methods. *Cryosphere* 15, 5041–5059. doi: 10.5194/tc-15-5041-2021

Marr, D. (1982). *Vision: a computational approach* (Cambridge, MA USA: MIT Press).

Mathis, M. W., and Mathis, A. (2020). Deep learning tools for the measurement of animal behavior in neuroscience. *Curr. Opin. Neurobiol.* 60, 1–11. doi: 10.1016/j.conb.2019.10.008

Mathis, A., Schneider, S., Lauer, J., and Mathis, M. W. (2020). A primer on motion capture with deep learning: principles, pitfalls, and perspectives. *Neuron* 108 (1), 44–65. doi: 10.1016/j.neuron.2020.09.017

McGill, B. J., Etienne, R. S., Gray, J. S., Alonso, D., Anderson, M. J., Benecha, H. K., et al. (2007). Species abundance distributions: moving beyond single prediction theories to integration within an ecological framework. *Ecol. Lett.* 10 (10), 995–1015. doi: 10.1111/j.1461-0248.2007.01094.x

Michaels, W. L., Handegard, N. O., Malde, K., and Hammersland-White, H. (2019). *Machine learning to improve marine science for the sustainability of living ocean resources: report from the 2019 Norway - U.S. workshop* (NOAA Tech. Memo), 99. Available at: https://spo.nmfs.noaa.gov/tech-memos/.

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., et al. (2019). Model cards for model reporting. in proceedings of the conference on fairness, accountability, and transparency. In *Proceedings of the conference on fairness, accountability, and transparency*. 220–229. doi: 10.1145/3287560.3287596

Moeller, A. K., Lukacs, P. M., and Horne., J. S. (2018). Three novel methods to estimate abundance of unmarked animals using remote cameras. *Ecosphere* 9 (8), e02331. doi: 10.1002/ecs2.2331

Nepovinnykh, E., Eerola, T., and Kalviainen, H. (2020). "Siamese Network based pelage pattern matching for ringed seal re-identification," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision workshops*, Snowmass, CO, USA. 25–34. doi: 10.1109/WACVW50321.2020.9096935

Norouzzadeh, M. S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M.S., Packer, C., et al. (2018). Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proc. Natl. Acad. Sci. U.S.A.* 115, E5716–E5725. doi: 10.1073/pnas.1719367115

Orenstein, E. C., Ayata, S. D., Maps, F., Becker, É.C., Benedetti, F., Biard, T., et al. (2022). Machine learning techniques to characterize functional traits of plankton from image data. *Limnology oceanography* 67 (8), 1647–1669. doi: 10.1002/lno.12101

Orenstein, E. C., and Beijbom, O. (2017). "Transfer learning and deep feature extraction for planktonic image data sets," in *2017 IEEE Winter Conf Appl. Comput. Vision (WACV)*, Santa Rosa, CA, USA. 1082–1088. doi: 10.1109/WACV.2017.125

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* 32, 1–12.

Peña, A., Pérez, N., Benitez, D. S., and Hearn, A. (2021). "Hammerhead shark species monitoring with deep learning," in *Applications of computational intelligence. ColCACI 2020. Communications in Computer and Information Science*, vol. 1346. Eds. A. D. Orjuela-Cañón, J. Lopez, J. D. Arias-Londoño and J. C. Figueroa-García (Cham: Springer. doi: 10.1007/978-3-030-69774-7_4

Picheral, M., Colin, S., and Irisson, J. O. (2017). EcoTaxa, a tool for the taxonomic classification of images. Available at: http://ecotaxa.obs-vlfr.fr. Accessed 05-27-2023

Piechaud, N., Hunt, C., Culverhouse, P. F., Foster, N. L., and Howell, K. L. (2019). Automated identification of benthic epifauna with computer vision. *Mar. Ecol. Prog. Ser.* 615, 15–30. doi: 10.3354/meps12925

Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. (2016). DeepFish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing* 187, 49–58. doi: 10.1016/j.neucom.2015.10.122

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. (2019). "Do imagenet classifiers generalize to imagenet?," in *International conference on machine learning*. in Proceedings of Machine Learning Research 97, 5389–5400. Available at: https://proceedings.mlr.press/v97/recht19a.html

Redmon, J., Divvala, S., and Girshick and Farhadi, R. A. (2016). You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* pp. 779–788.

ReefCloud (2021). Available at: https://reefcloud.ai.

Richards, B. L., Beijbom, O., Campbell, M. D., Clarke, M. E., Cutter, G., Dawkins, M., et al. (2019). Automated analysis of underwater imagery: accomplishments, products, and vision. *NOAA technical memorandum NMFS PIFSC* 83. doi: 10.25923/0cwf-4714

Robison, B. H., Reisenbichler, K. R., and Sherlock, R.E. (2017). The coevolution of midwater research and ROV technology at MBARI. *Oceanography* 30 (4), 26–37. doi: 10.5670/oceanog.2017.421

Rodriguez-Ramirez, A., González-Rivero, M., Beijbom, O., Bailhache, C., Bongaerts, P., Brown, K. T., et al. (2020). A contemporary baseline record of the world's coral reefs. *Sci. Data* 7 (1), 1–15. doi: 10.1038/s41597-020-00698-6

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 211–252. doi: 10.1007/s11263-015-0816-y

Saleh, A., Laradji, I. H., Konovalov, D. A., Bradley, M., Vazquez, D., and Sheaves, M. (2020). A realistic fish-habitat dataset to evaluate algorithms for underwater visual analysis. *Sci. Rep.* 10 (1), 1–10. doi: 10.1038/s41598-020-71639-x

Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., et al. (2016). Fish species classification in unconstrained underwater environments based on deep learning. *Limnology Oceanography Methods* 14, 570–585. doi: 10.1002/lom3.10113

Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M. R., Khurshid, K., et al. (2020). Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES J. Mar. Sci.* 77, 1295–1307. doi: 10.1093/icesjms/fsz025

Schneider, S., Greenberg, S., Taylor, G. W., and Kremer, S. C. (2020). Three critical factors affecting automated image species recognition performance for camera traps. *Ecol. Evol.* 10 (7), 3503–3517. doi: 10.1002/ece3.6147

Schoening, T., Durden, J. M., Faber, C., Felden, J., Heger, K., Hoving, H. J. T., et al. (2022). Making marine image data FAIR. *Sci. Data* 9 (1), 414. doi: 10.1038/s41597-022-01491-3

Schoening, T., Köser, K., and Greinert, J. (2018). An acquisition, curation and management workflow for sustainable, terabyte-scale marine image analysis. *Sci. Data* 5 (1), 1–12. doi: 10.1038/sdata.2018.181

Scoulding, B., Maguire, K., and Orenstein, E. C. (2022). Evaluating automated benthic fish detection under variable conditions. *ICES J. Mar. Sci.* 79 (8), 2204–2216. doi: 10.1093/icesjms/fsac166

Shankar, V., Roelofs, R., Mania, H., Fang, A., Recht, B., and Schmidt, L. (2020). "Evaluating machine accuracy on imagenet," in *International Conference on Machine Learning*. in Proceedings of Machine Learning Research. 119, 8634–8644. Available at: https://proceedings.mlr.press/v119/shankar20c.html.

Sharma, N., Scully-Power, P., and Blumenstein, M. (2018). Shark detection from aerial imagery using region-based CNN, a study. *AI 2018: Adv. Artifical Intell.* (Cham: Springer) 11320, 224–236. doi: 10.1007/978-3-030-03991-2_23

Sosik, H. M., and Olson, R. J. (2007). Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. *Limnology Oceanography: Methods* 5 (6), 204–216. doi: 10.4319/lom.2007.5.204

Tan, M., Langenkämper, D., and Nattkemper, T. W. (2022). The impact of data augmentations on deep learning-based marine object classification in benthic image transects. *Sensors* 22 (14), 5383. doi: 10.3390/s22145383

Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., and Schmidt, L. (2020). Measuring robustness to natural distribution shifts in image classification. *Adv. Neural Inf. Process. Syst.* 33, 18583–18599.

Tharwat, A. (2020). Classification assessment methods. *Appl. Computing Inf.* 17 (1), pp. 168–192. doi: 10.1016/j.aci.2018.08.003

Uijlings, J. R., Andriluka, M., and Ferrari, V. (2020). "Panoptic image annotation with a collaborative assistant," in *Proceedings of the 28th ACM International Conference on Multimedia*. (New York, NY, USA: Association for Computing Machinery) 3302–3310. 10.1145/3394171.3413812

Villon, S., Iovan, C., Mangeas, M., Claverie, T., Mouillot, D., Villéger, S., et al. (2021). Automatic underwater fish species classification with limited data using few-shot learning. *Ecol. Inf.* 63, 1–6. doi: 10.1016/j.ecoinf.2021.101320

Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., et al. (2018). A deep learning method for accu- rate and fast identification of coral reef fishes in underwater images. *Ecol. Inf.* 48, 238–244. doi: 10.1016/j.ecoinf.2018.09.007

Walker, J. L., and Orenstein, E. C. (2021). "Improving rare-class recognition of marine plankton with hard negative mining," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Montreal, BC, Canada. 3672–3682. doi: 10.1109/ICCVW54120.2021.00410

Williams, I. D., Couch, C. S., Beijbom, O., Oliver, T. A., Vargas-Angel, B., Schumacher, B. D., et al. (2019). Leveraging automated image analysis tools to transform our capacity to assess status and trends of coral reefs. *Front. Mar. Sci.* 6. doi: 10.3389/fmars.2019.00222

Wu, Y., Kirillov, A., Massa, F., Lo, W. Y., and Girshick, R. (2019). Detectron2.

Wyatt, M., Radford, B., Callow, N., Bennamoun, M., and Hickey, S. (2022). Using ensemble methods to improve the robustness of deep learning for image classification in marine environments. *Methods Ecol. Evol.* 13 (6), 1317–1328. doi: 10.1111/2041-210X.13841

Yusup, I. M., Iqbal, M., and Jaya, I. (2020). "Real-time reef fishes identifi- cation using deep learning," in *IOP Conference Series Earth and Envi- ronmental Science*, (Bristol, UK: IOP Publishing) Vol. 429. 012046.

Zhao, J., Li, Y., Zhang, F., Zhu, S., Liu, Y., Lu, H., et al. (2018). Semi- supervised learning-based live fish identification in aquaculture us- ing modified deep convolutional generative adversarial networks. *Trans. ASABE* 61, 699–710. doi: 10.13031/trans.12684

Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., and Hu, W. (2018). "Distractor-aware siamese networks for visual object tracking," in *Proceedings of the European conference on computer vision (ECCV) 2018: 15th European Conference*, Munich, Germany, September 8–14, 2018. (Berlin, Heidelberg: Springer-Verlag) 101–117. doi: 10.1007/978-3-030-01240-3_7

Zoph, B., Cubuk, E. D., Ghiasi, G., Lin, T. Y., Shlens, J., and Le, Q. V. (2020). "Learning data augmentation strategies for object detection," in *European conference on computer vision*. Lecture Notes in Computer Science (Cham: Springer) 566–583.

frontiers | Frontiers in Marine Science

TYPE Methods
PUBLISHED 08 June 2023
DOI 10.3389/fmars.2023.1187771

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Rubens Lopes,
University of São Paulo, Brazil
Chunsheng Wang,
Ministry of Natural Resources, China

*CORRESPONDENCE
Moritz S. Schmid
schmidm@oregonstate.edu

RECEIVED 16 March 2023
ACCEPTED 15 May 2023
PUBLISHED 08 June 2023

CITATION
Schmid MS, Daprano D, Damle MM, Sullivan CM, Sponaugle S, Cousin C, Guigand C and Cowen RK (2023) Edge computing at sea: high-throughput classification of *in-situ* plankton imagery for adaptive sampling.
Front. Mar. Sci. 10:1187771.
doi: 10.3389/fmars.2023.1187771

COPYRIGHT
© 2023 Schmid, Daprano, Damle, Sullivan, Sponaugle, Cousin, Guigand and Cowen. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Edge computing at sea: high-throughput classification of *in-situ* plankton imagery for adaptive sampling

Moritz S. Schmid[1*], Dominic Daprano[2], Malhar M. Damle[2], Christopher M. Sullivan[2,3], Su Sponaugle[1,4], Charles Cousin[5], Cedric Guigand[5] and Robert K. Cowen[1]

[1]Hatfield Marine Science Center, Oregon State University, Newport, OR, United States, [2]Center for Quantitative and Life Sciences, Oregon State University, Corvallis, OR, United States, [3]College of Earth, Ocean, and Atmospheric Sciences, Oregon State University, Corvallis, OR, United States, [4]Department of Integrative Biology, Oregon State University, Corvallis, OR, United States, [5]Bellamare LLC, San Diego, CA, United States

The small sizes of most marine plankton necessitate that plankton sampling occur on fine spatial scales, yet our questions often span large spatial areas. Underwater imaging can provide a solution to this sampling conundrum but collects large quantities of data that require an automated approach to image analysis. Machine learning for plankton classification, and high-performance computing (HPC) infrastructure, are critical to rapid image processing; however, these assets, especially HPC infrastructure, are only available post-cruise leading to an 'after-the-fact' view of plankton community structure. To be responsive to the often-ephemeral nature of oceanographic features and species assemblages in highly dynamic current systems, real-time data are key for adaptive oceanographic sampling. Here we used the new *In-situ* Ichthyoplankton Imaging System-3 (ISIIS-3) in the Northern California Current (NCC) in conjunction with an edge server to classify imaged plankton in real-time into 170 classes. This capability together with data visualization in a heavy.ai dashboard makes adaptive real-time decision-making and sampling at sea possible. Dual ISIIS-Deep-focus Particle Imager (DPI) cameras sample 180 L s^{-1}, leading to >10 GB of video per min. Imaged organisms are in the size range of 250 µm to 15 cm and include abundant crustaceans, fragile taxa (e.g., hydromedusae, salps), faster swimmers (e.g., krill), and rarer taxa (e.g., larval fishes). A deep learning pipeline deployed on the edge server used multithreaded CPU-based segmentation and GPU-based classification to process the imagery. AVI videos contain 50 sec of data and can contain between 23,000 - 225,000 particle and plankton segments. Processing one AVI through segmentation and classification takes on average 3.75 mins, depending on biological productivity. A heavyDB database monitors for newly processed data and is linked to a heavy.ai dashboard for interactive data visualization. We describe several examples where imaging, AI, and data visualization enable adaptive sampling that can have a transformative effect on oceanography. We envision AI-enabled adaptive sampling to have a high impact on our ability to resolve biological responses to important oceanographic features in the NCC, such as oxygen minimum

10.3389/fmars.2023.1187771

zones, or harmful algal bloom thin layers, which affect the health of the ecosystem, fisheries, and local communities.

KEYWORDS

adaptive sampling, edge computing, ocean technology, underwater imaging, plankton ecology, machine learning, data visualization, California Current

1 Introduction

Marine plankton form the base of most ocean food webs. Understanding how these communities are likely to change in the future in response to climate change is a critical knowledge need (Ratnarajah et al., 2023). Yet how specific environmental drivers impact different levels of the food web, and how this might transfer up and down different food webs remains poorly understood. Plankton communities in most oceans are diverse and complex. They range over many orders of magnitude in size, thus simultaneous sampling of many taxa can be challenging (Lombard et al., 2019). This issue is exacerbated by plankton net systems that destroy fragile organisms such as jellies and other gelatinous animals (e.g., appendicularians and salps; Wiebe and Benfield, 2003) known to be important to the oceanic carbon cycle (Hopcroft et al., 1998; Luo et al., 2022). Plankton *in-situ* imaging enables the sampling of plankton across a wide range in size, from a few hundred microns to > 10 cm, while keeping fragile organisms intact since no net, and thereby no physical contact, are involved. This can be achieved by a multitude of systems that have different purposes (e.g., O-Cam, Briseño-Avena et al., 2020a; Scripps Plankton Camera system, Orenstein et al., 2020; and PlanktonScope, Song et al., 2020).

The northern California Current (NCC) off the coast of California, Oregon, and Washington, is a dynamic, highly productive eastern boundary current that is of high importance to national fisheries and food security (Reese and Brodeur, 2006; Hickey and Banas, 2008). As part of a study of the planktonic food web dynamics of this system, we used the high resolution *In Situ* Ichthyoplankton Imaging System-3 (ISIIS-3; Figure 1) to image plankton ranging from 250 µm to 15 cm, in their *in-situ* (i.e., natural) environment (Cowen and Guigand, 2008). While ISIIS was developed initially to enhance research of ichthyoplankton (i.e., larval fishes), it obtains images of plankters ranging from diatoms and protists to copepods, jellies, and larval fishes, and has been successfully deployed in a multitude of systems (e.g., the NCC, Swieca et al., 2020; the Straits of Florida, Robinson et al., 2021; and in the Gulf of Mexico and the Mediterranean, Greer et al., 2023).

Use of ISIIS and now ISIIS-3 creates a big data challenge. The combination of high-resolution imagery and the need to image a large volume of water results in extremely high numbers of imaged plankton individuals (0.1 to > 1 billion per study; Schmid et al., 2020; Robinson et al., 2021; Schmid et al., 2021; Schmid et al., 2023b). The two line scan cameras of the ISIIS-3 gather 10 GB of data per min, and >35 TB for a typical two-week research cruise (160 h of imagery).

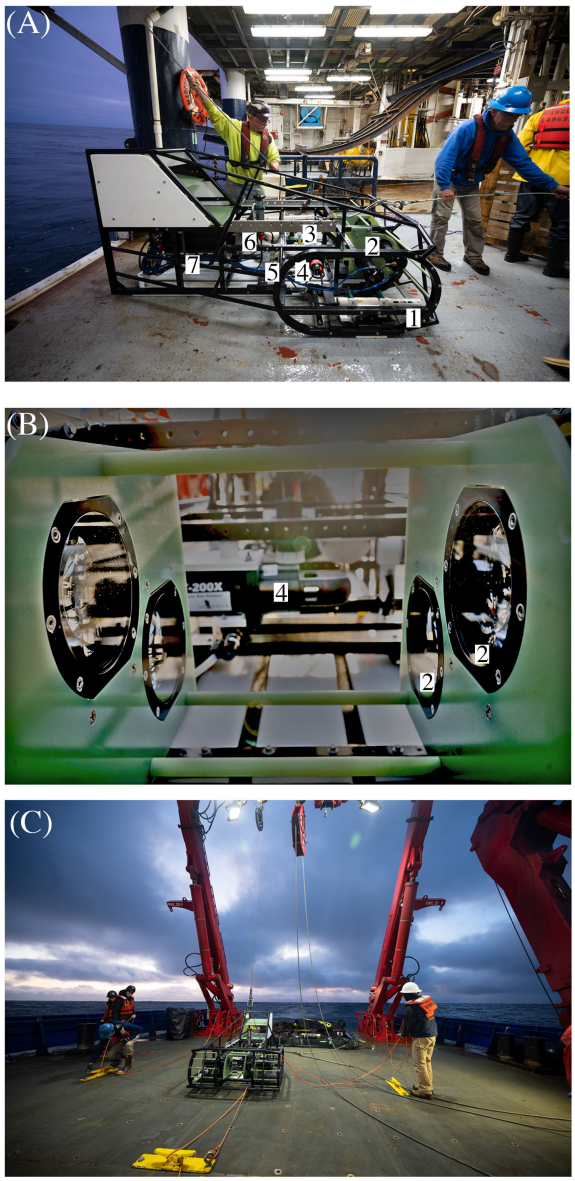


FIGURE 1
ISIIS-3 and its components **(A)** Lateral view; 1 = CTD; 2 = two shadowgraph line-scan cameras; 3 = fluorescence and pH sensors as well as altimeter; 4 = LISST-200X particle imager; 5 = pump and dissolved oxygen probe; 6 = flowmeter; 7 = main computer housing. **(B)** Close-up of the two stacked Bellamare ISIIS-DPI-125 camera units. ISIIS-3 can be deployed through a narrow gate and boom (e.g., on R/V *Langseth*, A) or via the A-frame (e.g., on R/V *Sikuliaq*, **(C)**, while side deployments using a crane are also possible and were carried out in the past (e.g., on R/V *Atlantis*). Photos credit: Ellie Lafferty.

Simultaneous with the development of the ISIIS technology over the last 10 yr., data processing and machine learning pipelines for plankton imagery have also undergone much development (Irisson et al., 2021). Initially, plankton underwater imagery was hand-sorted, but as hard- and software became increasingly available, plankton sorting was automated on desktops with dedicated graphics cards. More recently, university and national supercomputing center machines with enterprise-level graphics cards for machine learning (e.g., NVIDIA A100/V100/P100; Schmid et al., 2021) have become widely available. However, computing time on high-end machines with powerful graphics cards must often be shared with other labs. One solution to this limitation is to tap into nationally funded supercomputing centers, for instance through NSF's XSEDE infrastructure (now ACCESS; Schmid et al., 2021). XSEDE and ACCESS themselves allocate resources on major national supercomputing centers such as the San Diego Supercomputing Center, or the Pittsburgh Supercomputing Center. While such computing power is critical for analyzing large datasets, they are by necessity 'post-cruise' analysis tools, as large node clusters are not portable.

The fact that plankton imagery is usually analyzed after the cruise due to the large quantity of data, precludes it from being used for adaptive sampling, which by definition needs near-immediate data availability. With advancements in ocean technology, thanks to the increased affordability and availability of advanced hard-, and software, the number of studies working on real-time identification and adaptive sampling based on different underwater vehicles has increased though in recent years (Fossum et al., 2019; Ohman et al., 2019; Stankiewicz et al., 2021; Bi et al., 2022). However, having the necessary computing power at sea to classify large quantities of videography remains a bottleneck.

Recent increased availability of edge servers in the civilian sector may resolve this bottleneck, enabling oceanographers to take significant computing power to sea with the potential to acquire *and* analyze extensive data sets while at sea and even during active deployments. In the case of plankton imaging, edge servers coupled with deep-learning pipelines, enable researchers to not only store and back-up the data on redundant drives, but to process the incoming videography (i.e., segmentation and classification), and analyze the data for distributional patterns, all while the instrument is being towed behind the ship. These combined technologies enable the scientific sampling plan to change based on real-time information gathered at-sea. This approach has major consequences for the way oceanographic research can be conducted as it makes adaptive sampling possible - meaning that oceanographic features of interest, e.g., accumulations of particular taxa in low or even hypoxic oxygen waters on the NCC shelf (Chan et al., 2008; Chan et al., 2019), can be targeted for resampling immediately after their detection. A separate benefit of processing data at sea is the ability to reduce (or completely remove) the lag between scientific research cruise completion and being able to work with data for ecological analyses. Here we describe a deep learning pipeline for plankton classification at sea, including databasing and visualization for adaptive sampling. We describe the necessary hardware setup for such an adaptive sampling processing pipeline and how it could be adapted for other imaging systems. The major deliverable is the open-sourced code for the pipeline including classification as well as automation scripts for databasing and visualization. At-sea processing of complex data has the potential to transform oceanographic science.

2 Materials and equipment

2.1 *In-situ* ichthyoplankton imaging system-3

The *In-situ* Ichthyoplankton Imaging System (ISIIS) vehicle has undergone several design modifications since its early inception (Cowen and Guigand, 2008). Here we report on the third vehicle iteration or model - the ISIIS-3. ISIIS-3 (Figure 1) was developed based on several lessons learned from the original design, including a robust open-frame sled design and dual tow point bridle that promotes the shedding of buoyed markers of active fishing gear (e.g., crab pots). The system includes a dual camera setup (55 µm pixel resolution) instead of a single camera to enable a narrower sled design, but without compromising the total sampling volume of 180 L s^{-1}. The system is also more modular than the ISIIS-1 and ISIIS-2 towed vehicles, enabling easier integration of new electronic components. For instance, ISIIS-3 is fitted with a Sequoia Scientific LISST-200X particle imager covering the 1 µm - 500 µm size range, a CTD (Sea-Bird SBE 49 FastCAT), dissolved oxygen probe (Sea-Bird 43), fluorescence sensor (Wet Labs FLRT), photosynthetically active radiation sensor (PAR; Biospherical QCP-2300), and a pH sensor (Seabird SBE 18). ISIIS-3 is towed behind the ship at 2.5 m s^{-1} where it undulates typically between 1 m and 100 m depth or as close as 2 m above the seafloor in shallower waters on the shelf. Data are continuously multiplexed in the ISIIS-3 vehicle, and then sent to the ISIIS-3 control computer on the ship through the glass-fiber of the oceanographic wire, where data are then de-multiplexed and time-stamped.

2.2 Edge server configuration at sea

The edge server used here was a Western Digital (WD) Ultrastar-Edge MR with two Intel Xeon Gold 6230T 2.1 GHz CPUs, each with 20 cores (40 cores total), a NVIDIA Tesla T4 GPU, 512 GiB DDR4 memory, >60 TB of NVMe flash storage, as well as 100 GbE and 10 GbE networking (Figure 2). The edge server ran with Ubuntu 20.04 and DNS, DHCP, TFTP, and HTTP services, enabling the setup of an intranet around the edge server. The NVMe file space of the edge server was configured into a RAID to allow for limited redundancy; specifically, we use ZFS cut with RAIDZ2 with no spares. This provided around 40 TB of usable space and allowed failure of a drive without having to rebuild the drive during data collection. Rebuilding a drive during live data collection would slow down write speed substantially and potentially lead to a loss of image frames.

The DHCP on the edge server enabled other machines on the network (switch and VLAN) to be serviced by the edge server

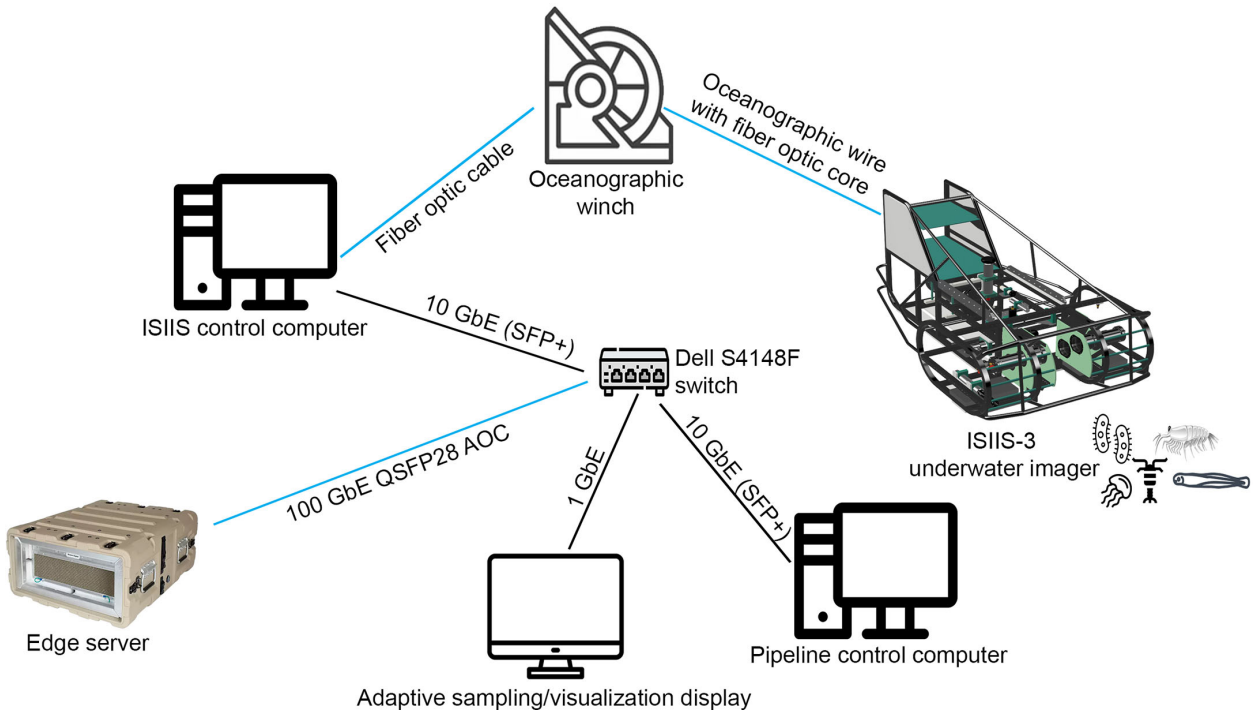


FIGURE 2
The hardware setup associated with the ISIIS-3 control computer and edge server. ISIIS-3 is connected to the ISIIS-3 control computer via fiber (all optic connections in blue). Incoming data are used for flying the sled (e.g., using depth information, altimeter, and speed through the water) and incoming imagery and environmental sensor data are time-stamped and deposited directly on the edge server. While the connection from ISIIS-3 control computer to the switch is rated at 10 GbE, the connection from the switch to the edge server is a 100 GbE active optics cable (AOC) to allow for additional I/O for running the pipeline, offloading data, and sending data to the visualization display.

(Figure 2). This allowed us to deploy a Dell S4148F switch with 10 GbE, 40 GbE and 100 GbE ports to support a large range of devices that needed to be connected to the edge server. SFP+ to RJ45 transceiver modules were used to allow laptops and other devices to connect to the isolated network. The DHCP server was configured to have known hosts with fixed addresses to best support services that relied on being on the same IP upon reboots. SAMBA services were used to allow the ISIIS-3 control computer (running Windows 10) to directly save incoming video data to the edge server. An additional Ubuntu 20.04 desktop was used to control the processing pipeline on the edge server through SSH, and a MacOSX desktop was used for running the webserver that visualized real-time classified plankton information (e.g., length of segmented particles and plankton as well as taxonomic identity), using the Python API 2.0 HeavyDB interface (Schmid et al., 2023a; see reference to heavyDB). A 10-m 100 GbE QSFP28 AOC cable allowed the set-up of the edge server in a separate temperature-controlled server room on the ship, removing the edge server fan noise from the science labs while retaining an extremely fast connection and leaving enough I/O for simultaneous writing of incoming imagery, data offload, pipeline control, and sending of data to a database. The ISIIS-3 control computer only supported a 10GbE network card, but over the SAMBA mounts the ISIIS-3 control computer was able to write to the edge server at ~400MB/s, about twice the throughput that was needed for the raw imagery, leaving plenty of I/O on the drives of the edge server to simultaneously process data.

3 Methods

3.1 Image processing pipeline

The image processing pipeline controller scripts are primarily written in Python 3 and call binaries that need to be compiled first (Figure 3). Segmentation (https://github.com/paradom/Threshold-MSER/tree/spectra-dev) and classification binaries are provided in the zenodo pipeline repository for this paper (http://dx.doi.org/10.5281/zenodo.7739010). Incoming video files are automatically ingested into the image processing pipeline by the automate.py script monitoring the incoming data folder (Figure 3). Incoming AVI files are segmented via threshold-MSER (T-MSER; Panaïotis et al., 2022) using the CPU cores of the edge server (Figure 3). T-MSER is optimized for multithreading and general speed due to the volume of data generated by the two ISIIS-Deep Particle Imager (DPI) cameras. Multithreading of segmentation and classification is controlled by the OpenMP Python library and based on available resources. On the edge server with 40 cores, 20 processes can be run in parallel. After the flat-fielding of individual frames, T-MSER uses a signal-to-noise ratio (SNR) switch, after which low noise frames are directly segmented using Maximally Stable Extremal Regions (MSER, Matas et al., 2004; Bi et al., 2015; Cheng et al., 2019), and high noise frames are first pre-processed with a thresholding approach before applying MSER. T-MSER was written in C++. The lower size cutoff for the segmentation, determining which size segments (i.e., plankton) are retained, can be set to the

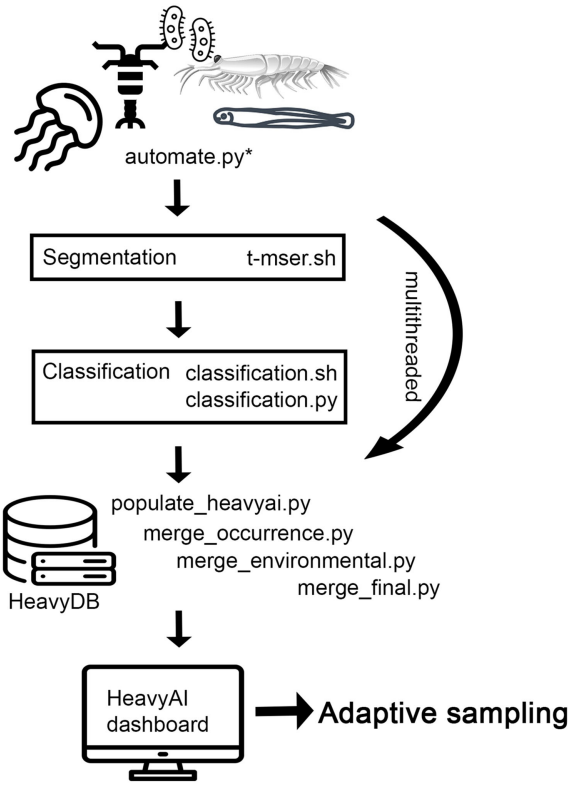


FIGURE 3
Pipeline schematic depicting the imagery data processing pipeline deployed at sea. The automate.py script controls all subsequent processes, including ingestion of imagery into segmentation and classification, merging of the different data products, and upload into HeavyDB. The HeavyAI dashboard monitors the HeavyDB and visualizes new data (e.g., depth stratified plankton identifications) as they become available. The dynamic and interlinked figures in the dashboard are then used for adaptive sampling. Flowchart text with file extensions depicts all the files necessary to run the pipeline, which can be found in the online repository. Text without file extensions describes larger concepts and gives context.

desired value based on the study's objectives; here we used 49 pixels of object area as the lower size cutoff for retention of segments.

As soon as AVIs are segmented automate.py starts the classification process on these segments using a sparse Convolutional Neural Net (sCNN, Graham et al., 2015; Luo et al., 2018; Schmid et al., 2021). The edge server's NVIDIA T4 GPU (Figure 3) supported four classification processes running in parallel. The sCNN was previously trained on an image library containing 170 classes of particles and plankton from the NCC, until the error rate of the classifier plateaued at ~ 5% after 399 epochs. After applying the classifier to new imagery, a random subset of images was classified by two human annotators and compared with the automated identifications to create a confusion matrix. Based on the confusion matrix information (e.g., false positives and true positives) and the known underlying assigned probabilities per image given by the sCNN, we used probability filtering (Faillettaz et al., 2016) to remove very low probability images from the dataset that lead to false positives and false negatives. Using LOESS modeling, we established at which assigned probability a cutoff had to be made to achieving 90% predictive accuracy for the taxon. Removal of these low-confidence images retains true spatial distributions (Faillettaz et al., 2016). The process and accuracies are described in more detail in previously published work (Briseño-Avena et al., 2020b; Schmid et al., 2020; Swieca et al., 2020; Schmid et al., 2021; Greer et al., 2023; Schmid et al., 2023b). The pipeline described here is open-sourced at: http://dx.doi.org/10.5281/zenodo.7739010.

3.2 Database and webserver visualization

Ship data (e.g., GPS feed), ISIIS-3 environmental sensor data (e.g., pH, dissolved oxygen), plankton size measurements, and classification probabilities are merged based on microsecond-accurate timestamps by the populate_heavyai.py script and its subroutines (Figure 3). The same script also uploads merged data into the HeavyDB database as soon as they become available. A heavy.ai dashboard that is linked to HeavyDB can then visualize the data in an immersive way, enabling data interpretation and adaptive sampling.

4 Results

4.1 At-sea processing with the edge server

In July 2022, ISIIS-3 was towed along six transects off the WA and OR coasts with each transect ranging from 8 to 14 h long. During these tows, ISIIS-3 imaged plankton ranging from small phytoplankton and protists, to crustaceans, gelatinous plankton such as salps and appendicularians, and larval fishes. These organisms spanned a large size range and differed significantly in their body form (e.g., fragile gelatinous plankton vs hard-shelled crustaceans, Figure 4). By imaging these different organisms in a non-invasive way, we obtained data on their overall distribution and abundance across multiple scales, as well as insights into their natural behaviors and orientations in the water column and potential predators-prey relationships (Ohman, 2019). Along the six transects, 36 TB of data were collected from the two ISIIS-DPI cameras, totaling over 120 h of imagery (60 h per camera).

T-MSER segmentation on the edge server's 40 CPUs took 1.1 mins per 50 sec of video data, while classification on the T4 GPUs took an additional 2.65 mins on average, bringing the total time lag between data collection and having classified results to 3.75 mins. The speed of the pipeline becomes even more apparent when taking into account that an AVI contains between 23,000 and 225,000 segments of particles and organisms, depending on the biological productivity (Panaïotis et al., 2022). Especially dense phytoplankton layers led to longer segmentation and classification times. With that in mind, segmentation and classification together can take between 2.5 - 5 min per 50 sec AVI.

4.2 Database and visualization of plankton classifications for adaptive sampling

The HeavyDB database updated automatically as new data were classified, and included the taxonomic identifications and lengths of

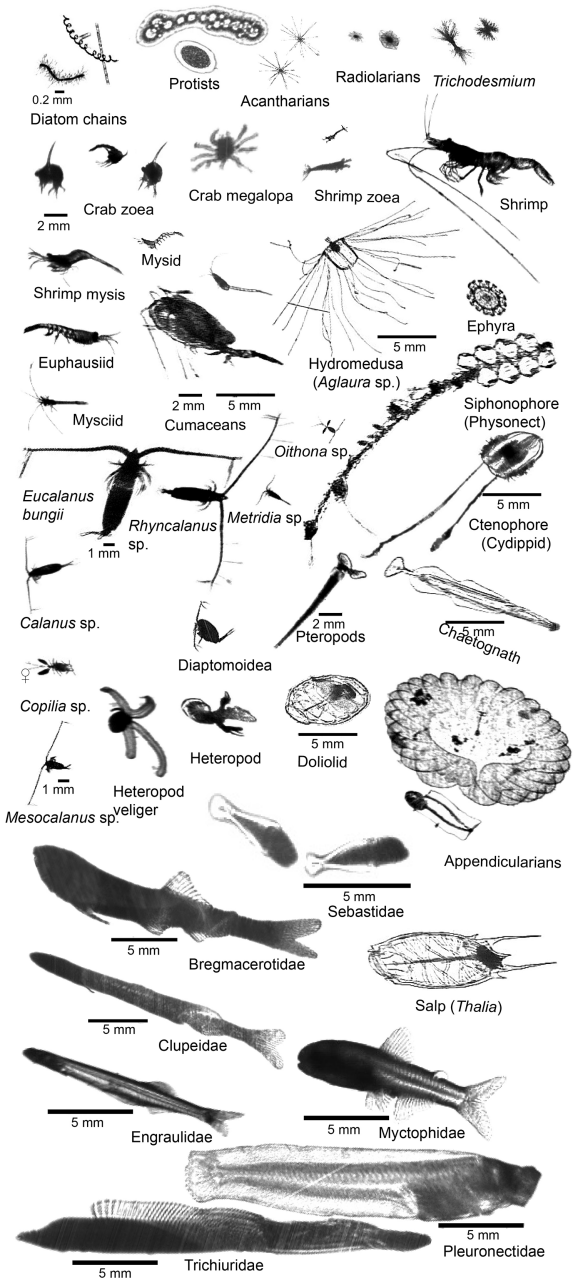


FIGURE 4
ISIIS-DPI images of key taxa in the Northern California Current including primary producers, protists, crustaceans, cnidarians, ctenophores, echinoderms, heteropods, pteropods, chaetognaths, pelagic tunicates, and larval fishes.

each detected object together with their environmental data (e.g., pH, oxygen), as well as GPS location from ship sensors. Database and heavy.ai dashboard were very responsive, running on the edge server's 512 GB memory and the NVMe flash storage. Hence, visualization of data on the heavy.ai dashboard was smooth and updated quickly based on the user selections (Figure 5). The dashboard can be customized by the user to show different data presentations. Shown here are standard features – number of classified images used in the data presentation, number of unique taxa classified, allocation of classified images across taxa, sampling location, as well as location specific sampling depth (note, in this case, our transect ran east-west along a constant latitude). The user can select which taxa (or all) to display – in this example, we show the vertical distribution (in 2-m depth bins) of all taxa combined. We also show the size spectrum of all classified segments across 76 bins of major axis segmented image size (i.e., based on number of pixels). Other data presentations can easily be developed by the user by clicking "add chart" on the dashboard. Data presentation is updated continually as new classifications are completed. Heavy.ai dashboard graphics are dynamic and interlinked so that selection of a taxon, size range, or time interval, leads to all other plots defaulting to that sub-selection. For instance, selection of *Oithona* sp. copepods in the taxa overview leads to the size spectrum and 3-D vertical distribution plots showing only data of *Oithona* sp. copepods. Multiple simultaneous selections are possible and a powerful and intuitive tool for adaptive sampling.

5 Discussion

Using the edge server for live classification of plankton imagery yielded bountiful data for exploration during the cruise and for adaptive sampling. Use cases for adaptive sampling in biological oceanography that have the potential to transform oceanography include on-the-fly and fast detection of species of interest, detection and resampling of thin layer associated organisms, as well as high spatial resolution adaptive sampling of taxa present in, or at the interface of, environmental features of high importance such as low oxygen zones on the NCC shelf.

5.1 Example applications for adaptive sampling

Access to real-time or near real-time taxon-specific distribution and abundance data is novel in most oceanographic studies, particularly access to very detailed spatial and vertical resolution. With such data in hand, while at sea, the researcher can be responsive to short-lived events (e.g., thin layers, sub-mesoscale eddies, other aggregative features), to specific taxa that might be ephemeral or highly patchy, and to environmental conditions that are of particular interest (e.g., low oxygen). With the ability to identify such features or taxa of interest while still at sea, the researcher can adapt their sampling to a more specific target. Below are several examples where sampling could be adapted in response to the detection of specific features or events.

5.1.1 Vertical migration

Diurnal vertical migration (DVM) is a well-known, but often challenging process to adequately sample biologically. Acoustic echograms can help visualize the movement of reflective organisms, but actual species composition of the observed acoustic signal requires *in situ* sampling. While a plankton net might be able to verify the dominant species present in such a feature, it will not provide detailed vertical distribution data of different species. Fine spatial separation may occur under some scenarios as different species may swim/rise at



FIGURE 5
(A) The HeavyAI dashboard displayed on the adaptive sampling display. The user can add and delete different figure types. Clockwise from the upper left, are: the vertical distribution of plankton counts in the water column, the relative abundance of taxa (as a pie chart), the geolocation of samples (map), the size distribution of plankton taxa (histogram), and the vertical distribution of plankton taxa with longitude. Selecting a swath of vertical distribution or a specific taxon in the pie chart automatically adapts all other figures to the sub-selection, for instance only showing a certain taxon – multiple sub-selections at the same time are possible (e.g., adapting all figures to only show *Oithona* sp. copepods in the top 20 meters that have a certain size). The HeavyAI dashboard monitors the underlying HeavyDB for new incoming data to display. **(B)** This setup lends itself to near real-time data exploration and adaptive sampling.

different speeds, and determination, let alone verification of that pattern is difficult at best with only acoustic data (Figure 6). Towing an imaging system such as ISIIS-3 with near-real time data output, can enable a detailed biological survey of the feature, even as it is rising or falling in the water column.

5.1.2 Thin layers and other patchy features

Algal thin layers are often highly transient in location and persistence. While their presence may be predictable in some situations (e.g., Greer et al., 2013; Greer et al., 2020; McManus et al., 2021), actual encounter of them may be a chance occurrence, and indication of their presence may be vague (e.g., Chl *a* signal appearing highly noisy). Verifying the presence, and detailing the vertical distribution of organisms associated with a thin layer can only be done with focused vertical sampling. Real-time high resolution imagery data can more accurately verify the presence of a thin layer and its various species constituents, and then can be utilized in developing an adaptive sampling plan to more fully resolve the dimensions and species interactions associated with the thin layer.

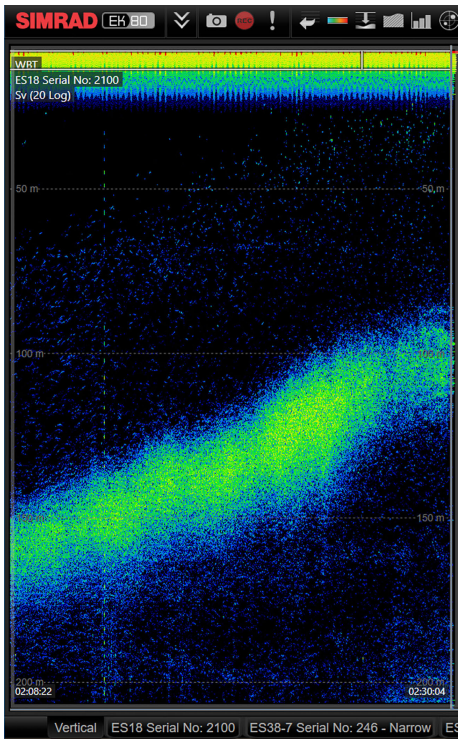


FIGURE 6
A snapshot from the EK80 18 kHz backscatter signal showing evidence of plankton diel vertical migration to surface waters during early evening hours. Time is on the x-axis, depth on the y-axis. Combining live observations from the EK80 with live ISIIS-DPI imagery and the heavyAI dashboard enables a new way of adaptive sampling by being able to pinpoint the taxa comprising such diel migration patterns.

Vertically and spatially discrete aggregations of other organisms are not uncommon (Robinson et al., 2021), though difficult to predict. Their presence may be associated with a specific life stage, or in response to certain biological or physical features and their relative importance (i.e., as a predator or prey source) may depend on the extent of the patch (or bloom). For example, small patches of dense hydromedusae aggregations (Figure 7), which can exert substantial predation pressure on larval fishes and copepods (Corrales-Ugalde and Sutherland, 2021; Corrales-Ugalde et al., 2021), are difficult to sample with nets. As with other aggregations, when hydromedusae are identified though *in-situ* imaging and real-time AI at sea, researchers have the potential to adjust sampling efforts to resolve the dimensions and density of patches.

5.1.3 Specific environmental conditions of high interest

As with focused sampling around biological aggregations, adaptive sampling around specific oceanographic conditions can reveal novel biological patterns and associations. Follow-up sampling at various physical interfaces, as identified by other sensors, might reveal changes in organism distributions warranting further study. For example, vertical or horizontal frontal features detected by Acoustic Doppler Current Profilers (ADCP; Figure 8), might suggest broad, then more fine-tuned sampling as real-time data analyses reveal spatial biological patterns. Eddie fronts (potentially detected by ADCP) are prime examples for where adaptive re-sampling of the eddy's interface could provide valuable insight into the taxonomic make-up of eddy, interface, and exterior water masses (Schmid et al., 2020).

Finally, coupling physical and optical sensors can enhance adaptive sampling capability. On the NCC shelf, in particular, low oxygen upwelled water can quickly become further hypoxic when primary productivity decays after phytoplankton blooms (Chan et al., 2008). Such low oxygen zones are increasing in frequency and duration and have become an emerging threat to fisheries (Chan et al., 2008; Chan et al., 2019) that can lead to substantial financial loss. Sensors on imaging systems can detect such low oxygen zones (Figure 9) and using the imager, these low oxygen waters can be re-sampled on transects passing from normoxic waters, through the interface, and into the core of hypoxic waters. Near-real-time processing can detect the expected and unexpected presence of different taxa, which can lead to new insights and hypotheses. For example, in 2016, anchovy larvae were imaged in low oxygen waters (Briseño-Avena et al., 2020b) on the Newport Hydrographic Line, a transect sampled since 1961 (Peterson and Miller, 1975).

In combination, the examples presented here are a considerable advancement in our ability to find, identify, and thoroughly sample ephemeral and other hard-to-detect features in the ocean. Adaptive sampling using cutting edge technology is critical to expand our understanding of the processes that are driving ocean biology.

5.2 Processing speeds

The edge server's NVMe flash drives and CPU succeeded in segmenting the incoming.avi video files almost at 1:1 ratio of collection time vs processing time. A single NVIDIA T4 GPU with 16 GB memory was able to classify data in four parallel instances, adding on average another 2.65 mins for classification of each AVI. While the achieved processing times were good and



FIGURE 7
A snapshot of ISIIS-DPI imagery as the sled is towed along a transect in the southern California Current. Dense patches of organisms, in this case hydromedusae, can be observed and re-sampled to identify the extent of patches and layers. Using near real-time classification with an edge server enables the identification and quantification of dense patches. The layer shown here spans 1.17 m from edge to edge.

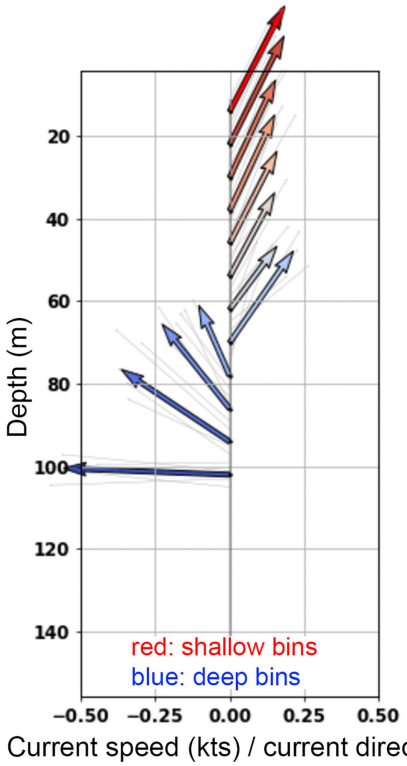


FIGURE 8
A snapshot of an ADCP vector diagram as seen on the real-time readout on research vessels. Using real time classification of encountered plankton in conjunction with ADCP data allows the immediate re-sampling of ocean conditions of high importance, such as vertical and horizontal fronts. In this ADCP vector diagram, surface waters (5-60 m) are characterized by distinct northeastward flow, while at depths below 75 m water is moving in a northwest to west direction. Using readouts from the heavyAI dashboard, the ISIIS-3 imager can be towed specifically at the interface of such divergent flows in order to collect the most insightful data on taxa distributions, potential predator-prey interactions facilitated by such features, as well as behavioral observations. The y-axis shows depth; however, each arrow has a directional and speed component. Colors are not quantitative but indicate shallow and deep bins. The direction of the arrows indicates 360 deg direction, with arrows upward indicating "North".

within our expectations, we envision more powerful hardware in conjunction with even more specialized software to segment incoming AVIs at a ratio of 1:0.5 or faster – and cutting down on classification time in a similar way, in order to go from near real-time processing and display of data to real-time classification and display. Depending on the detected oceanographic features or *a priori* features the user wants to investigate with regards to the distribution of taxa, the ability to see which taxa are present with a 1 min time lag vs a 5 mins time lag, likely makes a big difference.

5.3 Implementing the adaptive sampling pipeline with other imaging system setups and edge servers

The pipeline code and workflow described here were designed with the idea of being agnostic to the imaging system used as well as the specific edge server available. For instance, while our specific setup receives large quantities of data through a fiber optic cable that are then ingested into the pipeline on the edge server, this is by no means a necessary pathway. The output of any imaging system could be used with this setup by similarly creating network drives on the imaging system's data collection computer, pointing to the edge server for writing files and immediate processing – how the imagery gets to the edge server is of little importance as long as the time lag between collecting the data and starting to process is minimized. This also means that while the presented pipeline is targeting live data-feed imaging systems, one could easily take the setup described here and supply data from profilers that do not transmit data live (e.g., the Underwater Vision Profiler 6), as soon as the data from a profile is retrieved. In that context, a user can also replace the segmentation and classification described here with an instant segmentation approach such as the You Only Look Once (YOLO; Jiang et al., 2022) algorithm or similar. The idea of an edge server is to have powerful hardware (i.e., CPU, GPU, memory, storage) in a relatively low power consumption package that has a small footprint and is ruggedized. There are a diversity of edge servers available on the market that can be bought or home-built

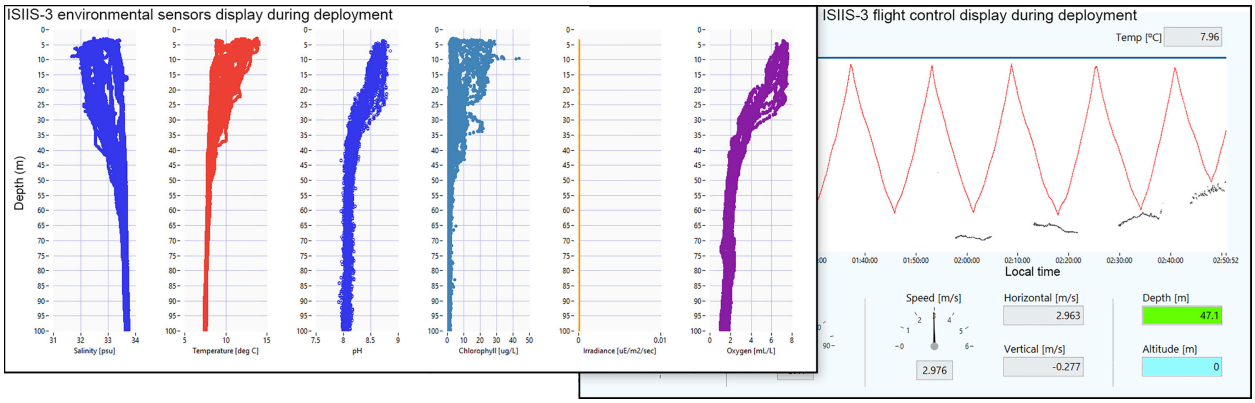


FIGURE 9
ISIIS-3 control display during a transect on the Heceta Head line (43.98° N) off Oregon, with environmental data plotted on the left (e.g., dissolved Oxygen as low as < 1 ml L^{-1} at 100 m depth). The right panel shows the undulating flight pattern (red line) and demonstrates ISIIS-3's ability to sample hypoxic waters at near bottom depths (blue points are the seafloor as indicated by the altimeter).

and that could be used instead of the one used here. When switching to a GPU-based YOLO or Mask R-CNN (He et al., 2017) object detection, the user would be less reliant on CPUs and thus might prefer a setup with fewer CPUs while swapping in several more powerful GPUs instead.

5.4 Conclusion

ISIIS-3 in conjunction with a deep learning pipeline deployed on an edge server at sea is a powerful combination for adaptive sampling, reducing lag between data collection and addressing on ecological questions, as well as for scientific discussions with cruise participants. Several applications of adaptive sampling were presented that have the potential to be transformative for oceanographic research, including *in-situ* target species identification, and HAB thin layer characterization. In the northern California Current, where hypoxia and ocean acidification are endangering commercially important taxa such as Dungeness crab and hence the livelihood of communities, adaptive sampling of taxa distributions in such features could prove a very effective tool for better understanding the responses of such taxa to environmental disturbances.

Data availability statement

The datasets presented in this study can be found in the online repositories below: Processing pipeline open-sourced at: http://dx.doi.org/10.5281/zenodo.7739010; NSF's BCO-DMO: https://www.bco-dmo.org/project/855248; R2R program: https://www.rvdata.us/search/vessel/Langseth.

Ethics statement

The animal study was reviewed and approved by Oregon State University's Institutional Animal Care and Use Committee (IACUC). Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

Author contributions

MS performed analyses and wrote the initial manuscript; MS, CS, SS, and RC conceptualized hypotheses and research questions; DD, MD, and CS wrote and deployed pipeline code on the edge server, MS and CS supervised pipeline development; MS, SS, and RC wrote grant proposals; MS, DD, and RC collected data; CG, CC, MS, and RC designed ISIIS-3, CG and CC were responsible for ISIIS-3 engineering. All authors contributed to the article and approved the submitted version.

Funding

Support for this study was provided by NSF OCE-1737399, NSF OCE-2125407, NSF RISE-1927710, and NSF XSEDE/ACCESS OCE170012.

Acknowledgments

We thank current and former Oregon State University, Hatfield Marine Science Center, Plankton Ecology Lab members who helped collect ISIIS-3 data during the R/V *Langseth* cruise: Elena Conser, Luke Bobay, Jami Ivory, and Megan Wilson. Jassem Shahrani from Sixclear Inc coded the ISIIS-3 control software using the JADE application development environment and was very helpful whenever questions came up. Sergiu Sanielevici and Roberto Gomez from the Pittsburgh Supercomputing Center were instrumental in the development of previous iterations of the pipeline, which later allowed successful porting of components to the edge server. We thank David A. Jarvis at Western Digital for working with us and providing access to the edge server.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The authors declare that the edge server was donated to Oregon State University by Western Digital, and that the results presented here are in no way influenced by this donation. ISIIS-3 was procured from Bellamare LLC. No other commercial or financial relationships that could be construed as a potential conflict of interest are present.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Bi, H., Guo, Z., Benfield, M. C., Fan, C., Ford, M., Shahrestani, S., et al. (2015). A semi-automated image analysis procedure for *In situ* plankton imaging systems. *PloS One* 10, e0127121. doi: 10.1371/journal.pone.0127121

Bi, H., Song, J., Zhao, J., Liu, H., Cheng, X., Wang, L., et al. (2022). Temporal characteristics of plankton indicators in coastal waters: high-frequency data from PlanktonScope. *J. Sea Res.* 189, 102283. doi: 10.1016/j.seares.2022.102283

Briseño-Avena, C., Prairie, J. C., Franks, P. J. S., and Jaffe, J. S. (2020a). Comparing vertical distributions of chl-a fluorescence, marine snow, and taxon-specific zooplankton in relation to density using high-resolution optical measurements. *Front. Mar. Sci.* 7. doi: 10.3389/fmars.2020.00602

Briseño-Avena, C., Schmid, M. S., Swieca, K., Sponaugle, S., Brodeur, R. D., and Cowen, R. K. (2020b). Three-dimensional cross-shelf zooplankton distributions off the central Oregon coast during anomalous oceanographic conditions. *Prog. Oceanogr.* 188, 102436. doi: 10.1016/j.pocean.2020.102436

Chan, F., Barth, J., Kroeker, K., Lubchenco, J., and Menge, B. (2019). The dynamics and impact of ocean acidification and hypoxia: insights from sustained investigations in the northern California current Large marine ecosystem. *Oceanography* 32, 62–71. doi: 10.5670/oceanog.2019.312

Chan, F., Barth, J. A., Lubchenco, J., Kirincich, A., Weeks, H., Peterson, W. T., et al. (2008). Emergence of anoxia in the California current Large marine ecosystem. *Science* 319, 920–920. doi: 10.1126/science.1149016

Cheng, K., Cheng, X., Wang, Y., Bi, H., and Benfield, M. C. (2019). Enhanced convolutional neural network for plankton identification and enumeration. *PloS One* 14, e0219570. doi: 10.1371/journal.pone.0219570

Corrales-Ugalde, M., Sponaugle, S., Cowen, R. K., and Sutherland, K. R. (2021). Seasonal hydromedusan feeding patterns in an Eastern boundary current show consistent predation on primary consumers. *J. Plankton Res.* 43, 712–724. doi: 10.1093/plankt/fbab059

Corrales-Ugalde, M., and Sutherland, K. R. (2021). Fluid mechanics of feeding determine the trophic niche of the hydromedusa clytia gregaria. *Limnol. Oceanogr.* 66, 939–953. doi: 10.1002/lno.11653

Cowen, R. K., and Guigand, C. M. (2008). *In situ* ichthyoplankton imaging system (ISIIS): system design and preliminary results. *Limnol. Oceanogr. Methods* 6, 126–132. doi: 10.4319/lom.2008.6.126

Faillettaz, R., Picheral, M., Luo, J. Y., Guigand, C., Cowen, R. K., and Irisson, J.-O. (2016). Imperfect automatic image classification successfully describes plankton distribution patterns. *Methods Oceanogr.* 15, 60–77. doi: 10.1016/j.mio.2016.04.003

Fossum, T. O., Fragoso, G. M., Davies, E. J., Ullgren, J. E., Mendes, R., Johnsen, G., et al. (2019). Toward adaptive robotic sampling of phytoplankton in the coastal ocean. *Sci. Robotics* 4, eaav3041. doi: 10.1126/scirobotics.aav3041

Graham, B. (2015). Fractional max-pooling. *Arxiv* 1–10. doi: 10.48550/arxiv.1412.6071

Greer, A. T., Boyette, A. D., Cruz, V. J., Cambazoglu, M. K., Dzwonkowski, B., Chiaverano, L. M., et al. (2020). Contrasting fine-scale distributional patterns of zooplankton driven by the formation of a diatom-dominated thin layer. *Limnol. Oceanogr.* 65, 2236–2258. doi: 10.1002/lno.11450

Greer, A. T., Cowen, R. K., Guigand, C. M., McManus, M. A., Sevadjian, J. C., and Timmerman, A. H. V. (2013). Relationships between phytoplankton thin layers and the fine-scale vertical distributions of two trophic levels of zooplankton. *J. Plankton Res.* 35, 939–956. doi: 10.1093/plankt/fbt056

Greer, A. T., Schmid, M. S., Duffy, P. I., Robinson, K. L., Genung, M. A., Luo, J. Y., et al. (2023). *In situ* imaging across ecosystems to resolve the fine-scale oceanographic drivers of a globally significant planktonic grazer. *Limnol. Oceanogr.* 68, 192–207. doi: 10.1002/lno.12259

He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). "Mask r-CNN," in *2017 Ieee Int Conf Comput Vis Iccv*. 2980–2988. doi: 10.1109/iccv.2017.322

Hickey, B., and Banas, N. (2008). Why is the northern end of the California current system so productive? *Oceanography* 21, 90–107. doi: 10.5670/oceanog.2008.07

Hopcroft, R. R., Roff, J. C., and Bouman, H. A. (1998). Zooplankton growth rates: the larvaceans appendicularia, fritillaria and oikopleura in tropical waters. *J. Plankton Res.* 20, 539–555. doi: 10.1093/plankt/20.3.539

Irisson, J.-O., Ayata, S.-D., Lindsay, D. J., Karp-Boss, L., and Stemmann, L. (2021). Machine learning for the study of plankton and marine snow from images. *Annu. Rev. Mar. Sci.* 14, 1–25. doi: 10.1146/annurev-marine-041921-013023

Jiang, P., Ergu, D., Liu, F., Cai, Y., and Ma, B. (2022). A review of yolo algorithm developments. *Proc. Comput. Sci.* 199, 1066–1073. doi: 10.1016/j.procs.2022.01.135

Lombard, F., Boss, E., Waite, A. M., Vogt, M., Uitz, J., Stemmann, L., et al. (2019). Globally consistent quantitative observations of planktonic ecosystems. *Front. Mar. Sci.* 6. doi: 10.3389/fmars.2019.00196

Luo, J. Y., Irisson, J., Graham, B., Guigand, C., Sarafraz, A., Mader, C., et al. (2018). Automated plankton image analysis using convolutional neural networks. *Limnol. Oceanogr. Methods* 16, 814–827. doi: 10.1002/lom3.10285

Luo, J. Y., Stock, C. A., Henschke, N., Dunne, J. P., and O'Brien, T. D. (2022). Global ecological and biogeochemical impacts of pelagic tunicates. *Prog. Oceanogr.* 205, 102822. doi: 10.1016/j.pocean.2022.102822

Matas, J., Chum, O., Urban, M., and Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image Vision Comput.* 22, 761–767. doi: 10.1016/j.imavis.2004.02.006

McManus, M., Greer, A., Timmerman, A., Sevadjian, J., Woodson, C., Cowen, R., et al. (2021). Characterization of the biological, physical, and chemical properties of a toxic thin layer in a temperate marine system. *Mar. Ecol. Prog. Ser.* 678, 17–35. doi: 10.3354/meps13879

Ohman, M. D. (2019). A sea of tentacles: optically discernible traits resolved from planktonic organisms in situ. *Ices J. Mar. Sci.* 76, 1959–1972. doi: 10.1093/icesjms/fsz184

Ohman, M. D., Davis, R. E., Sherman, J. T., Grindley, K. R., Whitmore, B. M., Nickels, C. F., et al. (2019). Zooglider: an autonomous vehicle for optical and acoustic sensing of zooplankton. *Limnol. Oceanogr. Methods* 17, 69–86. doi: 10.1002/lom3.10301

Orenstein, E. C., Ratelle, D., Briseño-Avena, C., Carter, M. L., Franks, P. J. S., Jaffe, J. S., et al. (2020). The Scripps plankton camera system: a framework and platform for *in situ* microscopy. *Limnol. Oceanogr. Methods* 18, 681–695. doi: 10.1002/lom3.10394

Panaïotis, T., Caray–Counil, L., Woodward, B., Schmid, M. S., Daprano, D., Tsai, S. T., et al. (2022). Content-aware segmentation of objects spanning a Large size range: application to plankton images. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.870005

Peterson, W. T., and Miller, C. B. (1975). Year-to-year variations in the planktology of the Oregon upwelling zone. *Fish Bull.* 73, 642–653.

Ratnarajah, L., Abu-Alhaija, R., Atkinson, A., Batten, S., Bax, N. J., Bernard, K. S., et al. (2023). Monitoring and modelling marine zooplankton in a changing climate. *Nat. Commun.* 14, 564. doi: 10.1038/s41467-023-36241-5

Reese, D. C., and Brodeur, R. D. (2006). Identifying and characterizing biological hotspots in the northern California current. *Deep Sea Res. Part Ii Top. Stud. Oceanogr* 53, 291–314. doi: 10.1016/j.dsr2.2006.01.014

Robinson, K. L., Sponaugle, S., Luo, J. Y., Gleiber, M. R., and Cowen, R. K. (2021). Big or small, patchy all: resolution of marine plankton patch structure at micro- to submesoscales for 36 taxa. *Sci. Adv.* 7, eabk2904. doi: 10.1126/sciadv.abk2904

Schmid, M. S., Cowen, R. K., Robinson, K., Luo, J. Y., Briseño-Avena, C., and Sponaugle, S. (2020). Prey and predator overlap at the edge of a mesoscale eddy: fine-scale, in-situ distributions to inform our understanding of oceanographic processes. *Sci. Rep.* 10, 921. doi: 10.1038/s41598-020-57879-x

Schmid, M. S., Daprano, D., Damle, M. M., Sullivan, C., Sponaugle, S., and Cowen, R. K. (2023a). Code for segmentation, classification, databasing, and visualization of in-situ plankton imagery on edge servers at sea. *Zenodo*. doi: 10.5281/zenodo.7739010

Schmid, M. S., Daprano, D., Jacobson, K. M., Sullivan, C., Briseño-Avena, C., Luo, J. Y., et al. (2021). A convolutional neural network based high-throughput image classification pipeline - code and documentation to process plankton underwater imagery using local HPC infrastructure and NSF's XSEDE. *Zenodo*. doi: 10.5281/zenodo.4641158

Schmid, M. S., Sponaugle, S., Sutherland, K., and Cowen, R. K. (2023b). Drivers of plankton community structure in intermittent and continuous coastal upwelling systems–from microscale in-situ imaging to large scale patterns. *bioRxiv* 1–43. doi: 10.1101/2023.05.04.539379

Song, J., Bi, H., Cai, Z., Cheng, X., He, Y., Benfield, M. C., et al. (2020). Early warning of noctiluca scintillans blooms using in-situ plankton imaging system: an example from dapeng bay, P.R. China. *Ecol. Indic.* 112, 106123. doi: 10.1016/j.ecolind.2020.106123

Stankiewicz, P., Tan, Y. T., and Kobilarov, M. (2021). Adaptive sampling with an autonomous underwater vehicle in static marine environments. *J. Field Robot* 38, 572–597. doi: 10.1002/rob.22005

Swieca, K., Sponaugle, S., Briseño-Avena, C., Schmid, M., Brodeur, R., and Cowen, R. (2020). Changing with the tides: fine-scale larval fish prey availability and predation pressure near a tidally modulated river plume. *Mar. Ecol. Prog. Ser.* 650, 217–238. doi: 10.3354/meps13367

Wiebe, P. H., and Benfield, M. C. (2003). From the hensen net toward four-dimensional biological oceanography. *Prog. Oceanogr.* 56, 7–136. doi: 10.1016/s0079-6611(02)00140-4

frontiers | Frontiers in Marine Science

TYPE Brief Research Report
PUBLISHED 14 June 2023
DOI 10.3389/fmars.2023.1140645

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Felipe Minuzzi,
Federal University of Santa Maria, Brazil
Jian Zhao,
University of Maryland, College Park,
United States

*CORRESPONDENCE
Chunyong Ma
chunyongma@ouc.edu.cn

RECEIVED 09 January 2023
ACCEPTED 31 May 2023
PUBLISHED 14 June 2023

CITATION
Wang Z, Chen G, Ma C and Liu Y (2023) Southwestern Atlantic ocean fronts detected from the fusion of multi-source remote sensing data by a deep learning model.
Front. Mar. Sci. 10:1140645.
doi: 10.3389/fmars.2023.1140645

COPYRIGHT
© 2023 Wang, Chen, Ma and Liu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Southwestern Atlantic ocean fronts detected from the fusion of multi-source remote sensing data by a deep learning model

Zhi Wang[1], Ge Chen[1,2], Chunyong Ma[1,2*] and Yalong Liu[3]

[1]College of Marine Technology, Ocean University of China, Qingdao, China, [2]Laboratory for Regional Oceanography and Numerical Modelling, Qingdao National Laboratory for Marine Science and Technology, Qingdao, China, [3]Yantai Marine Environment Monitoring Center Station, State Oceanic Administration, Yantai, China

In the Southwestern Atlantic, the Falkland Current intrudes onto the South American shelf, resulting in the meeting of two water masses which are completely different in temperature and dynamic characteristics, thus generating the Southwestern Atlantic Front (SAF). Therefore, the SAF has prominent characteristics of thermal and dynamics. The current ocean front detection is mainly by performing gradient operations on sea surface temperature (SST) data, where regions with large temperature gradients are considered as ocean fronts. The thermal gradient method largely ignores the dynamical features, leading to inaccurate manifestation of SAF. This study develops a deep learning model, SAFNet, to detect the SAF through the synergy of 10-year (2010-2019) satellite-derived SST and sea surface height (SSH) observations to achieve high accuracy detection of SAF with fused thermal and dynamic characteristics. The comparative experimental results show that the detection accuracy of SAFNet reaches 99.45%, which is significantly better than other models. By comparing the frontal probability (FP) obtained by SST, SSH and SST-SSH fusion data respectively, it is proved that the necessity of fusion multi-source remote sensing data for SAF detection. The detection results of fusion data can reflect the spatial distribution of SAF more comprehensively and accurately. According to the meridional variation of FP, the main reason for the seasonal variation of the SAF is the change in its thermal characteristics, and the SAF has stable dynamic characteristics.

KEYWORDS

Southwestern Atlantic fronts, multi-source remote sensing data, deep learning, ocean dynamics, ocean thermodynamics

1 Introduction

The Southwestern Atlantic (SA) mainly refers to the area of the Atlantic between 35°S-60°S and 50°W-70°W, that connects to the Drake Passage. Topographically, the area consists of the South American continental shelf in the northwest and the Argentine basin in the southeast. Due to its location between subtropical waters and the cold waters in the

Southern Ocean, the SA is rich in ocean currents and associated hydrological phenomena, including frontal systems. As shown in Figure 1A, the SA mainly contains three major ocean currents, the Brazil Current, the Falkland Current and the Antarctic Circumpolar Current (ACC). ACC is the strongest cold current in the South Hemisphere. As a tributary of the ACC at Cape Horn, the Falkland Current flows northward along the 1000m isobath and invades into the shelf waters of South American (within the 200m isobath) at about 45°S (Piola et al., 2013). Under the influence of the Brazil Current (a strong warm current), the shelf water on the west side is warmer than the Falkland Current (a strong cold current) on the east side. Since the ocean front refers to the boundary between different water masses in the ocean, the Falklands Current enters the waters of the South American continental shelf, resulting in the meeting of two water masses with completely different temperature and dynamic characteristics, generating the Southwestern Atlantic Front (SAF) (Wang et al., 2021). As an important part of the Southern Ocean Front (Chapman et al., 2020), the SAF has great impacts on the ecological environment, fishery production and material transport in the SA (Lopes et al., 2016). Therefore, it is of great significance to detect the SAF accurately.

In the frontal regions, the properties of water mass change rapidly, which are characterized with enhanced horizontal gradients of temperature, salinity, density, etc (Legeckis, 1979). Therefore, researchers often calculate the gradient magnitude map by gradient operation on satellite remote sensing observations (Text S1), and reserve the area with large gradient by a specific threshold to identify the ocean front (Moore et al., 1999; Dong et al., 2006; Wang et al., 2020). Among them, sea surface temperature (SST) data are widely used for ocean front detection (Freeman et al., 2016). Figures 1B, D, F are display the SST distribution over the SA, the SST gradient magnitude map, and the SST front (Southwestern Atlantic thermal front) obtained from the magnitude map by gradient threshold, respectively. Figure 1B shows that the temperature difference between the two sides of the 200m isobath is obvious. SST gradient magnitude map indicates the magnitude of the temperature gradient, which can reflect the intensity of the front. As can be seen from Figure 1D, the maximum SST gradient magnitude are mainly distributed along the west of the 200m isobaths, which coincides exactly with the spatial distribution of the SST front (yellow zone) in Figure 1F. Therefore, the Southwestern Atlantic thermal front (SST front) is mainly distributed along the South American shelf water on the western side of the 200m isobath. Apart from that, the SAF is a typical "current-induced front" (Wang et al., 2021) and thus has prominent dynamic characteristics. Since sea surface height (SSH) data can be used to represent the dynamic characteristics of ocean phenomena, they have been widely used in the study of dynamic fronts in recent years (Chambers, 2018). Figure 1C shows the SSH distribution over the SA. Different from the SST distribution, the SSH distribution is mainly divided by the 1000m isobath, which is exactly consistent with the pathway of the Falkland Current. The invasion of the South American shelf water by the Falkland Current along the 1000m isobath leads to the encounter of two water masses with different dynamic characteristics, resulting in a large difference in the SSH, thus generating the SSH front (Southwestern Atlantic dynamic front). According to Figures 1E, G, the Southwestern Atlantic dynamic front is mainly distributed along the 1000m isobath, which is different from the spatial distribution of the Southwestern Atlantic thermal front. It should be emphasized that for two distinct water masses, the fronts formed between them are unique. The reason for the difference in the spatial distribution of thermal and dynamic fronts comes from the different expression of front characteristics (Takahashi and Kawamura, 2005; Liu and Hou, 2012). Thermal front is the expression of the thermal characteristics of SAF, and the dynamic front is the performance of SAF's dynamic characteristics. Both of them are part of SAF. Therefore, to achieve high-precision SAF detection that fuses dynamic and thermal characteristics cannot only rely on SST or SSH but requires the synergy of SST and SSH. Meanwhile, it is challenging to establish a feature association between massive SST and SSH data and accurately identify the SAF from complex feature fusion data (Liu et al., 2021). Traditional gradient-based frontal detection method (Text S1) cannot solve the above problems (Kittler, 1983).

In recent years, deep learning methods, especially convolutional neural networks (CNNs) have shown excellent performance in mining complex rules hidden in multi-source long term series data, and are increasingly applied to the study of various ocean phenomena such as mesoscale eddies, internal waves, and sea ice (Gao et al., 2022; Zhang and Li, 2022; Li et al., 2022a). Since ocean fronts separate water mass classes and neural networks are robust in assigning classes in complex data, edge detection driven by the underlying neural network may be a good way to find fronts (Li et al., 2022b). Compared with traditional frontal detection methods, deep learning methods have advantages in automatic feature extraction and modeling the relationship between multi-source remote sensing data and ocean fronts. This study develops a deep learning model, SAFNet, to perform feature fusion of SST and SSH data spanning 10 years (2010-2019), and extract the SAF from the fusion data. Comparative experiments show that SAFNet can achieve accurate detection of SAF. Finally, by comparing the seasonal frontal probability (FP) derived from SST, SSH and SST-SSH fusion data respectively, the necessity of the fusion data for SAF detection is proved, and a new understanding of the spatiotemporal distribution and seasonal variation of SAF is obtained. Apart from that, the code of the SAFNet will be updated to GitHub: https://github.com/yangxiaomao225/SAFNet.

The rest of the paper is organized as follows. Section 2 introduces the multi-source remote sensing data used to establish the dataset for training and testing the proposed model and the structure of the SAFNet. Some comparative experiments and spatiotemporal distribution of the FP are shown and discussed in Section 3. In the last section, some conclusions are drawn.

2 Data and method

2.1 Data for training and testing the deep learning model

The altimeter data used in this study are generated by Copernicus Marine and Environmet Monitoring Service

10.3389/fmars.2023.1140645

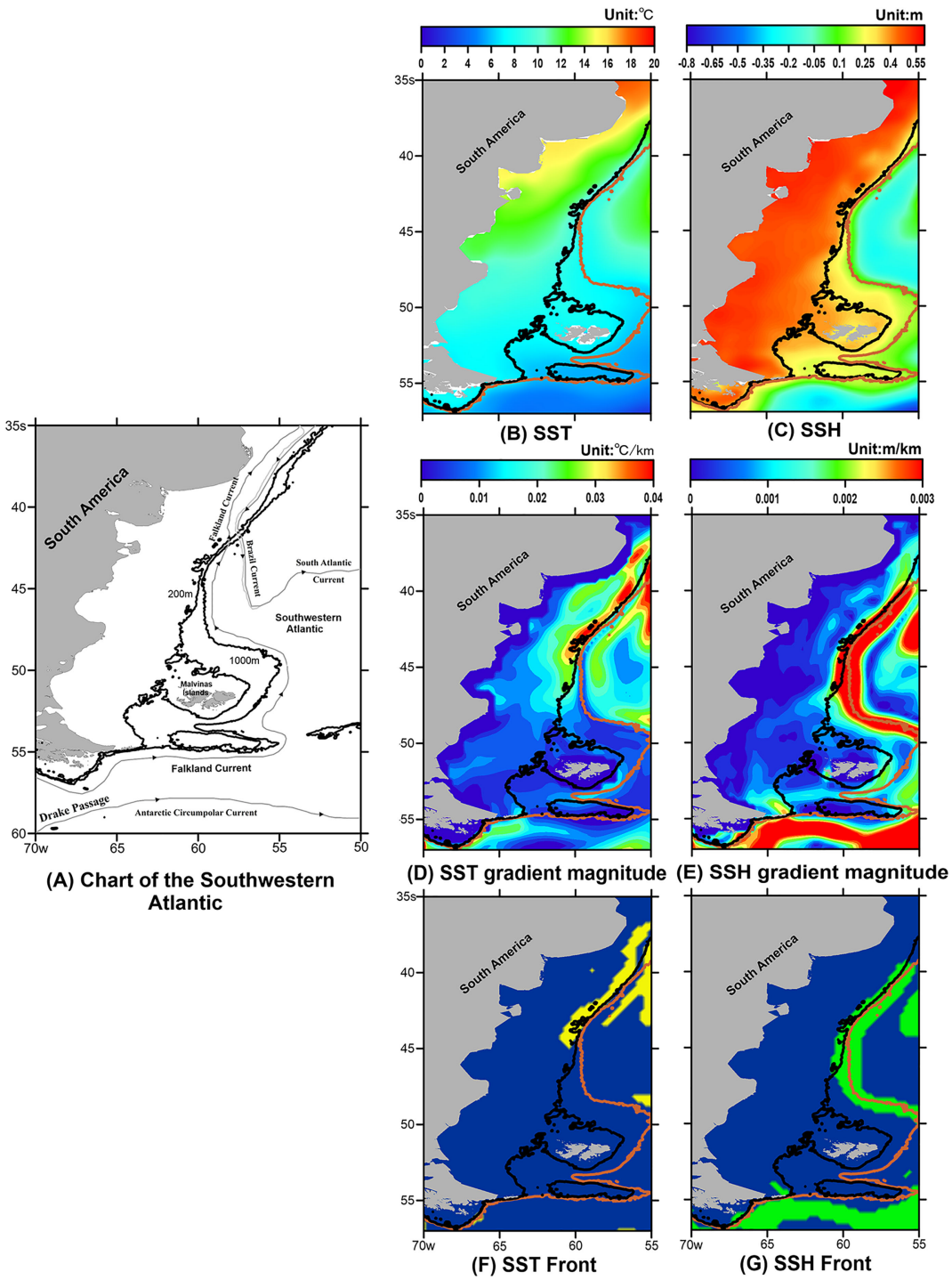


FIGURE 1
Introduction to the SAF background. **(A)** is the chart of the SA. **(B, C)** are 10-year (2010-2019) mean SST and SSH distributions over SA. **(D, E)** are mean SST and SSH gradient magnitude maps obtained from SST and SSH for 2010-2019. **(F, G)** are SST fronts (yellow zone) and SSH fronts (green zone) obtained from the corresponding gradient magnitude map by the threshold. The black and brown solid contours are 200m and 1000m isobaths, respectively.

(CMEMS) using data from the TOPEX/Poseidon, Jason-1, Jason-2, and Envisat missions. The daily gridded SSH data with a spatial resolution of 0.25°×0.25°from January 2010 to December 2019, spanning 10 years. Since SSH products contain two kinds of data, sea level anomalies (SLAs) and absolute dynamic topography (ADT), this study uses ADT, the sum of the time-mean dynamic topography and time-varying SLAs. The daily SST data with a 0.25° spatial resolution refers to the NOAA Optimum Interpolation (OI)

SST product (Reynolds et al., 2007), which is constructed from infrared satellite observations of the Advanced Very High-Resolution Radiometer (AVHRR) and has the same period as the SSH data.

In this study, with the help of the above SSH and SST data, a SAF dataset is established for training and testing the proposed SAFNet. Since the Southwestern Atlantic thermal front (SST front) and dynamic front (SSH front) are part of the SAF and represent different oceanographic characteristics of the SAF, this study first uses the traditional gradient-based front identification method (Text S1) to calculate SST and SSH front through SST and SSH, respectively. Then, the union of the two kinds of fronts is used to represent the SAF in the ideal state, which incorporated the thermal and dynamic characteristics. We describe the creation of the dataset with two samples from the SAF dataset (Text S2; Figure S1).

Thus, SAFs obtained from SST and SSH data in the SA (35°S-57°S, 55°W-70°W, 128×128 pixels) during the period 2010-2018 are used as the training dataset and SAFs obtained from 2019 data are used as the validation dataset in this study. There are 3,287 training samples and 365 validation samples, and pixels in each sample are labeled as "1" or "0" for front or non-front, respectively.

2.2 SAFNet

2.2.1 Overall Structure of the SAFNet

To achieve accurate detection of the SAF by fusing multiple oceanographic features, the proposed deep learning model needs to simultaneously obtain dynamic and thermal characteristics from SSH and SST data, and can accurately detect SAF from these features. Through the traditional gradient-based front detection method (Text S1), we know that the front are the pixels with large gradients. Therefore, the proposed model needs to have two capabilities: 1) The pixels with large gradients in SST and SSH data are extracted and fused as key features. 2) Accurately extract the pixels with large gradients from the fusion features, so as to achieve high-precision detection of SAF. Thus, the SAFNet model consists of two sub-networks: a data fusion network (DFN) to establish the SSH-SST feature fusion relationship and a feature extraction network (FEN) to accurately identify pixels with large gradients from the fusion data for SAF detection. Considering the complex nonlinear relationship between SST and SSH in the SAF, the DFN is developed based on CNNs containing dense connections, and the FEN is developed based on U-Net (Ronneberger et al., 2015), a classical semantic segmentation network in deep learning, as shown in Figure 2. In order to show the detection performance of SAFNet, this study compared SAFNet with two deep learning models on the validation set for SAF detection accuracy. The first one is LinkNet (Chaurasia and Culurciello, 2017), a classical semantic segmentation model, and the other one is D-LinkNet (Zhou et al., 2018), which has achieved excellent results in the field of road recognition. Since they do not contain a data fusion module, to make a fair comparison, this study adds DFN to LinkNet and D-LinkNet so that the two models can fuse the features of SST and SSH like SAFNet.

2.2.2 DFN

Considering that different satellite sensors observe SSH and SST data, the fusion of two multi-source heterogeneous data belongs to multi-modal data fusion. Multi-modal data fusion based on deep learning is widely applied in medical image segmentation. There are three data fusion strategies: input-level fusion, layer-level fusion, and decision-level fusion (Zhou et al., 2019). Unlike the other two data fusion strategies, layer-level fusion can effectively integrate and fully use multi-modal data. In the layer-level fusion strategy, DenseNet (Huang et al., 2017) is the most commonly used network, so an improved DenseNet structure (Dolz et al., 2019) is used as the DFN in this study. The SSH and SST data with 128×128 pixels are imported into two different data streams, respectively, and the features of SSH and SST are extracted through the convolutional layers in the data stream. These features are densely connected between layer pairs in the same data stream and between layer pairs across data streams, and finally a fused data set combining SST and SSH features is obtained, as shown in Figure 2. The mathematical expression of DFN is as follows:

$$\mathbf{x}_l^s = \mathbf{H}_l^s(\mathbf{x}_{l-1}^1, \mathbf{x}_{l-1}^2, \mathbf{x}_{l-2}^1, \mathbf{x}_{l-2}^2, \ldots, \mathbf{x}_0^1, \mathbf{x}_0^2) \quad \mathbf{s=1 \text{ or } 2} \tag{1}$$

where s refers to SSH or SST stream, x_l^1, x_l^2 denote the outputs of the lth layer in SSH and SST streams and H_l^s represents the mapping function of the two data streams at lth layer composed of a convolution layer followed by a batch normalization and a Rectified Linear Unit (ReLU) activation function. Therefore, DFN can alleviate the vanishing gradient problem, introduce implicit deep supervision, and reduce the risk of overfitting tasks with smaller training sets.

2.2.3 FEN

In this study, the FEN is used to accurately detect the SAF based on the output of the DFN. To improve the detection accuracy, the convolutional block attention module (CBAM) (Woo et al., 2018) and dilated convolution layers (DCLs) are integrated into the FEN. FEN uses an encoder-decoder structure. The architecture of FEN includes six parts: input, encoder, center, decoder, concatenations and output. The goal of the encoder is to gradually extract pixels with large gradients from the fusion data through various convolutional layers, to capture the SAF features at different representation levels. The encoder contains one $\mathrm{CNN}_{\mathrm{CBAM}}$ block, six $\mathrm{ResNet}_{\mathrm{CBAM}}$ blocks, and three Max pooling layers. A $\mathrm{CNN}_{\mathrm{CBAM}}$ block is one CNN layer stacking with the CBAM, and a $\mathrm{ResNet}_{\mathrm{CBAM}}$ block is a ResNet unit integrated with the CBAM. The legend on the right in Figure 2 shows that a ResNet unit contains two CNN layers, stacking the CBAM after the second CNN layer. Adding the attention mechanism to the FEN can effectively capture the thermal and dynamic dependencies of the SAF at different scales. CBAM is divided into two modules: channel attention module and spatial attention module. These two modules can generate the feature map's weight matrix in two dimensions. Then the weight matrices are multiplied by the input feature map for adaptive feature refinement so that the network is more targeted to extract features. The center part consists of several DCLs with skip connections. Considering the SAF's narrowness, connectivity, and

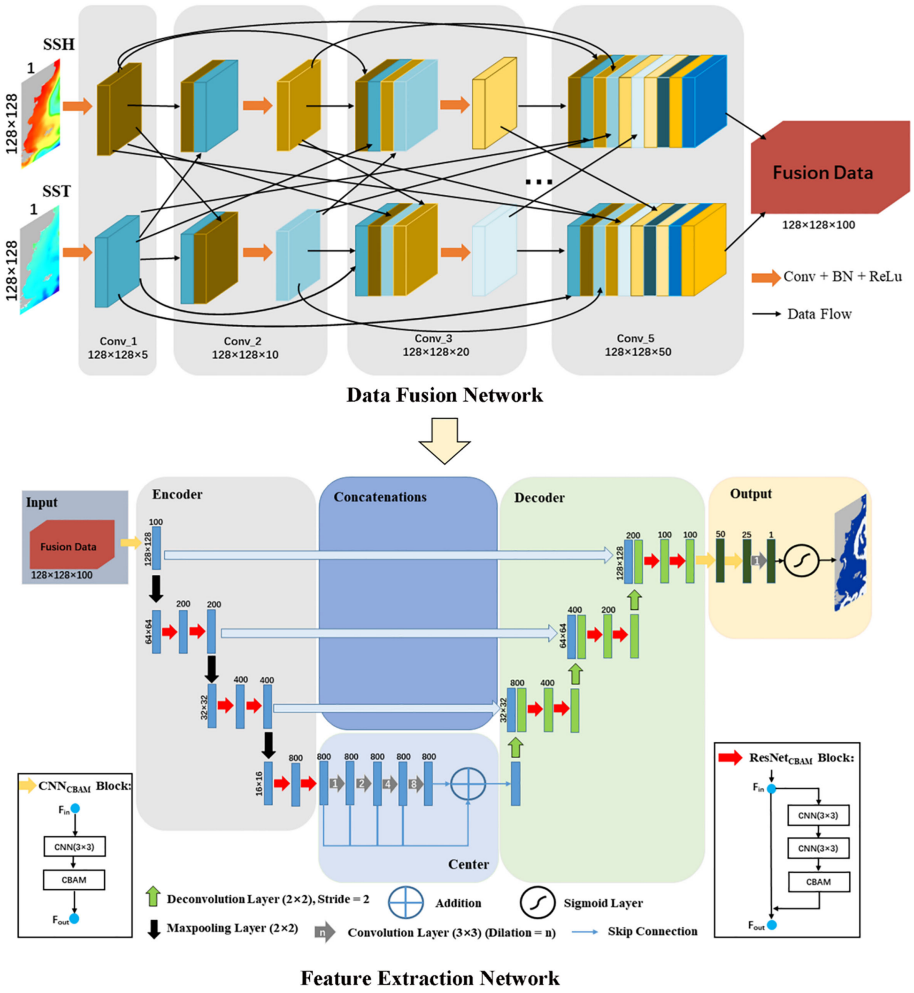


FIGURE 2
The overall structure of the SAFNet which consist of a data fusion network (DFN) and a feature extraction network (FEN).

complexity, it is important to increase the receptive field of feature points in the center part and keep detailed information. DCLs are undoubtedly the best option. The decoder includes four stages, and between each stage, the scale of the feature map is restored by upsampling until the output feature map is the same size as the input data. Six $ResNet_{CBAM}$ blocks are integrated into the decoder to recover the SAF's details accurately. The concatenation fuses the encoder and decoder at the same level, effectively preventing feature loss. The output consists of two CNN_{CBAM} blocks, a 3×3 convolutional layer, and a sigmoid layer, finally outputs a value between [0,1]. If it is greater than 0.5, the pixel is the SAF; Otherwise, it is a non-front.

2.2.4 Loss function

Detection of the SAF is a typical binary classification problem, which only needs to determine which pixels are fronts and which are not. Therefore, the binary cross-entropy loss function (BCELoss) is an effective training method. However, classifying the pixels as front or non-front is a highly imbalanced problem. The weight of BCELoss cannot be set correctly when the specific difference between positive and negative samples is unknown, so the detection effect cannot be guaranteed. The dice coefficient loss function can improve this problem (Zhou et al., 2018). The dice coefficient is a measure function used to evaluate the similarity of two samples, with a larger value indicating more similarity and a smaller dice coefficient loss. Thus, this study defined the loss function as follows:

$$\mathbf{Loss} = \mathbf{1} - \frac{\sum_{i=1}^{N}|\mathbf{P_i} \cap \mathbf{GT_i}|}{\sum_{i=1}^{N}(|P_i| + |GT_i|)} + \sum_{\mathbf{i=1}}^{\mathbf{N}}\mathbf{BCELoss(P_i,GT_i)} \quad (2)$$

$$\mathbf{BCELoss(P,GT)} = -\sum_{\mathbf{i=0}}^{\mathbf{W}}\sum_{\mathbf{j=0}}^{\mathbf{H}}[\mathbf{gt_{ij}}\cdot\mathbf{\log p_{ij}} + \mathbf{(1-gt_{ij})}\cdot\mathbf{\log\,(1-p_{ij})}] \quad (3)$$

where the N is the number of samples, P is the detection result map of the SAFNet, GT is the SAF that has been labeled in the dataset. W is the width of the feature map, H is the height of the feature map, gt is a pixel in GT, and p is a pixel in P.

3 Results and discussion

3.1 Performance of SAFNet

The SAFNet is trained using the NVIDIA RTX A6000 48G GPU and PyTorch deep learning packages. The ADAM optimizer with the learning rate set to 0.01 and the learning rate decay set to 0.1 to optimize the model. The batch size and the number of epochs are set to 32 and 50.

Four metrics are adopted to evaluate the performance of SAFNet and the compared methods (LinkNet, D-LinkNet), i.e., Intersection over Union (IoU), Accuracy, Precision and Recall (Text S3). The objective evaluation results of the three models on the validation set are presented in Table S1. To visually show the differences of each model in SAF detection, the ground truth of four days are arbitrarily selected from the validation set and compared with the detection results of the three models. As shown in Figure 3, SAFNet achieves 99.45% detection accuracy for SAF, which is significantly better than the other two models. Since CBAM and DCLs are integrated in SAFNet, this study proves that CBAM and DCLs can effectively improve the detection accuracy of the proposed model for SAF through ablation experiments (Text S4; Table S2; Figure S2), which further proves that the SAFNet can be used as an effective tool to detect the SAF accurately.

3.2 Spatiotemporal distributions of the SAF

In this study, the comparison experiment and ablation experiment in Section 3.1 fully proves that SAFNet can achieve high-precision detection of SAF. This subsection will prove that the detection results of SAF based on SST-SSH fusion data can reflect the spatial distribution of SAF more comprehensively and accurately than that based on SST or SSH alone. In oceanography, researchers often approximate the spatiotemporal distribution of a front by obtaining its climatological distribution. At present, there are two main methods used to calculate the climatological distribution of fronts. The first one is to calculate the gradient magnitude of the mean SST or SSH data, and use the gradient magnitude map to represent the distribution of fronts. The other one is to use the daily front distribution to calculate the frontal probability (FP), and use the FP distribution to represent the distribution of fronts. The region with large gradient in the gradient magnitude map corresponds to the region with large probability of the FP distribution (Figure S3), so both the gradient magnitude and FP can accurately reflect the spatial distribution of the front (Wang et al., 2020). Since the detection result of SAFNet is the spatial distribution of daily SAF that fuses SST and SSH features, FP is used to represent the climatological mean distribution of the SAF in this study. The FP at each pixel is defined as follows:

$$\textbf{Frontal Probability}=\frac{\mathbf{N_{front}}}{\mathbf{N_{total}}}\times\mathbf{100\%} \quad (4)$$

where N_{front} is the number of times that the pixel is identified as a front, N_{total} is the total number of observation days.

Figure 4 displays the seasonal spatiotemporal distributions of the SAF FP obtained by SST, SSH and the SST-SSH fusion data from 2010 to 2019. By comparing the detection results of the three kinds of data, it is found that the frontal signal of the Southwestern Atlantic thermal front (derived from the SST data) is abundant in the South American shelf waters (within the 200m isobath), while the signal of the Southwestern Atlantic dynamic front (derived from the SSH data) is almost lost in the shelf waters. This is mainly because the current over the shelf does not organize into intensified velocity core or pattern, so that no outstanding SSH gradient exist. However, due to the invasion of the Falkland Current, the shelf water has obvious temperature differences, producing noticeable thermal characteristics (Text S5; Figure S4). Furthermore, the seasonal variation of the thermal front is obvious, which is stronger in summer and weaker in winter. The dynamic front is stable in four seasons and exists all the time. The reasons for this phenomenon come from two aspects: 1) In winter, the increasing surface cooling effects make SSTs uniform, leading to a decrease in the temperature difference between the Falkland Current and shelf water and the disappearance of the thermal front. 2) The Falkland Current intrudes into the shelf water all year around, resulting in the stable existence of SSH difference between water masses, so the distribution of dynamic fronts is relatively stable (Text S5; Figure S4). Hence, in the detection of the SAF, the thermal front has seasonal limitations and the dynamic front has spatial limitations, which indicates that neither the Southwestern Atlantic thermal front nor the dynamic front can fully accurately reflect the SAF. They only reflect the SAF's thermal and dynamic characteristics, respectively (Text S5). The detection results of SAF by fusion data can fuse the information of the thermal front and the dynamic front and complement the advantages of the two fronts to realize the comprehensive and accurate detection of the SAF. Through the detection results of the fusion data, this study further understands the SAF distribution. North of 50°S, the SAF is mainly distributed along the continental slope break zone between the 200m and 1000m isobaths, and south of 50°S, the SAF is mainly distributed along 1000m isobath, as displayed in Figures 4L–O.

Figures 4A, F, K display the meridional variation of the FP for detection results of three kinds of data. According to the above results, neither the thermal or dynamic front can represent SAF accurately and comprehensively. Therefore, through the meridional variation of the thermal and dynamic fronts, the changes of the thermal and dynamic characteristics of the SAF can be revealed. By comparing the three graphs, we know that the SAF has stable dynamic characteristics, and the main reason for the seasonal variation of the SAF is the change in its thermal characteristics. Between the 40°S and 50°S, SAF is stronger in summer and fall than in winter and spring, mainly because the temperature gradient between the shelf water and the Falkland Current is not obvious due to the spatially uniform surface cooling in winter. However, in the 35°S-40°S region, SAF is weakest in summer. This is mainly because this region is affected by the Brazil Current (a strong warm current) and has a high temperature, while the shelf water temperature is

10.3389/fmars.2023.1140645

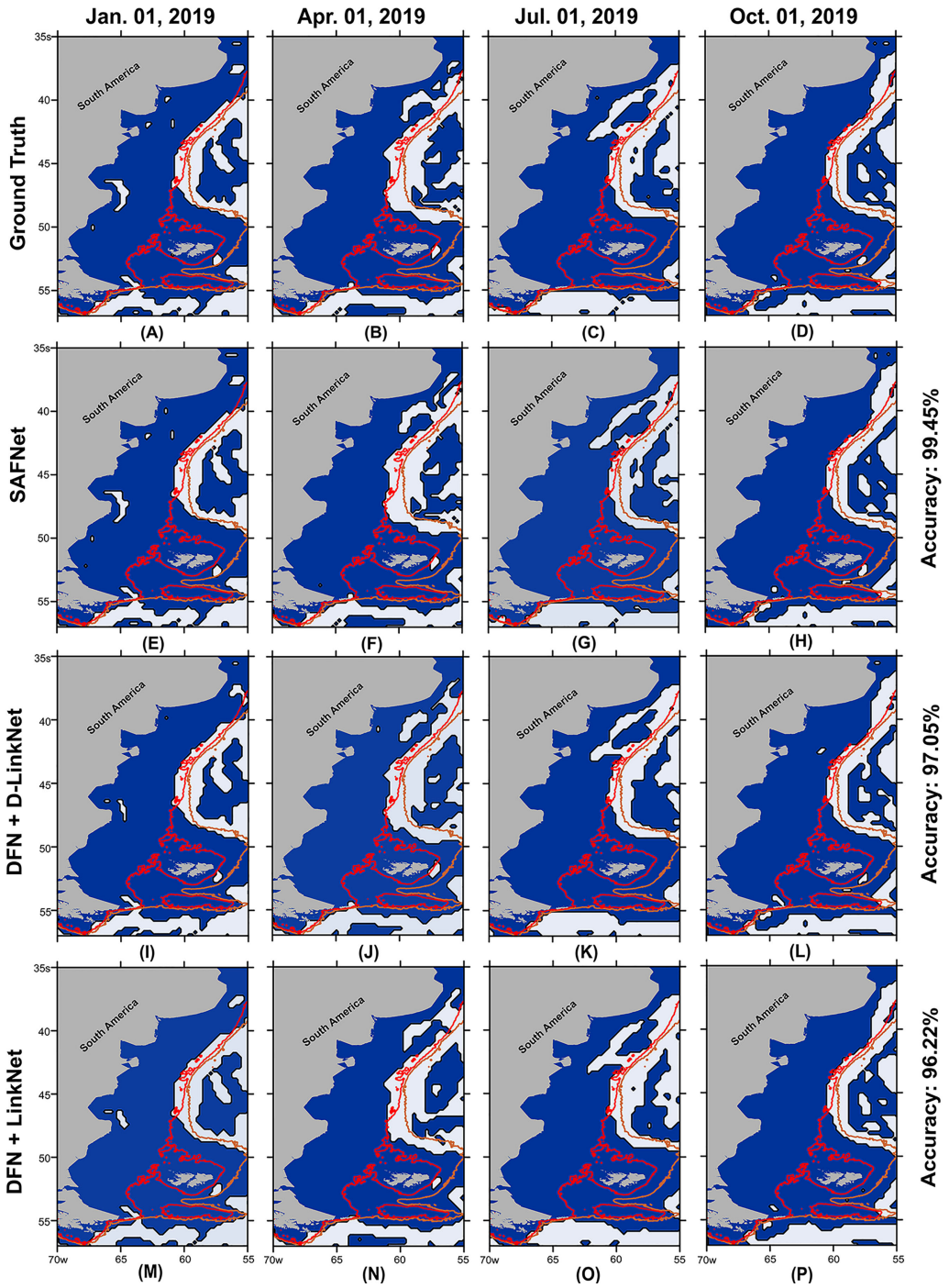


FIGURE 3
The results of the SAF detection. **(A–D)** are SAF ground truth of four days in validation set, the white zone represents the SAF, and the blue zone is the non-frontal zone. **(E–P)** are detection results of each model. The red and brown solid contours are 200m and 1000m isobaths, respectively.

also high in summer, which makes the temperature gradient small and the front intensity weak in this region.

4 Conclusion

SAF is a typical current-induced front with prominent dynamic and thermal characteristics. Therefore, this study proposes the SAFNet that can fuse SST and SSH features over 2010-2019 and detect the SAF from the fusion data accurately, thus achieving an overall high-precision detection of SAF by fusing thermal and dynamic characteristics. The CBAM and DCLs are integrated into the SAFNet. The comparative experiments and ablation experiments show that SAFNet can achieve high precision detection of SAF, and the detection accuracy reaches 99.45%. By comparing the seasonal detection

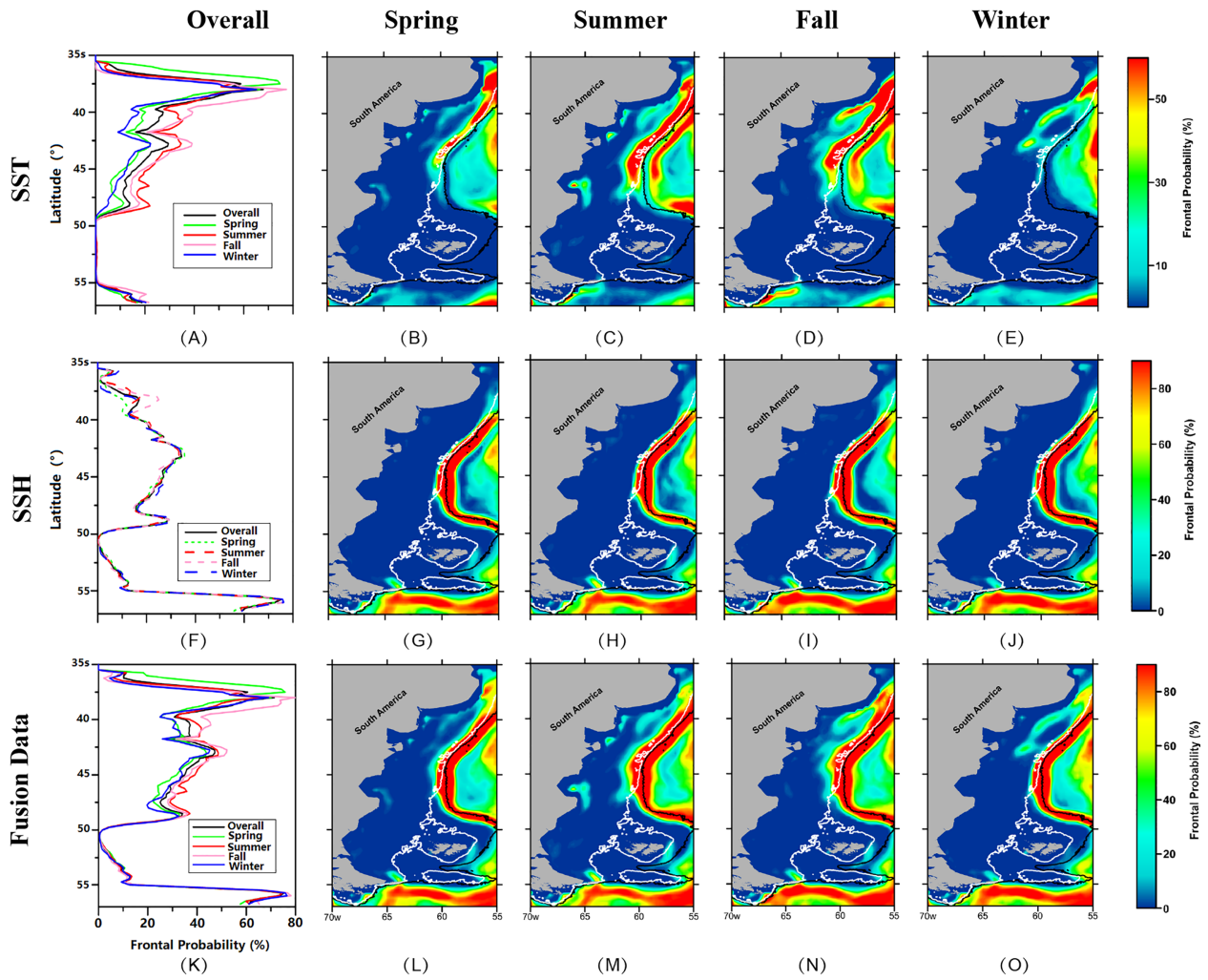


FIGURE 4
Spatiotemporal distributions of the SAF FP obtained by SST, SSH, and the fusion data from 2010 to 2019. The graphs and maps show meridional variation and seasonal spatial distribution of the SAF FP obatined by SST**(A–E)**, SSH **(F–J)** and the fusion data **(K–O)**.

results of SAF FP obtained by SST, SSH, and the fusion of SST-SSH, this study finds that SAF is mainly distributed along the continental slope break zone of South America and the 1000m isobath. According to the meridional variation of FP, we know that the SAF has stable dynamic characteristics, and the reason for the seasonal difference of SAF is the change in its thermal characteristics. The SAF between 40°S and 50°S is weakest in winter due to the uniform surface cooling. In the 35°S-40°S region, the warm Brazil Current makes the water temperature generally higher and the temperature gradient is not obvious in summer, which leads to the weakest SAF.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author.

Author contributions

CM: conceptualization. ZW: designed the experiments. GC, CM and YL: funding acquisition. ZW and YL: methodology. ZW: data processing. ZW: wrote the draft. GC and YL: review and validation. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by the International Research Center of Big Data for Sustainable Development Goals under Grant CBAS2022GSP01, and in part by the National Natural Science Foundation of China under Grant 42276179 and 42030406, and in part by the Pilot National Laboratory For Marine Science and Technology (Qingdao) Laboratory open fund project under Grant 2019B02.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1140645/full#supplementary-material

References

Chambers, D. P. (2018). Using kinetic energy measurements from altimetry to detect shifts in the positions of fronts in the southern ocean. *Ocean Sci.* 14, 105–116. doi: 10.5194/os-14-105-2018

Chapman, C. C., Lea, M. A., Meyer, A., Sallee, J. B., and Hindell, M. (2020). Defining southern ocean fronts and their influence on biological and physical processes in a changing climate. *Nat. Climate Change* 10, 209–219. doi: 10.1038/s41558-020-0705-4

Chaurasia, A., and Culurciello, E. (2017). "LinkNet: Exploiting encoder representations for efficient semantic segmentation", in *IEEE Visual Communications and Image Processing (VCIP)*, St. Petersburg, FL, USA. 2017, 1–4. doi: 10.1109/VCIP.2017.8305148

Dolz, J., Gopinath, K., Yuan, J., Lombaert, H., Desrosiers, C., and Ben Ayed, I. (2019). HyperDense-net: a hyper-densely connected CNN for multi-modal image segmentation. *IEEE Trans. Med. Imaging* 38, 1116–1126. doi: 10.1109/TMI.2018.2878669

Dong, S. F., Sprintall, J., and Gille, S. T. (2006). Location of the antarctic polar front from AMSR-e satellite sea surface temperature measurements. *J. Phys. Oceanogr* 36, 2075–2089. doi: 10.1175/JPO2973.1

Freeman, N. M., Lovenduski, N. S., and Gent, P. R. (2016). Temporal variability in the Antarctic polar front (2002–2014). *J. Geophysical Res.* 121, 7263–7276. doi: 10.1002/2016JC012145

Gao, L., Li, X., Kong, F., Yu, R., Guo, Y., and Ren, Y. (2022). AlgaeNet: a deep-learning framework to detect floating green algae from optical and SAR imagery. *IEEE J. Selected Topics Appl. Earth Observations Remote Sens.* 15, 2782–2796. doi: 10.1109/JSTARS.2022.3162387

Huang, G., Liu, Z., Laurens, V. D. M., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, HI, USA. pp. 4700–4708.

Kittler, J. (1983). On the accuracy of the sobel edge detector. *Image Vision Computing* 1, 37–42. doi: 10.1016/0262-8856(83)90006-9

Legeckis, R. (1979). A survey of worldwide sea surface temperature fronts detected by environmental satellites. *J. Geophysical Research: Oceans* 83, 4501–4522. doi: 10.1029/JC083iC09p04501

Li, Y., Liang, J., Da, H., Chang, L., and Li, H. (2022b). A deep learning method for ocean front extraction in remote sensing imagery. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2021.3081179

Li, X., Zhou, Y., and Wang, F. (2022a). Advanced information mining from ocean remote sensing imagery with deep learning. *J. Remote Sens.* 2022. doi: 10.34133/2022/9849645

Liu, Z., and Hou, Y. (2012). Kuroshio front in the East China Sea from satellite SST and remote sensing data. *IEEE Geosci. Remote Sens. Lett.* 9, 517–520. doi: 10.1109/LGRS.2011.2173289

Liu, Y., Zheng, Q., and Li, X. (2021). Characteristics of global ocean abnormal mesoscale eddies derived from the fusion of Sea surface height and temperature data by deep learning. *Geophysical Res. Lett.* 48. doi: 10.1029/2021GL094772

Lopes, R. M., Marcolin, C. R., and Brandini, F. P. (2016). Influence of oceanic fronts on mesozooplankton abundance and grazing during spring in the south-western Atlantic. *Mar. Freshw. Res.* 67, 626–635. doi: 10.1071/MF14357

Moore, J. K., Abbott, M. R., and Richman, J. G. (1999). Location and dynamics of the Antarctic polar front from satellite sea surface temperature data. *J. Geophysical Research-Oceans* 104, 3059–3073. doi: 10.1029/1998JC900032

Piola, A. R., Franco, B. C., Palma, E. D., and Saraceno, M. (2013). Multiple jets in the malvinas current. *J. Geophysical Research-Oceans* 118, 2107–2117. doi: 10.1002/jgrc.20170

Reynolds, R. W., Smith, T. M., Liu, C., Chelton, D. B., Casey, K. S., and Schlax, M. G. (2007). Daily high-resolution-blended analyses for sea surface temperature. *J. Climate* 20, 5473–5496. doi: 10.1175/2007JCLI1824.1

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: convolutional networks for biomedical image segmentation. *Med. Image Computing Computer-Assisted Intervention Pt Iii* 9351, 234–241. doi: 10.1007/978-3-319-24574-4_28

Takahashi, W., and Kawamura, H. (2005). Detection method of the kuroshio front using the satellite-derived chlorophyll-a images. *Remote Sens. Environ.* 97, 83–91. doi: 10.1016/j.rse.2005.04.019

Wang, Z., Chen, G., Han, Y., Ma, C., and Lv, M. (2021). Southwestern Atlantic ocean fronts detected from satellite-derived SST and chlorophyll. *Remote Sens.* 13. doi: 10.3390/rs13214402

Wang, Y., Yu, Y., Zhang, Y., Zhang, H. R., and Chai, F. (2020). Distribution and variability of sea surface temperature fronts in the south China sea. *Estuar. Coast. Shelf Sci.* 240. doi: 10.1007/978-3-662-61834-9

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). CBAM: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, Munich, Germany, pp. 3–19.

Zhang, X., and Li, X. (2022). Satellite data-driven and knowledge-informed machine learning model for estimating global internal solitary wave speed. *Remote Sens. Environ.* 283. doi: 10.1016/j.rse.2022.113328

Zhou, T., Ruan, S., and Canu, S. (2019). A review: deep learning for medical image segmentation using multi-modality fusion. *Array* 3-4. doi: 10.1016/j.array.2019.100004

Zhou, L., Zhang, C., and Ming, W. (2018). "D-LinkNet: LinkNet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* Salt Lake City, USA. 182–186.

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 21 June 2023
DOI 10.3389/fmars.2023.1162295

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Lina Zhou,
Hong Kong Polytechnic University,
Hong Kong SAR, China
Ning Wang,
Dalian Maritime University, China

*CORRESPONDENCE
Guoqiang Zhong
gqzhong@ouc.edu.cn
Dong Wang
wangdong@ouc.edu.cn

RECEIVED 09 February 2023
ACCEPTED 29 May 2023
PUBLISHED 21 June 2023

CITATION
Wang H, Zhong G, Sun J, Chen Y, Zhao Y, Li S and Wang D (2023) Simultaneous restoration and super-resolution GAN for underwater image enhancement. *Front. Mar. Sci.* 10:1162295. doi: 10.3389/fmars.2023.1162295

COPYRIGHT
© 2023 Wang, Zhong, Sun, Chen, Zhao, Li and Wang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Simultaneous restoration and super-resolution GAN for underwater image enhancement

Huiqiang Wang, Guoqiang Zhong*, Jinxuan Sun, Yang Chen, Yuxiao Zhao, Shu Li and Dong Wang*

College of Computer Science and Technology, Ocean University of China, Qingdao, China

Underwater images are generally of low quality, limiting the performance of subsequent perceptual tasks, such as underwater object detection and recognition. However, only a few methods can improve the quality of underwater images by simultaneously restoring and super-resolving underwater images. In this paper, we propose an end-to-end trainable model based on generative adversarial networks (GANs) called Simultaneous Restoration and Super-Resolution GAN (SRSRGAN) to obtain clear super-resolution underwater images automatically. In particular, our model leverages a cascaded architecture with two stages of carefully designed generative adversarial networks to restore and super-resolve corrupted underwater images in a coarse-to-fine manner. The major advantages of SRSRGAN are twofold. First, it is a unified solution that can simultaneously restore and super-resolve images. Second, SRSRGAN is not limited by the prior experience of the types and levels of underwater degraded images but can perform the inference using only observed corrupted images. These two advantages enable SRSRGAN to enjoy better flexibility and higher practicability in realistic underwater scenarios. Extensive experimental results demonstrate the superiority of SRSRGAN in underwater image restoration, super-resolution, and simultaneous restoration and super-resolution.

KEYWORDS

image enhancement, generative adversarial network, simultaneous restoration and super-resolution, deep learning, underwater images

1 Introduction

With 70% of the earth's surface covered by water, there is great potential for exploiting underwater resources. The underwater environment offers numerous valuable resources, such as marine biology, mineral resources, and tidal energy. However, there is a wide gap between the plentiful marine resources and their exploitation. To this end, various kinds of methods have been proposed to obtain information about the underwater environment to promote the use of marine resources. Among others, a crucial way to obtain information from the underwater environment is image understanding, while the images captured in

realistic underwater scenarios usually have severe defects, such as blurriness, noise, and color distortion (Soni and Kumare, 2020).

Specifically, underwater image defects are caused by various factors. Light rays exponentially decay as the underwater depth increases, which makes underwater images of low contrast and darkness (Ancuti et al., 2018). Furthermore, lights of different colors have different absorption rates underwater, depending on the wavelengths, which results in color distortion of underwater images (Chiang and Chen, 2012). In addition, bubbles and suspended particles in water may cause noise in underwater images (Lu et al., 2017b). Hence, the poor visibility of underwater images seriously affects the exploration of the underwater environment. On the other hand, high-resolution images are essential in many realistic underwater applications, such as marine animal recognition (Chen et al., 2021; Wang et al., 2023b), seabed detection, and deep ocean resources exploration (Lu et al., 2017a). Therefore, the critical tasks for underwater image enhancement are eliminating defects and obtaining super-resolution (SR) images.

To the best of our knowledge, only a few approaches can simultaneously restore and super-resolve underwater images. In particular, Cheng et al. (2018) propose a method that restores underwater images by the white balance (Liu et al., 1995) with the contrast limited adaptive histogram equalization (CLAHE) (Reza, 2004) and super-resolves the restored image by a super-resolution generative adversarial network (GAN) (Ledig et al., 2017). However, due to the fact that this method only utilizes traditional color correction as a preprocessing step for the input image of the super-resolution model during the restoration stage, it limits its ability to remove other types of degradation features. Recently, Islam et al. (2020a) also introduce an approach to learning enhanced super-resolution underwater images. However, their proposed method is limited in its ability to model complex degradation features due to its lack of consideration for capturing multi-scale features in the network architecture design. Due to these limitations, these methods are difficult to generate high-fidelity and high-quality super-resolution images in underwater image enhancement in real-world scenarios.

To address the above issues, we propose simultaneous restoration and super-resolution GAN (SRSRGAN) to obtain underwater images of high visual quality, which is an end-to-end trainable model based on GAN. With a two-stage design, SRSRGAN captures underwater degradation information and fine-grained high-frequency information in the restoration stage and the super-resolution stage, respectively. In the restoration stage, benefiting from the superior structure of the proposed multi-level degradation restoration generator (MLDRG), our model leverages degradation information among different scales, positions, and channels to transform degraded images to clean images. In the super-resolution stage, the high frequency learning module (HFLM) excavates fine-grained high-frequency information to super-resolve clean images. In addition, we adopt a relativistic discriminator to further enhance the quality of our generated underwater images. Thanks to the corporation of the restoration stage and the super-resolution stage, SRSRGAN enjoys two highly expected merits, i.e., i) it provides a unified solution for simultaneous underwater image restoration and super-resolution reconstruction; ii) it is free from the prior of the underwater corruption types and ratios. Extensive experimental results show that SRSRGAN is superior to the state-of-the-art (SOTA) methods

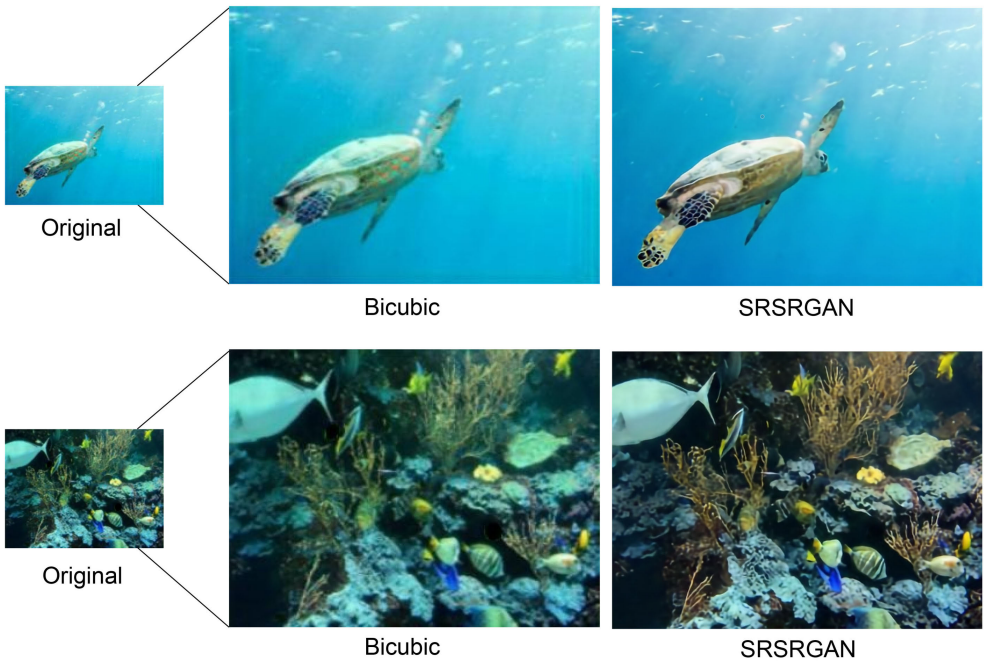


FIGURE 1
The proposed SRSRGAN model provides realistic underwater image enhancement results through an effective inference. The first and second rows, from left to right, are the original underwater image, the bicubic interpolation enhanced image, and the image enhanced by our method.

in underwater image restoration/enhancement, single image super-resolution (SISR), and simultaneous restoration and super-resolution. The qualitative enhancement effect of SRSRGAN is shown in Figure 1.

In summary, our main contributions are as follows:

- We propose a new underwater image enhancement model for simultaneous restoration and super-resolution of underwater images, called SRSRGAN, which does not require any prior image degradation information to perform inference in advance. Therefore, it can flexibly deal with complex underwater scenarios.
- We design a two-stage framework to process underwater images. In the restoration stage, the proposed MLDRG leverages degradation information from different scales, positions, and channels to transform degraded underwater images to clean images. Moreover, the super-resolution stage is presented to enhance the representational ability of high-frequency features for underwater image super-resolution. In addition, this staged design improves the flexibility of the network model.
- Qualitative and quantitative comparisons among SRSRGAN, underwater image restoration/enhancement methods, SISR methods, and existing simultaneous restoration and super-resolution methods show the superiority of SRSRGAN.

2 Related work

In the existing work, few methods are available for simultaneous restoration and super-resolution of underwater images, except the ones mentioned in Section 1 (Ledig et al., 2017; Cheng et al., 2018). Therefore, in this section, we mainly review the research progress of underwater image restoration/enhancement and SISR.

2.1 Underwater image restoration/enhancement

Traditional underwater image restoration/enhancement algorithms aim to recover a clean image from the degraded observation, including automatic white balance (Liu et al., 1995), histogram equalization (Hummel, 1977), and CLAHE (Reza, 2004). Although these methods improve the quality of underwater images to a certain extent, there are still various problems, such as color deviation, artificial artifacts, and noise amplification. Inspired by the morphology and function of the teleost fish retina, Gao et al. (2019) propose an underwater image enhancement model to solve the problems of blurring and nonuniform color biasing in underwater images. Moreover, several methods are proposed inspired by the dark channel prior (He et al., 2011). Particularly, Drews et al. (2013) consider underwater images' blue and green channels as underwater visual information sources, and apply a dark channel method to process underwater visual information. Galdran et al. (2015) propose a dark channel variant called the red channel method to restore the lost contrast and colors associated to short wavelengths in underwater images. Recently, Li et al. (2022) propose a framework called ACCE-D that uses multiple filters and adaptive color and contrast enhancement strategies to enhance underwater images. In addition, Alenezi et al. (2022) propose a method to enhance underwater images by estimating global background light and transmission maps. However, these methods have a common limitation in that the prior assumptions may be invalid with the changes in environmental status.

As convolutional neural networks develop rapidly, some deep networks are used to establish mapping relationships from an underwater image to the clear one (Hou et al., 2018; Lu et al., 2018). In particular, Li et al. (2020) give an overview of the previous work for underwater image restoration and establish a CNN model named Water-Net to get restored underwater images. Additionally, the emergence of GAN (Goodfellow et al., 2014) provides more chances for underwater image restoration. For example, Li et al. (2018) propose WaterGAN to generate underwater images from in-air images, which uses two fully convolutional networks to estimate the depth of the generated underwater images and correct their color, respectively. Different from it, UGAN uses two GAN-based models for underwater image generation and color correction, respectively (Fabbri et al., 2018). In recent work, Underwater GAN (Yu et al., 2019) uses Wasserstein GAN-GP (Gulrajani et al., 2017) as the network's backbone for underwater image restoration. Additionally, Guo et al. (2020) propose a multi-scale dense GAN (UWGAN) for underwater image enhancement. These methods improve the quality of underwater images to a certain extent. However, they only focus on restoring the color contrast and color distortion of underwater images and do not further improve the image quality by improving the image resolution. Liu et al. (2022) propose a twin adversarial contrastive learning method to enhance the visual quality of underwater images. Many previous underwater image enhancement methods have only focused on restoring the color contrast and color distortion of underwater images. However, their method has limited ability to remove noise in underwater images. Therefore, Wang et al. (2023a) propose an end-to-end underwater attention generative adversarial network to alleviate the influence of underwater noise problem. These methods improve the quality of underwater images to some extent.

2.2 Single image super-resolution

In some early survey papers on image super-resolution (Nasrollahi and Moeslund, 2014; Köhler et al., 2017; Yang et al., 2019), there are two principal categories of image super-resolution: multiple image super-resolution (MISR) (Tsai, 1984; Capel and Zisserman, 2001; Caner et al., 2003; Farsiu et al., 2004; Harmeling et al., 2010) and SISR (Storkey, 2002; Lian, 2006; Yang et al., 2008; Yang et al., 2010; Dong et al., 2016). Here, we mainly introduce SISR, as the number of underwater images is still very small in general.

Interpolation-based SR methods are typically used to increase the resolution of an image, such as bicubic interpolation (Keys,

1982) and Lanczos filtering (Duchon, 1979). The reconstructed edges are generally blurred in the super-resolution images obtained by these methods. These methods obtain the SR image, while the reconstructed edges are generally blurry. Subsequent methods focus on matching the edges of the low-resolution (LR) and high-resolution (HR) images (Li and Orchard, 2000; Muresan, 2005). However, the HR images they generate still suffer from blurring and artifacts.

Sparse representation SR methods are based on the sparse signal representation and compressed sensing theory. Yang et al. (Yang et al., 2008; Yang et al., 2010) train two dictionaries for LR and HR patches jointly. They consider the sparse representations of LR and HR images and utilize the spare representations of the LR images to obtain the HR images. Moreover, the natural image prior framework is added to guide the sparse representation SR method (Kim and Kwon, 2010). However, such an SR method based on the spare representation needs a long time to train the sparse coding dictionary. More recently, Timofte et al. (Timofte et al., 2013; Timofte et al., 2014) improve the efficiency of sparse representation SR methods using anchored neighborhood regression on the LR patch in the dictionary. Nevertheless, the texture details are generally absent from the generated SR images.

With the rapid development of deep learning, many SR methods based on deep learning have emerged and achieved excellent performance in recent years. Dong et al. (Dong et al., 2014; Dong et al., 2016) propose a fully convolutional network to establish a mapping between the LR and HR images, which has great superiority over the previous approaches. Later, Shi et al. (2016) propose a sub-pixel convolutional neural network, which expands the channels of output features by the convolutional layers and then rearranges the tensor to obtain the HR images. With the depth of neural networks increasing, Kim et al. (2016) use a deep neural network similar to VGG-net to generate SR images. In addition, some researchers propose to utilize the residual networks to achieve an excellent SR effect (Lim et al., 2017; Zhang et al., 2018b; Zhang et al., 2018c; Chen et al., 2019). Furthermore, with the flourishing of GAN-based models, recent work has shown great success in SISR. Ledig et al. (2017) propose a super-resolution generative adversarial network (SRGAN) to recover SR images from LR images. Wang et al. (2018) enhance SRGAN by modifying the generator with residual in-residual dense blocks, which can generate realistic images with natural textures. Unfortunately, these methods are only suitable for images taken in the air but cannot perform well on underwater images.

At present, a few researchers are involved in the field of underwater image super-resolution. Particularly, Lu et al. (2017a) propose a two-step method for underwater image super-resolution. Firstly, they obtain a scattered HR image and a descattered HR image by self-similarity SR methods; secondly, they fuse the two HR images to obtain the final image. More recently, Islam et al. (2020b) propose a fully convolutional neural network using residual learning for underwater SISR, called super-resolution using deep residual multipliers (SRDRM). In addition, they also formulate an adversarial training pipeline (SRDRM-GAN). However, the generated images by these methods have limited image quality and visual perception. In this paper, we propose an end-to-end trainable GAN-based model called SRSRGAN, which can simultaneously restore and super-resolve underwater images.

3 The proposed model

SRSRGAN aims to build an effective simultaneous restoration and super-resolution model for underwater image enhancement, which obtains an input underwater image I^U and outputs a clear super-resolution image I^{SR}.

In this section, we elaborate on the proposed end-to-end trainable model SRSRGAN, which consists of a restoration stage (Rstage, $f_R(\cdot)$) and a super-resolution stage (SRstage, $f_{SR}(\cdot)$), as shown in Figure 2.

In brief, SRSRGAN first feeds it into $f_R(\cdot)$ to learn the clear image I^R for a given degraded image I^U. Then, I^R is further passed through $f_{SR}(\cdot)$ to obtain the super-resolution image I^{SR}. In the following part, we first illustrate the restoration stage in Section 3.1.

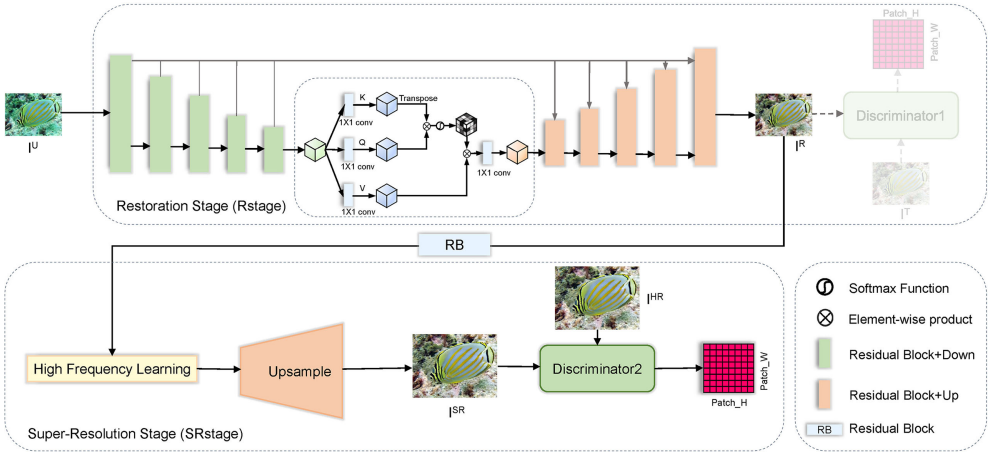


FIGURE 2
The overall network architecture of SRSRGAN.

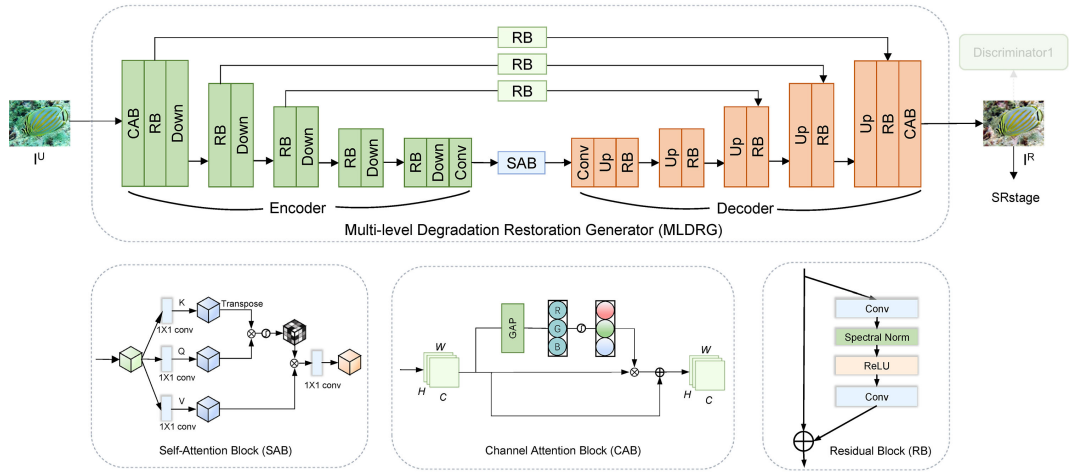


FIGURE 3
The network architecture of the restoration stage.

Then, we describe the super-resolution stage in Section 3.2. Finally, we introduce our end-to-end trainable framework of SRSRGAN in Section 3.3.

3.1 Restoration stage

The light passing through the water attenuates as the depth increases, and the background light will also affect the underwater images. In order to restore a clear image I^R from an underwater image I^U containing noise and distortion, we propose a GAN-based model in the restoration stage. Mathematically,

$$\min_{G_R}\max_{D_R} V(G_R, D_R) = E_{I^T\sim p_{data}(I^T)}[\log\ D_R(I^T)] + E_{I^U\sim p_G(I^U)}[\log\ (1 - D_R(G_R(I^U)))], \quad (1)$$

where I^U and I^T denote the image to be restored and the ground-truth image, respectively.

3.1.1 Network architecture

The overall network architecture of our proposed restoration model is shown in Figure 3, which is designed to restore the clear image I^R from the noisy and distorted underwater image I^U. To make SRSRGAN better deal with underwater degradation, we carefully design a multi-level degradation restoration generator (MLDRG) consisting of an encoder and a decoder.

3.1.1.1 Multi-level degradation restoration generator

Our MLDRG focuses on dealing with noise and color degradation because color degradation and noise degradation are frequent in underwater images.

Specifically, color distortion is an extremely important issue when processing underwater images. Due to the unique properties of the underwater environment, color distortion in underwater images is even more severe, which has an adverse impact on the quality and usability of the images. To address this issue, we utilize the channel attention blocks (CAB) (Hu et al., 2018) to enhance the network's focus on color information, thereby improving the network's color restoration capability. To better handle the color restoration of shallow feature colors, we place the CAB in the first and last part of the decoder. The role of the CAB is to enhance the network's focus on shallow features, thereby improving the network's color restoration capability. In the first part of the decoder, the CAB can assist the network in better capturing the color information of shallow features, providing a better foundation

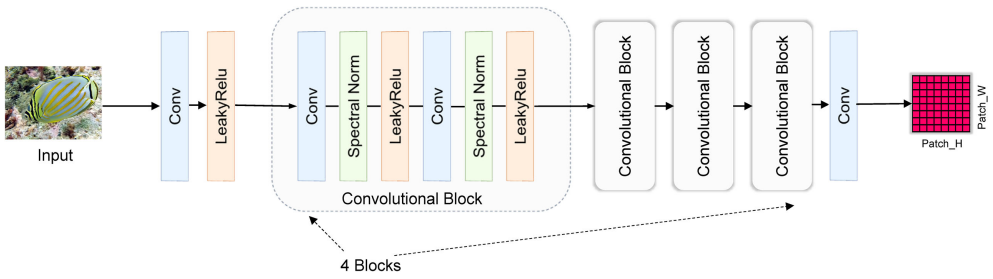


FIGURE 4
The patch discriminator of the restoration stage.

for subsequent color restoration. In the last part of the decoder, the CAB can further enhance the network's focus on shallow features, thereby improving the network's color restoration capability.

On the other hand, due to the random, irregular, and uneven distribution of noise in images, the presence of noise can increase the difficulty of model restoration and negatively impact image quality. Therefore, we implement a self-attention block (SAB) (Vaswani et al., 2017) in front of the decoder to remove the noise with different characteristics in underwater images. Specifically, SAB can determine the weight of each pixel by calculating its similarity with other pixels, allowing the network to focus more on the contextual information related to the current pixel. This can help the network better understand the structural information in the image and remove the random and irregular noise. Hence, the generator has the capacity to better handle the noise of underwater images than that without the attention block. For implementation details, the global feature maps are coded into queries, keys, and values in d_q, d_k, and d_v dimensions, respectively. The attention function is defined as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{2}$$

where Q is a matrix for queries, K and V are matrices for keys and values, respectively.

In addition, we implement the spectral normalization (SN) (Miyato et al., 2018) after each convolutional layer in MLDRG. SN layers control the Lipschitz constant of MLDRG by constraining the spectral norm ($\sigma(W) = 1$) of each layer. In particular, the Lipschitz constant describes the intensity of the output as it changes with the input. For the Lipschitz continuous function ϕ, if it satisfies

$$\frac{||\phi(x^{'}) - \phi(x)||_2^2}{||x^{'} - x||_2^2} \leq k, \tag{3}$$

then k ($k \geq 0$) is called the Lipschitz constant. In other words, MLDRG is insensitive to the perturbation of the inputs. Hence, it can better handle noisy underwater images than that without the SN layer. This discovery has the same viewpoint as (Lin et al., 2019) that the Lipschitz continuity is effective for image-denoising tasks.

Ideally, the generator should be able to retain more multi-scale information and spatial context information while providing flexibility for the super-resolution stage. To this end, we employ the residual block with the SN layer to extract features at 5 scales. In order to switch scales in our restoration framework, we use a 3×3 convolutional layer with stride 2 for downsampling and a bilinear interpolation algorithm for upsampling after a 3×3 convolutional layer with stride 1. Considering the intrinsic information loss in downsampling and upsampling, we add a residual block at each sale to fuse useful information from the encoder to the decoder.

3.1.1.2 Restoration discriminator

Figure 4 shows the discriminator of the restoration stage. The discriminator D_R distinguishes the ground-truth image I^T from the generated image I^R at the level of image patches. Specifically, given an input image I_d to be discriminated, it first extracts the shallow features F_0 of the discriminated image by a 3×3 convolutional layer. Mathematically,

$$F_0 = Conv(I_d). \tag{4}$$

Since the extraction of deep features is essential to discriminate the images, we design 4 convolutional blocks, each containing a 3×3 convolutional layer with stride 1 and a 3×3 convolutional layer with stride 2. In addition, considering the Lipschitz continuity of the discriminator, we add an SN layer and a Leaky ReLu function after each convolutional layer to stabilize its network training. Therefore, we feed the previously extracted shallow feature F_0 into 4 designed convolutional blocks to further excavate the deep features of the input image. Mathematically,

$$F_i = H_{Block_i}(F_{i-1}), \quad i = 1, 2, \ldots, N, \tag{5}$$

where H_{Block_i} denotes the i-th convolutional block in the path discriminator, and N denotes the number of convolutional blocks. Hence, F_N is the final output of the patch discriminator, and $F_1 \sim F_{N-1}$ are intermediate feature maps extracted from our convolutional blocks.

Finally, we utilize a 3×3 convolutional layer to predict a 16×16 probability matrix P_{output} for image patch discrimination. Through the probability matrix, we can increase the discriminator's sensitivity to image patch detail discrimination, thus forcing MLDRG to generate more realistic details. Mathematically,

$$P_{output} = Conv(F_N). \tag{6}$$

Moreover, the discriminator loss in the restoration stage can be defined as:

$$L_D^R = -E_{I^T \sim p_{data}(I^T)}[\log\ D_R(I^T)] + E_{I^U \sim p_G(I^U)}[\log\ (D_R(G_R(I^U)))]. \tag{7}$$

3.1.2 Loss function

To remove the corruption from the observed underwater images, we formulate some objective functions. First, the adversarial loss for MLDRG can be formulated as:

$$L_{Adv}^R = -E_{I^U \sim p_G(I^U)}[\log\ (D_R(G_R(I^U)))]. \tag{8}$$

In the restoration stage, we define the mean absolute error (MAE) loss to measure the pixel gap between the generated images and the target images. Mathematically,

$$L_{MAE}^R = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}\left|I_{i,j}^T - G_R(I^U)_{i,j}\right|. \tag{9}$$

Moreover, in order to enhance the human visual quality of reconstructed images, we formulate a perceptual loss to measure the distance between the restored images and the ground-truth images on the perceptual feature space. It can be formulated as:

$$L_{Perceptual}^R = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}\left|\phi_{x,y}(I^T)_{i,j} - \phi_{x,y}(G_R(I^U))_{i,j}\right|, \tag{10}$$

where $\phi_{x,y}$ symbol denotes a pre-trained VGG-net feature extractor, which can obtain the feature map of the y-th convolutional layer before the x-th max-pooling layer (Simonyan and Zisserman, 2015), while W and H symbols denote the width and height of the obtained corresponding feature map, respectively.

3.2 Super-resolution stage

SISR is aimed at generating the high-resolution image I^{SR} from the low-resolution image I^{LR}. Generally, the degradation process from I^{HR} to I^{LR} is unknown and can be affected by various factors, such as defocusing and noise. Following the common practice (Zhang et al., 2017; Zhang et al., 2018a; Liang et al., 2022), we obtain I^{LR} by a downsampling operation with the scaling factor r. For an image with C channels, the I^{LR} and I^{SR} are described as a $C \times W \times H$ tensor and a $C \times rW \times rH$ tensor, respectively.

To add the texture details to the restored image I^{R} fed from the restoration stage, we propose another GAN-based model, which is aimed at generating the corresponding super-resolution image I^{SR} from the restored image I^{R}. The objective function of the super-resolution stage is formulated as:

$$\min_{G_{SR}} \max_{D_{SR}} V(G_{SR}, D_{SR}) = E_{I^{HR} \sim p_{data}(I^{HR})}[\log\ D_{SR}(I^{HR})] + E_{I^{LR} \sim p_G(I^{LR})}[\log\ (1 - D_{SR}(G_{SR}(I^{LR})))], \tag{11}$$

where I^{LR} and I^{HR} symbols denote the corresponding high-resolution and low-resolution images, respectively.

3.2.1 Network architecture

The generator of the super-resolution stage is illustrated in Figure 5, which consists of a high frequency learning module (HFLM) and an upsampling module (UM). Due to the effective collaboration of HFLM and UM, our super-resolution stage can restore many high-frequency details from low-resolution underwater images.

3.2.1.1 High frequency learning module

After obtaining the restored images after the restoration stage, we further seek to excavate the high-frequency information of underwater images. High-frequency information can be described as Figure 6, where I^{LR} is obtained by bicubic interpolation of the restored image, and I^{HR} is the high-resolution ground truth image. Our task in the super-resolution stage is to learn these high-frequency information. The task of our high-frequency feature learning module is to learn these high-frequency information. To this end, HFLM first directly transmits the low frequency information of the low resolution image to the upsampling module through a connection, and then learns the high frequency information of the image through 16 residual-in-residual blocks (Wang et al., 2018). In each dense block, it captures and transmits high frequency information by establishing dense residual connections in the network. Specifically, each layer is connected to all previous layers, making it easier for high frequency features to propagate throughout the network and be better captured and represented. Mathematically,

$$F_i^H = \begin{cases} \rho \cdot H_{D_i}(I_{LR}) + I_{LR}, & i = 1; \\ \rho \cdot H_{D_i}(F_{i-1}) + F_{i-1}^H, & i = 2, \ldots, N, \end{cases} \tag{12}$$

where I_{LR} and ρ denote the output of the restoration stage and the residual scaling parameter, respectively, H_{D_i} represents the i-th dense block, N represents the number of the dense blocks, and F_i^H represents the i-th intermediate high-frequency feature. Specifically, we set ρ and N to 0.2 and 16, respectively. Furthermore, we add an SN layer after each convolutional layer to constrain the Lipschitz continuity of HFLM, and the leak rate of the Leaky Relu activation function is set to 0.2.

3.2.1.2 Upsampling module

After obtaining the high-frequency information provided F_N^H by the HFLM, we adopt a pixshuffle layer for upsampling, passing the F_N^H through convolutional layers and inter-channel recombination to obtain a high-resolution feature map I^{SR}. Similar to HFLM, we

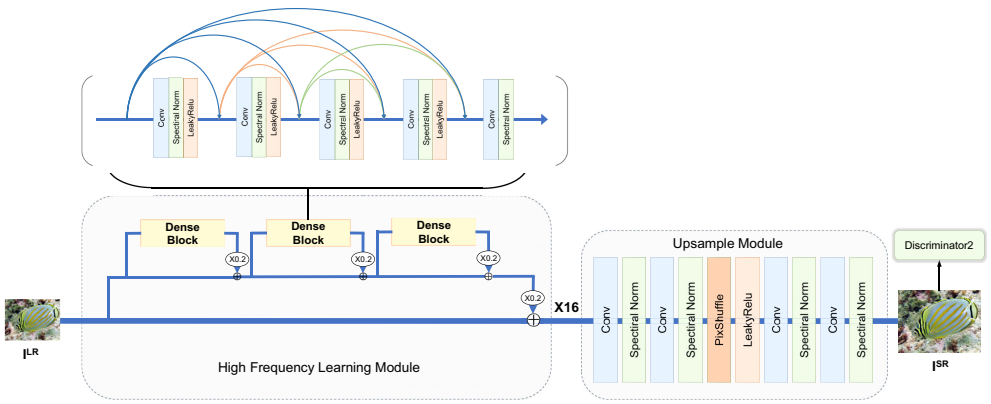


FIGURE 5
The generator of the super-resolution stage.



FIGURE 6
The representation of high frequency information.

add an SN layer after each convolutional layer to constrain the Lipschitz continuity of UM and stabilize the network training. Mathematically,

$$I^{SR} = UM_{\uparrow}(F_N^H + I^{LR}). \tag{13}$$

3.2.1.3 Super-resolution discriminator

The architecture of the super-resolution stage's discriminator is similar to that of the restoration stage. However, for image super-resolution, we expect the output by its discriminator to be the probability that the real image I^{HR} is relatively more realistic than the fake image I^{SR}. To this end, we use a relativistic discriminator (Jolicoeur-Martineau, 2018), which is defined as:

$$D_{SR}(I^{HR}, I^{SR}) = \sigma(P_{SR}(I^{HR}) - E[P_{SR}(I^{SR})]), \tag{14}$$

$$D_{SR}(I^{SR}, I^{HR}) = \sigma(P_{SR}(I^{SR}) - E[P_{SR}(I^{HR})]), \tag{15}$$

where I^{SR} denotes the output of the generator in the super-resolution stage, $P_{SR}(I)$ denotes the probability output of the patch discriminator, $E[\cdot]$ represents the operation of taking the average probability output obtained from mini-batch images, and σ denotes the sigmoid activation function. Then, we formulate the discriminator loss as follows:

$$L_D^{SR} = -E_{I^{HR}\sim p_{data}(I^{HR})}[\log\ D_{SR}(I^{HR}, G_{SR}(I^{LR}))] +$$

$$- E_{I^{LR}\sim p_G(I^{LR})}[\log\ (1 - D_{SR}(G_{SR}(I^{LR}), I^{HR}))]. \tag{16}$$

3.2.2 Loss function

Similar to the restoration stage, The MAE loss and perception loss are both used to optimize its generator in the super-resolution stage for a better reconstruction effect. Mathematically,

$$L_{MAE}^{SR} = \frac{1}{r^2 WH}\sum_{i=1}^{rW}\sum_{j=1}^{rH}\left|I_{i,j}^{HR} - G_{SR}(I^{LR})_{i,j}\right|, \tag{17}$$

$$L_{Perceptual}^{SR} = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}\left|\phi_{x,y}(I^{HR})_{i,j} - \phi_{x,y}(G_{SR}(I^{LR}))_{i,j}\right|. \tag{18}$$

In addition, we formulate the adversarial loss for the generator as follows:

$$L_{Adv}^{SR} = -E_{I^{HR}\sim p_{data}(I^{HR})}[\log\ (1 - D_{SR}(I^{HR}, G_{SR}(I^{LR})))] +$$

$$- E_{I^{LR}\sim p_G(I^{LR})}[\log\ D_{SR}(G_{SR}(I^{LR}), I^{HR})]. \tag{19}$$

It can be clearly seen that the adversarial loss in our super-resolution stage includes both I^{HR} and $I^{SR} = G_{SR}(I^{LR})$. Hence, the gradient of the generator in the super-resolution stage benefits from both the generated images and the ground-truth images. In contrast, the gradient of the generator in the previous stage only benefits from the generated images.

3.3 SRSRGAN

SRSRGAN combines the restoration stage and the super-resolution stage into an end-to-end trainable model. Concretely, the generator of SRSRGAN combines generators of the restoration stage and the super-resolution stage. For training, we adopt a two-stage training strategy. In the first stage, we use a restoration discriminator to supervise the restoration stage generator's training, which serves as a pre-training for the restoration stage generating adversarial network. In the second stage, we directly use the super-resolution stage discriminator to supervise the entire SRSRGAN model's training. Finally, we can train SRSRGAN as an end-to-end GAN-based model.

During inference, by feeding degraded underwater images into the SRSRGAN model, we can obtain clean high-resolution underwater images end-to-end.

In addition, by doing so, SRSRGAN has the following advantages in addition to the benefits brought by its well-designed model structure:

- Removing noise from images will generally introduce artifacts to the images, while the devised super resolution stage in SRSRGAN can generate texture details to avoid the artifacts.
- The generator of SRSRGAN benefits from both the image restoration and image super-resolution tasks, leading to better performance than using the restoration stage and super-resolution stage sequentially.
- During inference, degraded images only require one forward pass through the network to complete both image restoration and super-resolution reconstruction.

4 Experiments

We applied our SRSRGAN to underwater image restoration/enhancement, SISR, and simultaneous restoration and super-resolution. We also made a comparison with the state-of-the-art (SOTA) methods for underwater images.

We took the underwater image I^U as input in the restoration stage and the ground-truth image I^T for model training. I^R was the restored image generated in the restoration stage. In the super-resolution stage, we downsampled I^R (I^{HR}) with a scaling factor r = 2 to get I^{LR}. I^{SR} was the super-resolved image generated in the super-resolution stage.

We conducted experiments in PyTorch on NVIDIA GeForce RTX 3090 GPUs. To optimize SRSRGAN, we employed the Adam (Kingma and Ba, 2014) optimizer to perform global iterative learning with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and its learning rate was set to 2×10^{-4}. Considering the model depth, we adopted the warming-up strategy (He et al., 2016) to improve the learning rate gradually.

4.1 Data and metrics

4.1.1 Dataset

We used 790 images from the UIEBD (Li et al., 2020) dataset and 1500 images from the UFO-120 (Islam et al., 2020a) dataset to train SRSRGAN and the compared methods, except Gao et al. (2019) and SESR (Islam et al., 2020a). For Gao et al. (2019)'s method, we downloaded the results from the author's GitHub webpage; for SESR, we downloaded the released well-trained model from the author's GitHub webpage. In addition, we employed various datasets to test them, including the other 100 images in the UIEBD dataset with the corresponding reference images, 120 images in the UFO-120 dataset with the corresponding reference images, the same underwater scene shot by seven different professional cameras (Ancuti et al., 2018), 248 images in the USR-248 (Islam et al., 2020b) dataset, 25 images previously used for the evaluation in related papers (Emberton et al., 2015; Galdran et al., 2015; Ancuti et al., 2018; Guo et al., 2020), and 19 real underwater images we collected from the Internet.

4.1.2 Full-reference metrics

We performed a full-reference evaluation of underwater images with widely used metrics, i.e., peak signal to noise ratio (PSNR) and structural similarity index (SSIM) (Wang et al., 2004). We treated the clear HR image as the ground-truth image. The higher the PSNR value is, the closer the enhanced image is to the ground-truth image in terms of image content. Similarly, the higher the SSIM value is, the closer the enhanced image is to the ground-truth image in terms of image texture and structure.

4.1.3 Non-reference metrics

We adopted two commonly used non-reference metrics for underwater image quality evaluation, i.e., UCIQE (Yang and Sowmya, 2015) and UIQM (Panetta et al., 2016). A higher UCIQE score indicates that the enhanced image has less color cast, less blur, and better contrast. Meanwhile, a higher UIQM score indicates that the enhanced image is more in line with human perception.

4.2 Evaluation on underwater image restoration

We first qualitatively compared SRSRGAN with several SOTA underwater image restoration/enhancement methods. As Figure 7 illustrates, FUnIE-GAN (Islam et al., 2020c), CycleGAN (Zhu et al., 2017), and Gao et al. (2019)'s method have limited positive effects on the greenish water image, while FUnIE-GAN (Islam et al., 2020c) has a less positive effect on the bluish water image. Pix2Pix (Isola et al., 2017) has an obvious reddish color shift. UGAN (Fabbri et al., 2018) and Gao et al. (2019)'s method aggravate the noise effect that introduces light spots in the first image. In addition, TACL (Liu et al., 2022)'s ability to correct the green and blue tones of underwater images is limited. In contrast, SRSRGAN can rectify the greenish and bluish hue of the images, and eliminate the blurring and noise on the images.

In addition, Figure 8 shows that images contain the standard *Macbeth Color Checker* taken by seven different professional cameras, i.e., Panasonic TS1, Pentax W80, Olympus Tough 8000, Pentax W60, Olympus Tough 6000, FujiFilm Z33, and Canon D10. The images processed by Gao et al. (2019)'s method still suffer from obvious color distortion. FUnIE-GAN (Islam et al., 2020c) deals well with the bluish color deviation but produces a reddish color shift when handling the dark image. On the contrary, SRSRGAN obtains the best color correction for different cameras.

The performance of SRSRGAN and its comparison methods is quantitatively evaluated in terms of the full-reference and non-reference metrics, as shown in Table 1. For the full-reference evaluation, the results are obtained by comparing the results of each method with the corresponding ground truth (reference) images. It can be seen that SRSRGAN achieves the highest PSNR and SSIM value, which means that the images generated by SRSRGAN have the closest content and structure to the ground-truth images. Moreover, SRSRGAN obtains the highest UCIQE and

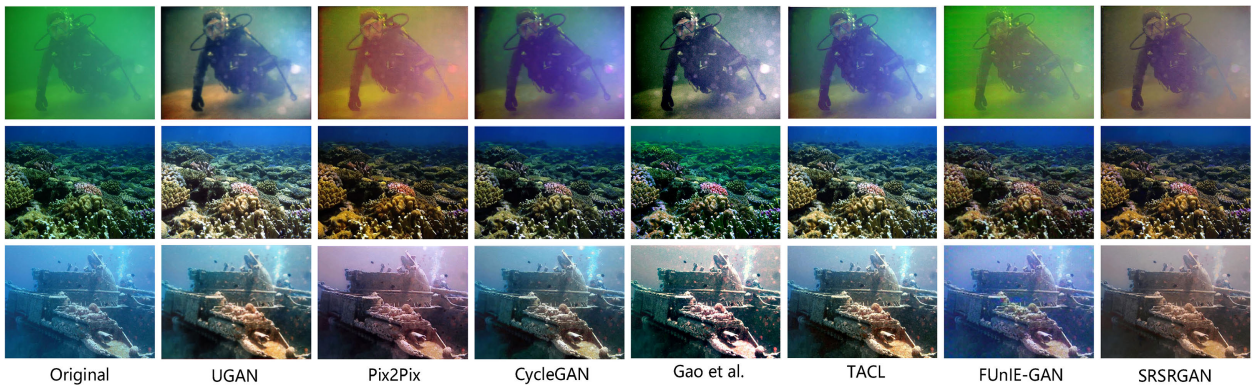


FIGURE 7
Qualitative comparison between SRSRGAN and the SOTA restoration/enhancement methods. From left to right are the original underwater images, the results of UGAN (Fabbri et al., 2018), Pix2Pix (Isola et al., 2017), CycleGAN (Zhu et al., 2017), Gao et al. (2019)'s method, TACL (Liu et al., 2022), FUnIE-GAN (Islam et al., 2020c), and SRSRGAN. Our sample image is sourced from the public dataset UIEBD (Li et al., 2020).

UIQM score, which indicates that the images generated by SRSRGAN have the best color and human visual perception.

4.3 Evaluation on underwater image super-resolution

Following the same procedure, we evaluated the qualitative and quantitative SR performance of SRSRGAN, respectively. In Particular, we took the existing underwater SISR methods for comparison, i.e., SRDRM (Islam et al., 2020b) and SRDRM-GAN (Islam et al., 2020b). In addition, we compared SRSRGAN with some SOTA SISR methods (for images taken in the air), including VDSR (Kim et al., 2016), EDSR (Lim et al., 2017), DBPN (Haris et al., 2018), SRCNN (Dong et al., 2016), SRGAN (Ledig et al., 2017), and ESRGAN (Wang et al., 2018). From Figure 9, we can find that ESRGAN achieves the best results among the SISR methods for images taken in the air, while SRDRM and SRDRM-GAN can better handle underwater images than ESRGAN. In contrast, SRSRGAN generates clear super-resolution images with the correct color and sharp texture details.

Table 2 illustrates the quantitative evaluation on SRSRGAN and the compared methods. It is obvious that SRSRGAN obtains the highest score for both PSNR and SSIM, which indicates that the images generated by SRSRGAN have the highest pixel similarity and structure consistent with the ground-truth images.

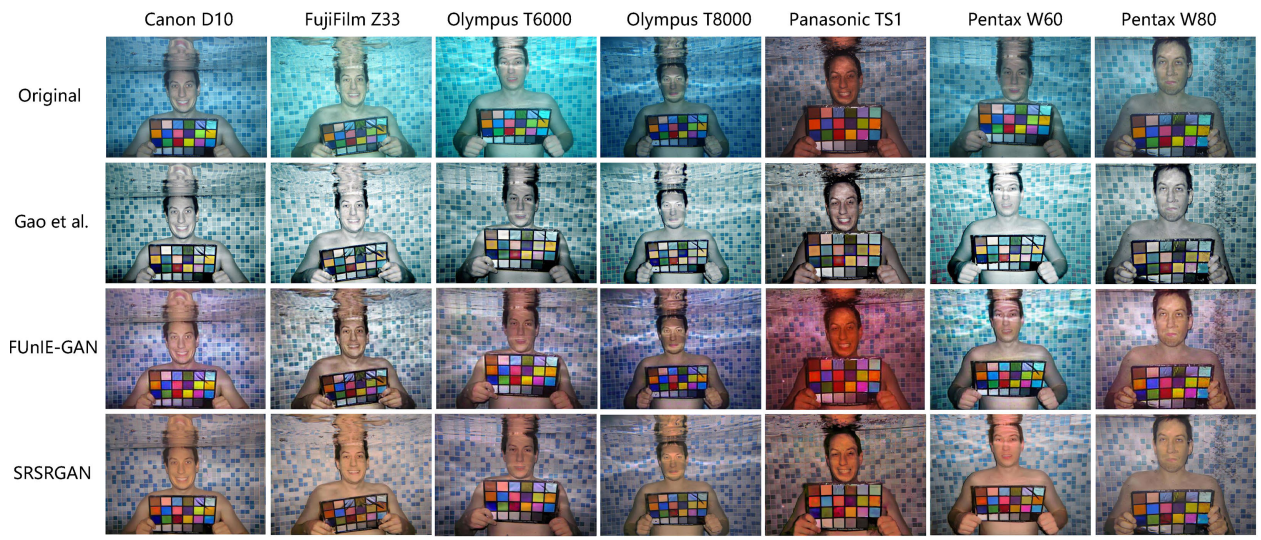


FIGURE 8
The results of SRSRGAN and the compared methods on a set of underwater images taken by different professional cameras, which contain the standard Macbeth Color Checker (Ancuti et al., 2018). The names of the cameras used to take the photos are listed at the top of each column. From top to bottom are original underwater images, the results of the Gao et al. (2019)'s method, FUnIE-GAN (Islam et al., 2020c), and SRSRGAN, respectively. Our sample image is sourced from publicly available image data in the paper (Ancuti et al., 2018). Our sample image is sourced from the public dataset UIEBD (Li et al., 2020).

TABLE 1 Quantitative evaluation on the restored images generated by SRSRGAN and the compared methods on the UIEBD dataset.

Method	PSNR/SSIM	UCIQE	UIQM
UGAN (Fabbri et al., 2018)	19.79/0.7108	0.6299	3.3218
Pix2Pix (Isola et al., 2017)	20.02/0.7230	0.5941	3.1349
CycleGAN (Zhu et al., 2017)	18.71/0.7547	0.5941	3.0144
Water-Net (Li et al., 2020)	19.13/0.7471	0.5721	3.0593
FUnIE-GAN (Islam et al., 2020c)	20.44/0.7257	0.5541	3.1255
Shallow-UWnet (Naik et al., 2021)	20.11/0.728	0.5123	3.0730
TACL (Liu et al., 2022)	20.41/0.733	0.5447	3.168
SRSRGAN	**20.92/0.7731**	**0.6453**	**3.3467**

The best results are shown in boldface.

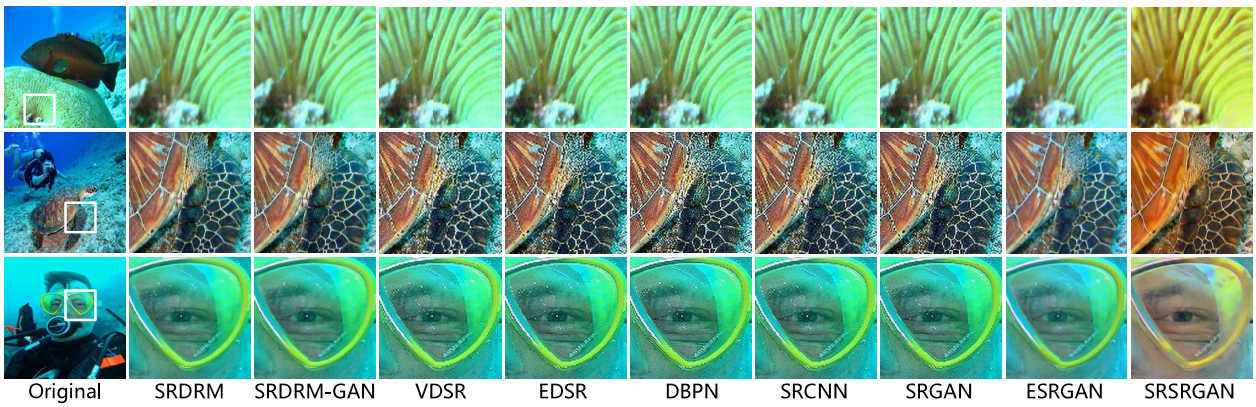


FIGURE 9
Visual comparison with several SOTA SISR methods, including SRDRM (Islam et al., 2020b), SRDRM-GAN (Islam et al., 2020b), VDSR (Kim et al., 2016), EDSR (Lim et al., 2017), DBPN (Haris et al., 2018), SRCNN (Dong et al., 2016), SRGAN (Ledig et al., 2017), and ESRGAN (Wang et al., 2018). Our sample image is sourced from the public dataset UIEBD (Li et al., 2020).

4.4 Evaluation on simultaneous restoration and super-resolution

In this experiment, we compared SRSRGAN with existing methods for simultaneous restoration and super-resolution of underwater images, i.e., Cheng et al. (2018)'s method and SESR (Islam et al., 2020a). The results of the qualitative comparison are shown in Figure 10. It can be seen that Cheng et al. (2018)'s method increases the contrast and brightness of underwater images while the images still have a bluish shift in some patches. It can also be seen from Figure 10 that SESR (Islam et al., 2020a) tends to produce artifacts on the enhanced images. SRSRGAN is more effective in restoring the colors and increasing the resolution of underwater images. This is because the end-to-end trainable model forces the generator of SRSRGAN to complete the ultimate task that restores and super-resolves underwater images simultaneously. In other words, the generator of SRSRGAN benefits from both the restoration stage and the super-resolution stage, so that it can better adapt to the simultaneous restoration and super-resolution task for underwater images.

The quantitative evaluation of SRSRGAN and its comparison methods are shown in Tables 3 and 4. It is obvious that SRSRGAN is effective for color correction, deblurring, and contrast restoration, with the highest scores. In addition, SRSRGAN delivers sharpness and fine-grained texture details with the highest PSNR and SSIM.

TABLE 2 Quantitative evaluation on the underwater image super-resolution on the UIEBD dataset.

Method	PSNR	SSIM	Method	PSNR	SSIM
SRDRM (Islam et al., 2020b)	18.85	0.7102	SRDRM-GAN (Islam et al., 2020b)	18.93	0.7210
VDSR (Kim et al., 2016)	18.36	0.6440	EDSR (Lim et al., 2017)	18.83	0.7166
SRCNN (Dong et al., 2016)	18.78	0.7197	SRGAN (Ledig et al., 2017)	18.54	0.7159
ESRGAN (Wang et al., 2018)	18.22	0.6462	SRSRGAN	**20.92**	**0.7731**

The best results are shown in boldface.

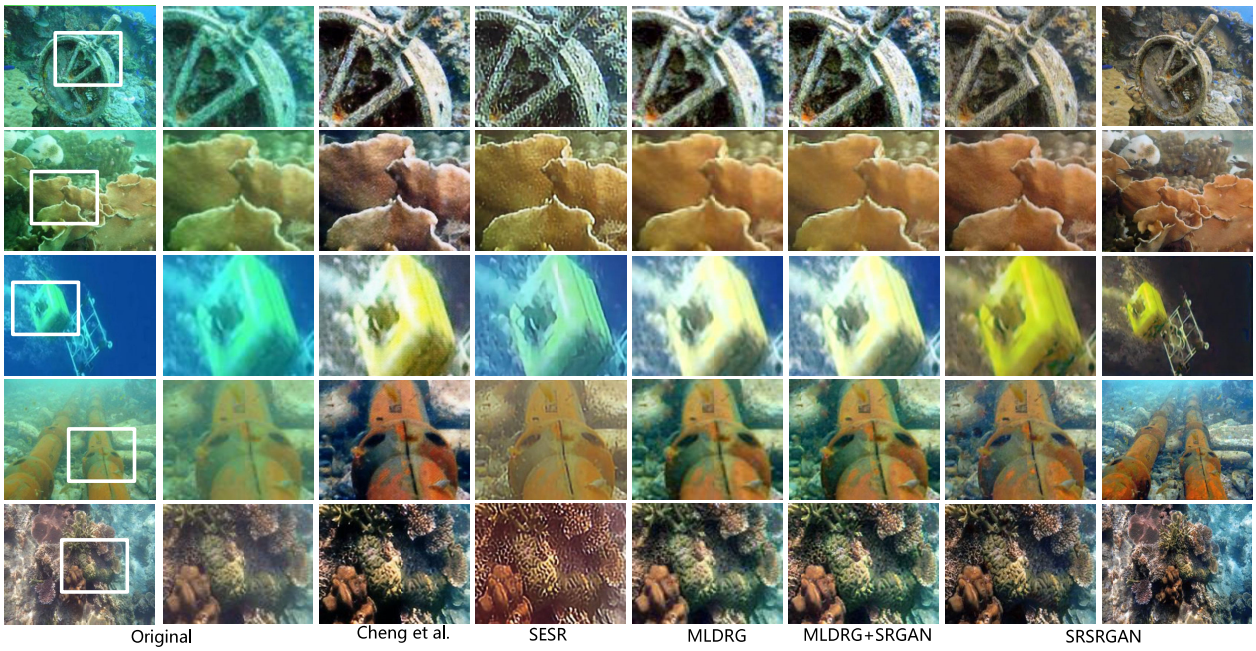


FIGURE 10
Visual comparison between SRSRGAN and several SOTA methods on simultaneous restoration and super-resolution of underwater images, i.e., Cheng et al. (2018) and SESR (Islam et al., 2020a).

4.5 Ablation study

For the proposed SRSRGAN, we added a self-attention block (SAB) to the generator of the restoration stage to learn the important parts of the global features. Meanwhile, we used the spectral normalization (SN) to constrain the Lipschitz continuity of the generators and the discriminators. Furthermore, in the super-resolution stage, we employed the high frequency learning module (HFLM) to excavate fine-grained features from the input LR images. To test the effects of these components in SRSRGAN, we conducted an ablation study to verify their effectiveness.

Table 5 illustrates the performance of SRSRGAN and its variants with different components in terms of PSNR and SSIM. We can see that SAB makes a slight improvement to the performance of SRSRGAN than that without SAB. This is due to the fact that the attention block learns the important parts of the underwater images. Hence, the generator can make more efforts to restore the important parts. In addition, we can see that removing SN from SRSRGAN greatly degrades the performance of SRSRGAN. The reason behind this is that SN effectively guarantees the Lipschitz continuity of SRSRGAN. To be specific, SN stabilizes the training of SRSRGAN by constraining the Lipschitz continuity of its generator and discriminator.

Last but not least, HFLM plays an important role in SRSRGAN, and the variant without HFLM gets the lowest score for both PSNR and SSIM. This can be attributed to the fact that HFLM effectively extracts features of the input images. As a result, with the valid extraction features, the generator of the super-resolution stage can accurately increase the resolution of the images.

5 Conclusion

In this paper, we propose an end-to-end trainable model called SRSRGAN, which is free from the prior of corruption types and levels of underwater images. Meanwhile, SRSRGAN is a unified solution for simultaneous restoration and super-resolution of underwater images. Specifically, it captures underwater degradation information and fine-grained high-frequency information in two stages. Moreover, benefiting from the superior structure of the proposed MLDRG, our model leverages degradation information among different scales, positions, and

TABLE 3 Quantitative evaluation of SRSRGAN and the compared methods on the UFO-120 and USR-248 datasets.

Method	UFO-120		USR-248	
	PSNR	SSIM	PSNR	SSIM
Cheng et al. (2018)	26.03	0.77	27.23	0.81
SESR (Islam et al., 2020a)	27.17	0.77	26.16	0.77
SRSRGAN	**28.08**	**0.78**	**29.13**	**0.85**

The best results are shown in boldface.

TABLE 4 Quantitative evaluation of SRSRGAN and the compared methods on the UIEBD dataset.

Method	PSNR/SSIM	UCIQE	UIQM
Cheng et al. (2018)	19.92/0.7381	0.5792	2.7404
SESR (Islam et al., 2020a)	18.19/0.6917	0.5385	2.7064
MLDRG	18.55/0.6698	0.5389	2.5356
MLDRG+SRGAN	18.36/0.6592	0.5185	2.6286
SRSRGAN	**20.92/0.7731**	**0.6453**	**3.3467**

The best results are shown in boldface.

TABLE 5 Comparisons of the performance of SRSRGAN and its variants with different components on the UIEBD dataset.

Method	PSNR	SSIM
SRSRGAN without SAB and SN	19.29dB	0.7204
SRSRGAN without SAB	19.96dB	0.7528
SRSRGAN without SN	19.75dB	0.7335
SRSRGAN without HFLM	18.79dB	0.7153
SRSRGAN	**20.92*dB***	**0.7731**

The best results are shown in boldface.

channels in the restoration stage to transform degraded images to clean images. Besides, the HFLM excavates fine-grained high-frequency information to super-resolve clean images. Extensive experimental results demonstrate the superiority of SRSRGAN in underwater image restoration, super-resolution, and simultaneous restoration and super-resolution.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding authors.

Author contributions

HW: Conceptualization, Methodology, Model design, Investigation, Writing - Original Draft. GZ and DW: Methodology, Writing - Review and Editing, Funding acquisition, Supervision. JS and YC: Conceptualization, Methodology, Investigation, Writing – Original Draft. YZ: Formal analysis, Writing - Review and Editing. SL: Visualization, Validation. All authors contributed to the article and approved the submitted version.

Funding

This work was partially supported by the National Key Research and Development Program of China under Grant No. 2018AAA0100400, HY Project under Grant No. LZY2022033004, the Natural Science Foundation of Shandong Province under Grants No. ZR2020MF131 and No. ZR2021ZD19, Project of the Marine Science and Technology cooperative Innovation Center under Grant No. 22-05-CXZX-04-03-17, the Science and Technology Program of Qingdao under Grant No. 21-1-4-ny-19-nsh, and Project of Associative Training of Ocean University of China under Grant No. 202265007.

Acknowledgments

We want to thank “Qingdao AI Computing Center” and “Eco-Innovation Center” for providing inclusive computing power and technical support of MindSpore during the completion of this paper.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alenezi, F., Armghan, A., and Santosh, K. (2022). Underwater image dehazing using global color features. *Eng. Appl. Artif. Intell.* 116, 105489. doi: 10.1016/j.engappai.2022.105489

Ancuti, C. O., Ancuti, C., De Vleeschouwer, C., and Bekaert, P. (2018). Color balance and fusion for underwater image enhancement. *IEEE Trans. Image Process.* 27, 379–393. doi: 10.1109/TIP.2017.2759252

Caner, G., Tekalp, A. M., and Heinzelman, W. B. (2003). "Super resolution recovery for multi-camera surveillance imaging," in *International Conference on Multimedia and Expo* (Baltimore, MD, USA: IEEE), Vol. 1. 109–112. doi: 10.1109/ICME.2003.1220866

Capel, D., and Zisserman, A. (2001). "Super-resolution from multiple views using learnt image models," in *IEEE Conference on Computer Vision and Pattern Recognition* (Kauai, HI, USA: IEEE), Vol. 2. II–II. doi: 10.1109/CVPR.2001.991022

Chen, Y., Sun, J., Jiao, W., and Zhong, G. (2019). Recovering super-resolution generative adversarial network for underwater images. *Neural Information Processing* 4, 75–83. doi: 10.1007/978-3-030-36808-1

Chen, T., Wang, N., Wang, R., Zhao, H., and Zhang, G. (2021). One-stage cnn detector-based benthonic organisms detection with limited training dataset. *Neural Networks* 144, 247–259. doi: 10.1016/j.neunet.2021.08.014

Cheng, N., Zhao, T., Chen, Z., and Fu, X. (2018). "Enhancement of underwater images by super-resolution generative adversarial networks," in *International Conference on Internet Multimedia Computing and Service* (Nanjing, Jiangsu, China: IEEE). doi: 10.1145/3240876.3240881

Chiang, J. Y., and Chen, Y.-C. (2012). Underwater image enhancement by wavelength compensation and dehazing. *IEEE Trans. Image Process.* 21, 1756–1769. doi: 10.1109/TIP.2011.2179666

Dong, C., Loy, C. C., He, K., and Tang, X. (2014). "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision* (Zurich, Switzerland: Springer). 184–199.

Dong, C., Loy, C. C., He, K., and Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 295–307. doi: 10.1109/TPAMI.2015.2439281

Drews, J., do Nascimento, E., Moraes, F., Botelho, S., and Campos, M. (2013). "Transmission estimation in underwater single images," in *IEEE International Conference on Computer Vision Workshops* (Sydney, Australia: IEEE). 825–830. doi: 10.1109/ICCVW.2013.113DBLP:conf/iccvw/DrewsNMBC13

Duchon, C. (1979). Lanczos filtering in one and two dimensions. *J. Appl. meteorol.* 18, 1016–1022. doi: 10.1175/1520-0450(1979)018<1016:LFIOAT>2.0.CO;2duchon1979lanczos

Emberton, S., Chittka, L., and Cavallaro, A. (2015) *British Machine Vision Conference (BMVC)* (Swansea, UK: BMVA Press), Vol. 125. 1–125.

Fabbri, C., Islam, M. J., and Sattar, J. (2018). "Enhancing underwater imagery using generative adversarial networks," in *IEEE International Conference on Robotics and Automation* (Brisbane, Australia: IEEE). 7159–7165. doi: 10.1109/ICRA.2018.8460552DBLP:conf/icra/FabbriIS18

Farsiu, S., Robinson, M. D., Elad, M., and Milanfar, P. (2004). Fast and robust multiframe super resolution. *IEEE Trans. Image Process.* 13, 1327–1344. doi: 10.1109/TIP.2004.834669DBLP:journals/tip/FarsiuREM04

Galdran, A., Pardo, D., Picón, A., and Alvarez-Gila, A. (2015). Automatic red-channel underwater image restoration. *J. Visual Communication Image Represent.* 26, 132–145. doi: 10.1016/j.jvcir.2014.11.006DBLP:journals/jvcir/GaldranPPA15

Gao, S.-B., Zhang, M., Zhao, Q., Zhang, X.-S., and Li, Y.-J. (2019). Underwater image enhancement using adaptive retinal mechanisms. *IEEE Trans. Image Process.* 28, 5580–5595. doi: 10.1109/TIP.2019.2919947DBLP:journals/tip/GaoZZZL19

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial networks. *Adv. Neural Inf. Process. Syst.* 3,, 139–144. doi: 10.1145/3422622DBLP:journals/corr/GoodfellowPMXWOCB14

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. (2017). Improved training of wasserstein gans. *Neural Inf. Process. Syst. (NIPS)*, 5769–5779. doi: 10.48550/arXiv.1704.00028

Guo, Y., Li, H., and Zhuang, P. (2020). Underwater image enhancement using a multiscale dense generative adversarial network. *IEEE J. Oceanic Eng.* 45, 862–870. doi: 10.1109/JOE.2019.29114478730425

Haris, M., Shakhnarovich, G., and Ukita, N. (2018). "Deep back-projection networks for super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, Utah: IEEE). 1664–1673. doi: 10.1109/CVPR.2018.00179haris2018deep

Harmeling, S., Sra, S., Hirsch, M., and Schölkopf, B. (2010). "Multiframe blind deconvolution, super-resolution, and saturation correction via incremental em," in *IEEE International Conference on Image Processing (ICIP)* (Hong Kong, China: IEEE). 3313–3316. doi: 10.1109/ICIP.2010.5651650DBLP:conf/icip/HarmelingSHS10

He, K., Sun, J., and Tang, X. (2011). Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 2341–2353. doi: 10.1109/TPAMI.2010.168DBLP:journals/pami/He0T11

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, Nevada: IEEE). 770–778. doi: 10.1109/CVPR.2016.90DBLP:conf/cvpr/kaimingHe2016

Hou, M., Liu, R., Fan, X., and Luo, Z. (2018). "Joint residual learning for underwater image enhancement," in *IEEE International Conference on Image Processing* (Athens, Greece: IEEE). 4043–4047. doi: 10.1109/ICIP.2018.8451209DBLP:conf/icip/HouLFL18

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT, USA: IEEE). 7132–7141. doi: 10.1109/CVPR.2018.00745DBLP:conf/ieeetrans/senet

Hummel, R. (1977). Image enhancement by histogram transformation. *Comput. Graphics Image Process.* 6, 184–195. doi: 10.1016/S0146-664X(77)80011-7hummel1975image

Islam, M., Luo, P., and Sattar, J. (2020a). Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception. *Robot.: Sci. Sys.* doi: 10.15607/RSS.2020.XVI.018DBLP:journals/corr/abs-2002-01155

Islam, M. J., Sakib Enan, S., Luo, P., and Sattar, J. (2020b). "Underwater image super-resolution using deep residual multipliers," in *IEEE International Conference on Robotics and Automation* (Cambridge, Massachusetts, USA: MIT Press). 900–906. doi: 10.1109/ICRA40945.2020.9197213DBLP:conf/icra/IslamELS20

Islam, M. J., Xia, Y., and Sattar, J. (2020c). Fast underwater image enhancement for improved visual perception. *IEEE Robot. Auto. Lett.* 5, 3227–3234. doi: 10.1109/LRA.2020.2974710DBLP:journals/ral/IslamXS20

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI, USA: IEEE). 5967–5976. doi: 10.1109/CVPR.2017.632DBLP:conf/cvpr/IsolaZZE17

Jolicoeur-Martineau, A. (2018). The relativistic discriminator: a key element missing from standard gan. *arXiv*. doi: 10.48550/arXiv.1704.00028

Keys, R. (1982). Cubic convolution interpolation for digital image processing. ieee trans acoust speech signal process. *IEEE Trans. acoustics speech Signal Process.* 29, 1153–1160. doi: 10.1109/TASSP.1981.1163711keys1981cubic

Kim, K. I., and Kwon, Y. (2010). Single-image super-resolution using sparse regression and natural image prior. *IEEE Trans. Pattern Anal. Mach. Intell.* (Las Vegas, NV, USA: IEEE) 32, 1127–1133. doi: 10.1109/TPAMI.2010.25DBLP:journals/pami/KimK10

Kim, J., Lee, J., and Lee, K. M. (2016). "Accurate image super-resolution using very deep convolutional networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1646–1654. doi: 10.1109/CVPR.2016.182DBLP:conf/cvpr/KimLL16a

Kingma, D., and Ba, J. (2014). "Adam: A method for stochastic optimization," in *International Conference on Learning Representations* (Banff, Canada: OpenReview.net).

Köhler, T., Bätz, M., Naderi, F., Kaup, A., Maier, A. K., and Riess, C. (2017). Benchmarking super-resolution algorithms on real data. *CoRR* abs/1904.08444. 10.48550/arXiv.1709.04881

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., et al. (2017). "Photo-realistic single image super-resolution using a generative adversarial network," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI, USA: IEEE). 105–114. doi: 10.1109/CVPR.2017.19DBLP:conf/cvpr/LedigTHCCAATTWS17

Li, C., Guo, C., Ren, W., Cong, R., Hou, J., Kwong, S., et al. (2020). An underwater image enhancement benchmark dataset and beyond. *IEEE Trans. Image Process.* 29, 4376–4389. doi: 10.1109/TIP.2019.2955241DBLP:journals/tip/LiGRCHKT20

Li, X., Hou, G., Li, K., and Pan, Z. (2022). Enhancing underwater image via adaptive color and contrast enhancement, and denoising. *Eng. Appl. Artif. Intell.* 111, 104759. doi: 10.1016/j.engappai.2022.104759DBLP:journals/EAAI/xinjieLi22

Li, X., and Orchard, M. (2000). "New edge directed interpolation," in *IEEE International Conference on Image Processing* (Paris, France: IEEE), Vol. 2. 311–314. doi: 10.1109/ICIP.2000.899369DBLP:conf/icip/LiO00

Li, J., Skinner, K. A., Eustice, R. M., and Johnson-Roberson, M. (2018). Watergan: unsupervised generative network to enable real-time color correction of monocular underwater images. *IEEE Robot. Auto. Lett.* 3, 387–394. doi: 10.1109/LRA.2017.2730363DBLP:journals/ral/LiSEJ18

Lian, H. (2006). "Variational local structure estimation for image super-resolution," in *International Conference on Image Processing* (Atlanta, Georgia, USA: IEEE). 1721–1724. doi: 10.1109/ICIP.2006.312713DBLP:conf/icip/Lian06

Liang, J., Zeng, H., and Zhang, L. (2022). "Details or artifacts: a locally discriminative learning approach to realistic image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition* (New Orleans, LA, US: IEEE). 5647–5656. doi: 10.1109/CVPR52688.2022.00557DBLP:conf/cvpr/jieLiangDA2022

Lim, B., Son, S., Kim, H., Nah, S., and Lee, K. M. (2017). "Enhanced deep residual networks for single image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops* (Honolulu, HI, USA: IEEE). 1132–1140. doi: 10.1109/CVPRW.2017.151DBLP:conf/cvpr/LimSKNL17

Lin, J., Gan, C., and Han, S. (2019). Defensive quantization: when efficiency meets robustness. doi: 10.48550/arXiv.1904.08444DBLP:journals/corr/abs-1904-08444

Liu, Y.-C., Chan, W.-H., and Chen, Y.-Q. (1995). Automatic white balance for digital still camera. *IEEE Trans. Consumer Electron.* 41, 460–466. doi: 10.1109/30.468045liu1995automatic

Liu, R., Jiang, Z., Yang, S., and Fan, X. (2022). Twin adversarial contrastive learning for underwater image enhancement and beyond. *IEEE Trans. Image Process.* 31, 4922–4936. doi: 10.1109/tip.2022.3190209liu2022twin

Lu, H., Li, Y., Nakashima, S., Kim, H., and Serikawa, S. (2017a). Underwater image super-resolution by descattering and fusion. *IEEE Access* 5, 670–679. doi: 10.1109/ACCESS.2017.2648845DBLP:journals/access/LuLNKS17

Lu, H., Li, Y., Uemura, T., Kim, H., and Serikawa, S. (2018). Low illumination underwater light field images reconstruction using deep convolutional neural networks. *Future Gen. Comput. Syst.* 82, 142–148. doi: 10.1016/j.future.2018.01.001DBLP:journals/fgcs/LuLUKS18

Lu, H., Li, Y., Zhang, Y., Chen, M., Serikawa, S., and Kim, H. (2017b). Underwater optical image processing: a comprehensive review. mob. netw. *Appl* 22, 1204–1211. doi: 10.1007/s11036-017-0863-4DBLP:journals/monet/LuLZCSK17

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *Int. Conf. Learn. Represent. (ICLR)*. doi: 10.48550/arXiv.1802.05957

Muresan, D. D. (2005). Fast edge directed polynomial interpolation 2, II–990. doi: 10.1109/ICIP.2005.1530224DBLP:conf/icip/Muresan05

Naik, A., Swarnakar, A., and Mittal, K. (2021). "Shallow-uwnet: compressed model for underwater image enhancement," in *Proceedings of the AAAI Conference on Artificial Intelligence* (Vancouver, Canada: AAAI), Vol. 35. 15853–15854. doi: 10.1609/aaai.v35i18.17923naik2021shallow

Nasrollahi, K., and Moeslund, T. (2014). Super-resolution: a comprehensive survey. *Mach. Vision Appl.* 25, 1423–1468. doi: 10.1007/s00138-014-0623-4DBLP:journals/mva/NasrollahiM14

Panetta, K., Gao, C., and Agaian, S. (2016). Human-visual-system-inspired underwater image quality measures. *IEEE J. Oceanic Eng.* 41, 541–551. doi: 10.1109/JOE.2015.2469915panetta2015human

Reza, A. M. (2004). Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *J. VLSI Signal Process. Syst. signal image video Technol.* 38, 35–44. doi: 10.1023/B:VLSI.0000028532.53893.82DBLP:journals/vlsisp/Reza04

Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., et al. (2016). "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV, USA: IEEE). 1874–1883. doi: 10.1109/CVPR.2016.207DBLP:conf/cvpr/ShiCHTABRW16

Simonyan, K., and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. doi: 10.48550/arXiv.1409.1556

Soni, O. K., and Kumare, J. S. (2020). "A survey on underwater images enhancement techniques," in *IEEE 9th International Conference on Communication Systems and Network Technologies* (Gwalior, India: IEEE). 333–338. doi: 10.1109/CSNT48778.2020.9115732sahu2014survey

Storkey, A. J. (2002). Dynamic structure super-resolution. *Neural Inf. Process. Syst. (NIPS)*.

Timofte, R., De, V., and Gool, L. V. (2013). "Anchored neighborhood regression for fast example-based super-resolution," in *IEEE International Conference on Computer Vision* (Sydney, Australia: IEEE) 15, 1920–1927. doi: 10.1109/ICCV.2013.241DBLP:conf/iccv/TimofteDG13

Timofte, R., Smet, V. D., and Gool, L. J. V. (2014). "A+: adjusted anchored neighborhood regression for fast super-resolution," in *Asian Conference on Computer Vision* (Singapore: Springer), Vol. 9006. 111–126.

Tsai, R. (1984). Multiframe image restoration and registration. *Adv. Comput. Visual Image Process.* 1, 317–339.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). "Attention is all you need," in *Conference on Neural Information Processing Systems* (Long Beach, USA: Curran Associates Inc). 6000–6010.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* 13, 600–612. doi: 10.1109/TIP.2003.819861DBLP:journals/tip/WangBSS04

Wang, N., Chen, T., Kong, X., Chen, Y., Wang, R., Gong, Y., et al. (2023a). Underwater attentional generative adversarial networks for image enhancement. *IEEE Trans. Human-Machine Syst.*, 1–11. doi: 10.1109/THMS.2023.3261341WangUnderwater

Wang, N., Chen, T., Liu, S., Wang, R., Karimi, H. R., and Lin, Y. (2023b). Deep learning-based visual detection of marine organisms: a survey. *Neurocomputing* 532, 1–32. doi: 10.1016/j.neucom.2023.02.018WANG20231

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., et al. (2018). "ESRGAN: enhanced super-resolution generative adversarial networks," in *European Conference on Computer Vision* (Piscataway, NJ, USA: IEEE). 63–79.

Yang, M., and Sowmya, A. (2015). An underwater color image quality evaluation metric. *IEEE Trans. Image Process.* 24, 6062–6071. doi: 10.1109/TIP.2015.2491020DBLP:journals/tip/YangS15

Yang, J., Wright, J., Huang, T., and Ma, Y. (2008). "Image super-resolution as sparse representation of raw image patches," in *IEEE Conference on Computer Vision and Pattern Recognition* (Anchorage, Alaska, USA: IEEE). 1–8. doi: 10.1109/CVPR.2008.4587647DBLP:conf/cvpr/YangWHM08

Yang, J., Wright, J., Huang, T. S., and Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Trans. Image Process.* 19, 2861–2873. doi: 10.1109/TIP.2010.2050625DBLP:journals/tip/YangWHM10

Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.-H., and Liao, Q. (2019). Deep learning for single image super-resolution: a brief review. *IEEE Trans. Multimedia* 21, 3106–3121. doi: 10.1109/TMM.2019.2919431DBLP:journals/corr/abs-1808-03344

Yu, X., Qu, Y., and Hong, M. (2019). Underwater-GAN: underwater image restoration via conditional generative adversarial network. *International Conference on Pattern Recognition*, 66–75. doi: 10.1007/978-3-030-05792-3_7

Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., and Fu, Y. (2018b). "Image super-resolution using very deep residual channel attention networks," in *European Conference on Computer Vision* (Munich, Germany: Springer). 294–310. doi: 10.1007/978-3-030-01234-2_1

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. (2018c). "Residual dense network for image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT, USA: IEEE). 2472–2481. doi: 10.1109/CVPR.2018.00262DBLP:conf/cvpr/ZhangTKZ018

Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017). "Learning deep cnn denoiser prior for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI, USA: IEEE). 2808–2817. doi: 10.1109/CVPR.2017.300DBLP:conf/cvpr/kaiZhangLDPI2017

Zhang, K., Zuo, W., and Zhang, L. (2018a). "Learning a single convolutional super-resolution network for multiple degradations," in *IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT, USA: IEEE). doi: 10.1109/CVPR.2018.00344DBLP:conf/cvpr/kaiZhangDS2018

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision* (Venice, Italy: IEEE). 2242–2251. doi: 10.1109/ICCV.2017.244DBLP:conf/iccv/ZhuPIE17

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 07 July 2023
DOI 10.3389/fmars.2023.1161399

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Peng Ren,
China University of Petroleum (East China), China
Yubo Wang,
Xidian University, China

*CORRESPONDENCE
Zhibin Yu
yuzhibin@ouc.edu.cn

†These authors have contributed equally to this work

RECEIVED 08 February 2023
ACCEPTED 29 May 2023
PUBLISHED 07 July 2023

CITATION
Zheng Z, Xin Z, Yu Z and Yeung S-K (2023) Real-time GAN-based image enhancement for robust underwater monocular SLAM. *Front. Mar. Sci.* 10:1161399. doi: 10.3389/fmars.2023.1161399

COPYRIGHT
© 2023 Zheng, Xin, Yu and Yeung. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Real-time GAN-based image enhancement for robust underwater monocular SLAM

Ziqiang Zheng[1†], Zhichao Xin[2†], Zhibin Yu[2,3*] and Sai-Kit Yeung[1]

[1]Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, Hong Kong SAR, China, [2]Faculty of Information Science and Engineering, Ocean University of China, Qingdao, China, [3]Sanya Oceanographic Institution, Sanya, China

Underwater monocular visual simultaneous localization and mapping (SLAM) plays a vital role in underwater computer vision and robotic perception fields. Unlike the autonomous driving or aerial environment, performing robust and accurate underwater monocular SLAM is tough and challenging due to the complex aquatic environment and the collected critically degraded image quality. The underwater images' poor visibility, low contrast, and color distortion result in ineffective and insufficient feature matching, leading to the poor or even failure of the existing SLAM algorithms. To address this issue, we propose introducing the generative adversarial network (GAN) to perform effective underwater image enhancement before conducting SLAM. Considering the inherent real-time requirement of SLAM, we conduct knowledge distillation to achieve GAN compression to reduce the inference cost, while achieving high-fidelity underwater image enhancement and real-time inference. The real-time underwater image enhancement acts as the image pre-processing to build a robust and accurate underwater monocular SLAM system. With the introduction of real-time underwater image enhancement, we can significantly promote underwater SLAM performance. The proposed method is a generic framework, which could be extended to various SLAM systems and achieve various scales of performance gain.

KEYWORDS

generative adversarial networks, SLAM, knowledge distillation, underwater image enhancement, real-time, underwater SLAM

1 Introduction

Recently, many vision-based state estimation algorithms have been developed based on the monocular, stereo, or multi-camera systems in indoor (García et al., 2016), outdoor (Mur-Artal and Tardós, 2017; Campos et al., 2021), and underwater environments Rahman et al. (2018); Rahman et al. (2019b). Underwater SLAM (Simultaneous Localization and Mapping) is an autonomous navigation technique used by underwater robots to build a map of an unknown environment and localize the robot within the map. Underwater SLAM provides a safe, efficient, and cost-effective way to explore and survey unknown

underwater environments. Specifically, monocular visual SLAM provides an effective solution to many navigation applications Bresson et al. (2017), detecting unknown environments and assisting in decision-making, planning, and obstacle avoidance based on only a single camera. Monocular cameras are the most common vision sensors, which are inexpensive and ubiquitous mobile agents, making them a popular choice of sensor for SLAM.

There has been increasing attention on using an autonomous underwater vehicle (AUV) or remotely operated underwater vehicle (ROV) to conduct the monitoring of marine species migration Buscher et al. (2020) and coral reefs Hoegh-Guldberg et al. (2007), the inspection of submarine cables and wreckage Carreras et al. (2018), deep ocean exploration Huvenne et al. (2018) and underwater cave exploration Rahman et al. (2018); Rahman et al. (2019b). Unlike atmospheric imaging, the captured underwater images have issues with low contrast and color distortion due to the strong scattering and absorption phenomena. In detail, underwater pictures are usually critically degraded due to large suspended particles, poor visibility, and under-exposure. Thus it is complex and challenging to detect robust features to track for visual SLAM systems. As a result, directly performing the current available vision-based SLAM usually cannot obtain a satisfactory and robust result.

To address this issue, Cho et al. Cho and Kim (2017) combined Contrast-limited Adaptive Histogram Equalization (CLAHE) Reza (2004) to conduct real-time underwater image enhancement to promote the underwater SLAM performance. Furthermore, Huang et al. Huang et al. (2019) performed underwater image enhancement by converting RGB images to HSV space and then performing color correction based on Retinex theory. Then the enhanced outputs were applied for downstream underwater SLAM. However, these methods only achieved marginal improvement and could not work in highly turbid conditions.

Generative adversarial networks (GANs) Goodfellow et al. (2014) had been adopted for underwater image enhancement Anwar and Li (2020); Islam et al. (2020a) to boost underwater vision perception. Compared 48 with the model-free enhancement methods Drews et al. (2013); Huang et al. (2019), GAN-based image-to image (I2I) translation algorithms could enhance textile and content representations and generate realistic images with clear and plausible features Ledig et al. (2017), especially in highly turbid conditions Han et al. (2020); Islam et al. (2020c). This line of research has mostly taken place in the computer vision fields, with the main focus on underwater single image restoration Akkaynak and Treibitz (2019); Islam et al. (2020b). Benefiting from the superior performance of GAN-based approaches, some researchers attempted to use CycleGAN to boost the performance of ORB-SLAM in an underwater environment Chen et al. (2019). The experimental results have shown that CycleGAN-based underwater image enhancement can lead to more matching points in a turbid environment. However, CycleGAN Zhu et al. (2017) could not meet the real-time requirement. Nevertheless, CycleGAN-based underwater image enhancement may also increase the risk of incorrect matching pairs. Besides, the feature matching analysis and detailed quantitative SLAM results are missing in Chen et al. (2019) for discussing the potential of adopting the underwater enhancement for promoting underwater SLAM performance in real-world underwater environments. To address these issues, we aim to comprehensively analyze this point.

In this paper, we target to perform a lightweight GAN-based image enhancement framework for underwater monocular SLAM to promote performance. The proposed GAN-based image enhancement can promote the feature matching performance (Please refer to section 4.3.2 and Figure 1 for more details), which can further lead to better and more robust SLAM results. To speed

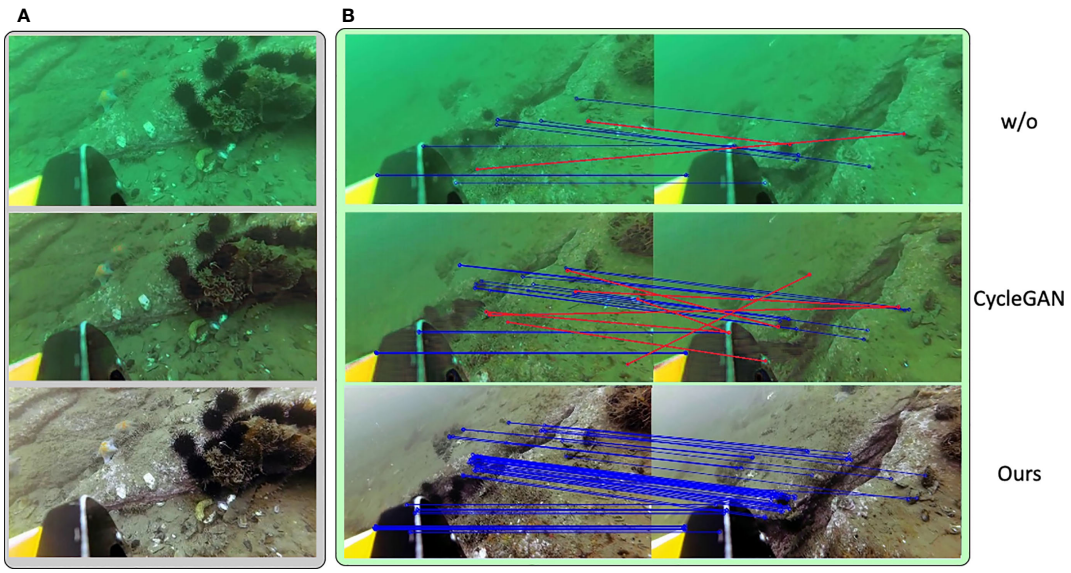


FIGURE 1
The illustrations of **(A)** underwater images and **(B)** the ORB [41] feature-matching under three settings: without any enhancement, with CycleGAN, and with our method. Blue lines represent correct feature matching pairs and Red lines represent incorrect feature matching pairs. The proposed underwater enhancement can significantly promote feature-matching performance. Best viewed in color.

up the underwater image enhancement progress and reduce the risk of incorrect matching pairs, we propose to perform GAN compression Li et al. (2020b) to accelerate underwater image enhancement inference. The knowledge distillation Aguinaldo et al. (2019) is adopted to reduce the computational costs and the inference time. We propose a generic robust underwater SLAM framework shown in Figure 2, which could be extended to various SLAM systems (e.g., ORB-SLAM2 Mur-Artal and Tardós (2017), Dual-SLAM Huang et al. (2020) and ORB-SLAM3 Campos et al. (2021)) and achieve a performance gain with the real-time GAN-based underwater image enhancement module. The proposed method performs favorably against state-of-the-art methods in both position estimation and system stability. To sum up, our main contributions are listed as follows:

- We introduce a generic robust underwater monocular SLAM system, which can be extended to different SLAM algorithms and achieve a large performance gain.
- To accelerate GAN-based image enhancement, we perform GAN compression through knowledge distillation for performing real-time underwater image enhancement as a compelling image pre-processing module. As a result, we can obtain more robust, stable, and accurate state estimation outputs.
- Our method can achieve current state-of-the-art performance and tailored analysis about 1) underwater image enhancement, 2) robust and accurate feature matching, and 3) SLAM performance is included in our paper.

2 Related work

2.1 Underwater image enhancement

The underwater image enhancement algorithms could mainly fall into three categories: 1) model free Asmare et al. (2015); 2) model-based Akkaynak and Treibitz (2019) and 3) data-driven Li et al. (2018); Islam et al. (2020b); Islam et al. (2020c) algorithms. The representative model-free CLAHE Reza (2004) method could enhance an underwater image without the image formation process. Asmare et al. Asmare et al. (2015) converted the images into the frequency domain and proposed to enhance the high-frequency component to promote the image quality. Though these model-free methods could perform image enhancement with a very high speed, they still heavily suffered from over-enhancement, color distortion, and low contrast Li et al. (2020a), and they only achieved slight improvement under highly turbid conditions. Model-based methods considered the physical parameters and formulated an explicit image formation process. Drews et al. Drews et al. (2013) proposed to apply dark channel prior He et al. (2010) in the underwater setting to perform underwater dehazing. The Sea-Thru method Akkaynak and Treibitz (2019) firstly proposed to estimate the backscattering coefficient and then recover the color information with the known range based on RGB-D images. However, collecting a large-scale underwater RGB-D image dataset is expensive and time-consuming. The latter data-driven underwater image enhancement algorithms Li et al. (2018); Han et al. (2020); Islam et al. (2020b); Islam et al. (2020c) combined deep CNNs to conduct underwater image restoration based on large-scale paired or unpaired data. UWGAN Li et al. (2018) proposed to combine multi-style underwater image synthesis for the underwater depth estimation. SpiralGAN Han et al. (2020) proposed a spiral training strategy to promote image enhancement performance. FUnIE-GAN Islam et al. (2020c) could perform real-time underwater image enhancement for underwater object detection. Unlike this object-level enhancement algorithm, we target to perform real-time GAN-based underwater image enhancement for a more challenging underwater SLAM, which requires high-fidelity pixel correspondences.

2.2 Underwater SLAM

The popular ORB-SLAM Mur-Artal and Tardós (2017); Elvira et al. (2019) introduced an efficient visual SLAM solution based on

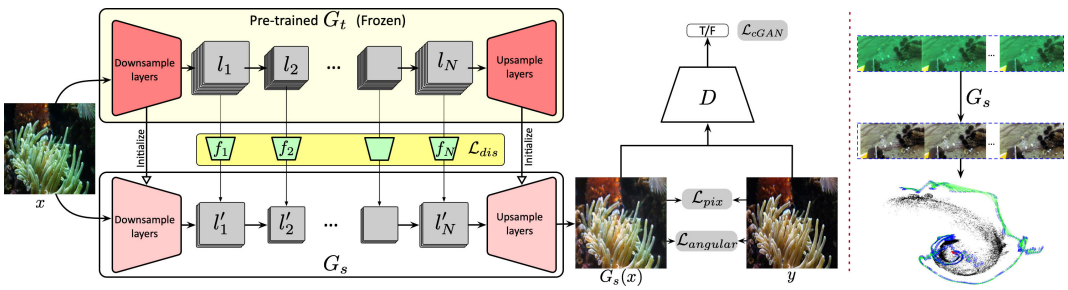


FIGURE 2
The overview framework of the proposed method. The left part on the red dotted line illustrates the GAN-based underwater image enhancement module. In contrast, the right part shows the downstream underwater monocular SLAM with learned G_s for real-time inference. G_t and G_s indicate the teacher and student generators, respectively. Given a pre-trained G_t, we aim to conduct GAN compression through knowledge distillation. $\{l_n\}_1^N$ represent the chosen N layers of G_t to compress and $\{l'_n\}_1^N$ indicate the compressed layers of G_s. f_n indicates the additional convolutional layer to achieve channel reduction to achieve shape matching between the intermediate layers of G_t and G_s. $\mathcal{L}_{dis}$ is computed to transfer the learned knowledge of G_t to G_s. We compute pixel-wise loss $\mathcal{L}_{pix}$, angular loss $\mathcal{L}_{angular}$ and conditional GAN loss $\mathcal{L}_{cGAN}$ between $G_s(x)$ and y.

ORB feature descriptor Rublee et al. (2011). VINS Qin et al. (2018); Qin and Shen (2018) proposed a general monocular framework with the IMU information. Unlike the aerial setting, underwater is a typical global positioning system (GPS) denied environment, where visual information provides valuable navigation queues for robot navigation. Currently, without the GPS for camera pose ground truth generation, a recent work Ferrera et al. (2019a) adopted Colmap Schönberger and Frahm (2016); Schönberger et al. (2016) to generate relatively precise camera trajectory based on structure-from-motion (SFM). UW-VO Ferrera et al. (2019b) further adopted the generated trajectory as ground truth to evaluate the underwater SLAM performance. Because of the good properties of sound 116 propagation in the water, some sonar-based methods Rahman et al. (2018); Rahman et al. (2019a; Rahman et al. (2019b) (e.g., SVIN Rahman et al. (2018) and SVin2 Rahman et al. (2019b)) combined the additional sparse depth information from the sonar sensor to perform more accurate position estimation. However, these are more suited for long-range underwater missions rather than close-range ones. Besides, the sonar sensor is still expensive, and we target to propose a general underwater SLAM framework based on the visual information.

3 Methodology

3.1 Overall framework

We aim to propose a generic robust underwater monocular SLAM framework, which contains two main procedures: **Real-time GAN-based Underwater Image Enhancement** and **Downstream Underwater SLAM** based on the enhanced underwater images generated from the former stage. First, we refer the readers to check the overall framework in Figure 2. To perform real-time GAN-based I2I translation for underwater image enhancement, we adopt the knowledge distillation Aguinaldo et al. (2019)for GAN compression to achieve better performance-speed tradeoff. The network parameters and computational costs could be heavily reduced after compression while achieving comparable or even better underwater enhancement performance.

3.2 GAN-based underwater image enhancement

To achieve underwater image enhancement from a source domain $\mathbb{X}$ to a target domain $\mathbb{Y}$ (*e.g.*, the source turbid underwater image domain and another target clear underwater image domain). The conditional GAN pipeline Mirza and Osindero (2014) is chosen in our work since it could generate more natural and realistic image outputs based on full supervision. For generating reasonable image outputs in the target domain, the adversarial loss is applied:

$$\mathcal{L}_{cGAN} = \mathbb{E}_{x,y}[\log\ D(x,y)] + \mathbb{E}_{x}[\log\ (1 - D(x, G_s(x)))], \quad (1)$$

where x and y are image samples from $\mathbb{X}$ and $\mathbb{Y}$, respectively. The adversarial loss $\mathcal{L}_{cGAN}$ Isola et al. 137 (2017) could reduce the distance between the generated sample distribution and the real sample distribution. Besides the adversarial loss, the pixel-wise $\mathcal{L}_{pix}$ is also included to measure the pixel difference (1-norm) between the generated image output and the corresponding real clear image:

$$\mathcal{L}_{pix} = \| G_s(x) - y \|_1; \quad (2)$$

please note that we compute both $\mathcal{L}_{pix}$ and $\mathcal{L}_{cGAN}$ based on the output of G_s rather than G_t.

Angular loss. To further promote the naturalness of synthesized outputs, we adopt the angular loss $\mathcal{L}_{angular}$ Han et al. (2020) to obtain better image synthesis:

$$\mathcal{L}_{augular} = \mathbb{E}_{\mathbb{X},\mathbb{Y}}[\angle (G_s(x), y)], \quad (3)$$

where $\angle$ indicates the angular distance between $G_s(x)$ and y in RGB space. It is observed that the used $\mathcal{L}_{angular}$ could lead to better robustness and enhancement outputs to some critical over-under exposure problems in the underwater images. The color distortion could be effectively alleviated by $\mathcal{L}_{angular}$. Through the integration of the above-mentioned loss functions, we could achieve effective and reasonable underwater image enhancement. However, it cannot meet the real-time inference requirement in Isola et al. (2017); Han et al. (2020); Zhu et al. (2017).

3.3 GAN compression through knowledge distillation

We perform GAN compression through knowledge distillation to save computational costs and achieve the tradeoff between enhancement performance and inference speed. The detailed design of the proposed GAN compression module is shown in Figure 2, which contains the teacher generator G_t, the student generator G_s, and the discriminator D. In detail, we transfer the learned knowledge learned from G_t to G_s by matching the distribution of the feature representations. We initialize the teacher network G_t with a pre-trained underwater enhancement model and G_t is frozen during the whole training procedure. The optimized teacher network could guide the student network on extracting effective feature representations and achieving better enhancement performance. The distillation objective can be formulated as:

$$\mathcal{L}_{dis} = \sum_{n=1}^{N} \| f_n(G_s^n(x)) - G_t^n(x) \|_2, \quad (4)$$

where $G_s^n(x)$ and $G_t^n(x)$ (*with channel number* $c = 16$) are the intermediate feature representations of the n-th selected feature layer in G_s and G_t, and N denotes the number of selected layers. f_n is the convolutional layer with 1×1 kernel to achieve channel reduction, which will not introduce many training parameters. In our experiments, we set $N = 6$ and select the middle intermediate feature representations. Different from Li et al. (2020b), we do not perform a neural architecture search (NAS) considering its huge time consumption. The channel number in G_s is set to 16. More

ablation studies about the channel number c selection can be found in Sec. 4.4.

3.4 Full objective function

We update the final objective function of the proposed method as:

$$\mathcal{L} = \mathcal{L}_{cGAN} + \mathcal{L}_{dis} + \mathcal{L}_{angular} + \lambda \mathcal{L}_{pix}, \quad (5)$$

where λ is a hyper-parameter to balance the loss component. We set $\lambda = 10$ in our experiments following the setup in Isola et al. (2017) to better balance the contribution of pixel-wise supervision and other components in the proposed method.

3.5 Downstream underwater SLAM

For the downstream monocular SLAM module, we have explored different in-air SLAM systems: **ORB-SLAM2**, **Dual-SLAM** and **ORB-SLAM3** to perform state estimation based on the enhanced underwater images after the image resizing for obtaining the approximate image inputs. To be noted, the two modules are optimized separately and the SLAM system is running in a hard-core engineering manner. The in-air visual SLAM algorithms underperform in the aquatic environment as the critical image degradation. With the real-time GAN-based underwater image enhancement module, the model could better model the complex marine environment and find robust features to track from the enhanced underwater images, which leads to more stable and continuous SLAM results. Besides, the proposed framework could be extended to various SLAM systems to achieve performance gain.

4 Experiments

In this section, we first provide the implementation details of the proposed method and review the experimental setup. Then we report the inference speed comparison of different underwater image enhancement algorithms, followed by the detailed performance comparison of different algorithms. Next, we target to analyze the underwater SLAM performance from three aspects: 1) underwater image enhancement performance, 2) feature matching analysis, and 3) qualitative and quantitative underwater SLAM performance of different SLAM baselines under various settings. Finally, we provide ablation studies to explore the tradeoff between the underwater enhancement performance and the inference speed.

4.1 Implementation details and experimental setup

4.1.1 Implementation details

To obtain a lightweight and practical underwater enhancement module, we perform the knowledge distillation to compress the enhancement module to meet the real-time inference requirement. The trained SpiralGAN model (also other GAN models) is chosen as the teacher model G_t to stabilize the whole training procedure and speed up the convergence. It is worth noting that G_t is frozen when performing the knowledge distillation. The image resolution of underwater enhancement is set to 256×256, and we perform upsampling to resize the enhanced image outputs to 640×480 based on bilinear interpolation for further SLAM. The hyperparameter c for selected feature layers is set to 16, and we include the discussion about choices of c in our ablation studies. For optimizer, we choose Adam optimizer Kingma and Ba (2014) in all our experiments and set the initial learning rate to 0.0002.

4.1.2 Datasets

4.1.2.1 Training datasets for underwater image enhancement

We adopt the training dataset from the previous work Fabbri et al. (2018), which contains 6,128 paired underwater turbid-clear images synthesized from CycleGAN Zhu et al. (2017). The proposed method has been only trained with one underwater dataset and can be extended to different unseen underwater image sequences for performing underwater image enhancement.

4.1.2.2 Datasets for underwater SLAM

URPC dataset contains a monocular video sequence collected by the ROV in a real aquaculture farm. The ROV navigates at a water depth of about 5 meters. Operating ROV collected a total of 190 seconds of a video sequence with an acquisition frequency 24Hz. A total of 4,538 frames of RGB images (640×352 image resolution) were obtained. The collected video sequence has large scene changes and low water turbidity. The image suffers severe distortion, and the watercolor is bluish-green. Considering the first 2,000 consecutive images do not contain meaningful objects, we remove them and only choose the last 2,538 images for experimental testing. We choose the open-source offline SFM library Colmap Schönberger and Frahm (2016); Schönberger et al. (2016) to generate the camera pose trajectory for evaluating the underwater SLAM performance. OUC fisheye Zhang et al. (2020) dataset is a monocular dataset collected by the fisheye camera in a highly turbid underwater environment. It provides 10 image sequences from three water turbidity: 1) slight water turbidity with about 6m visibility; 2) middle water turbidity with about 4m visibility and 3) high water turbidity with about 2m visibility. The image sequences are collected with the acquisition frequency of 30Hz and each sequence lasts about 45 seconds. In our experiments, we evaluate an image sequence containing 1,316 frames ($1,920 \times 1,080$ image resolution) with high water turbidity. The severe distortion and heavy backscattering lead to a significant influence on feature tracking. The trajectory is also generated from Colmap Schönberger and Frahm (2016); Schönberger et al. (2016).

4.1.3 SLAM baselines

We chose three baselines: ORB-SLAM2, Dual-SLAM and ORB-SLAM3 for comparison:

- ORB-SLAM2 Mur-Artal and Tardós (2017) is a complete SLAM system for monocular, stereo, and RGB-D cameras. The adopted ORB-SLAM2 system has various applications for indoor and outdoor environments. We choose it as the baseline for performing underwater mapping and reconstruction.
- Dual-SLAM Huang et al. (2020) extended ORB-SLAM2 to save the current map and activate two new SLAM threads: one is to process the incoming frames for creating a new map and another is to link the created new map and older maps together for building a robust and accurate system.
- ORB-SLAM3 Campos et al. (2021) perform visual, visual-inertial, and multi-map SLAM based on monocular, stereo, and RGB-D cameras, which has achieved current state-of-the-art performance and provided a more comprehensive analysis system.

4.1.4 Evaluation metric for SLAM

To measure the SLAM performance, we choose 1) Absolute Trajectory Error (ATE), 2) Root Mean Square Error (RMSE), and 3) Initialization performance for evaluation. ATE directly calculates the difference between the camera pose ground truth and the estimated trajectory from SLAM. RMSE can describe the rotation and translation errors of the two trajectories. The smaller the RMSE is, the better the system trajectory fits. The initialization performance indicates the number of frames to perform the underwater SLAM initialization. The lower the initialization frames, the better SLAM performance, and more stable and continuous outputs. To make a fair comparison, we repeat the underwater SLAM experiments 5 times to obtain the best result for all methods.

4.2 Inference speed comparison

In this section, we target to provide the inference speed comparison of different underwater image enhancement methods under the same experimental setting. For underwater image enhancement methods, we choose CLAHE Reza (2004), UDCP Drews et al. (2013) and FUnIE-GAN Islam et al. (2020c) for underwater image enhancement comparison. To measure the frames per second (FPS) for different methods, we test the speed of various methods on the practical Jetson AGX Xavier, which is widely equipped on underwater ROVs and AUVs. The detailed FPS and memory access cost (MAC) comparison is shown in Table 1 (the testing image resolution is set to 640×480 (default image resolution of ORB-SLAM2 and ORB-SLAM3) for all methods to make a fair comparison). Compared with the default image resolution (256×256) adopted in FUnIE-GAN, the proposed FPS computation setting is more practical and can lead to more reasonable and accurate translated outputs. As reported, UDCP has a very low underwater image enhancement speed, and it costs several seconds to process only one image. Besides, SpiralGAN and FUnIE-GAN cannot perform real-time (*e.g.*, ≥ 30) underwater image enhancement. Our method has fewer network parameters and can achieve real-time GAN-based underwater image enhancement.

4.3 Performance comparison

4.3.1 Underwater image enhancement results

Firstly, we target to demonstrate that the proposed method could generate high-quality image synthesis outputs after the underwater image enhancement module. We have provided a direct comparison with the model-free image enhancement algorithms (CLAHE and UDCP) and GAN-based image enhancement method (FUnIE-GAN) in Figure 3 on the URPC dataset. Compared with the previous model-free image enhancement methods, the proposed method could enhance the content representations of the objects. The synthesis image by FUnIE-GAN has visible visual artifacts. In contrast, the proposed GAN-based image enhancement method could achieve better results with more reasonable outputs. To be noted, the proposed method has been only trained on one underwater dataset and can be extended to different unseen underwater image sequences for testing. The strong generalization ability could alleviate the efforts of the model-based algorithms to change the physical parameters, which is also time-consuming. The GAN-based image enhancement module has shown powerful effectiveness and achieved better results. We provide more underwater image enhancement result comparisons in our supplementary.

4.3.2 Feature matching analysis

We have designed comprehensive feature-matching experiments to reveal whether the proposed underwater image enhancement

TABLE 1 Quantitative FPS, MACs(G) and Parameter(M) comparison of various methods.

Method	FPS ↑	MACs (G) ↓	Parameter(M) ↓
CLAHE	**260.0**	–	–
UDCP	0.041	–	–
FUnIE-GAN	7.36	47.96	7.02
CycleGAN	3.25	266.40	11.38
SpiralGAN	17.63	34.75	4.99
Proposed	31.83	**8.81**	**1.28**

The image resolution is set to 640×480 for evaluation based on practical Jetson AGX Xaviver. ↑ (↓) indicates that the larger (smaller) the value is, the better the performance. The best results are in bold. "-" means not applicable.

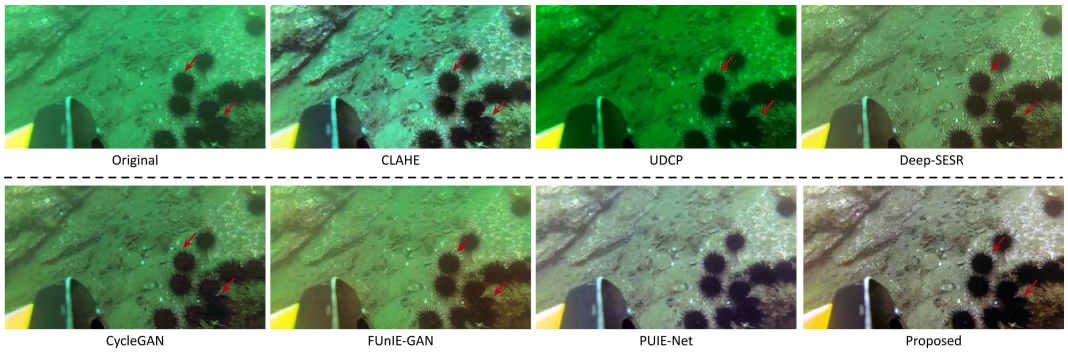


FIGURE 3
The qualitative results of different underwater image enhancement methods. Best viewed in color.

could promote the feature-matching performance for SLAM. First, following the experimental setup in Cho and Kim (2017), we report the ORB, SIFT, and SURF feature-matching results. For a fair comparison, 500 same image pairs are chosen for performing feature matching based on various feature descriptors with two different frame intervals: 20 and 30. If the matching points number is larger than 50, we regard the matching as successful and report the successful matching rate. The detailed results are illustrated in Figure 4. Besides, we also provide the average number of matching points of different feature descriptors. Compared with feature matching performance conducted on the original images, UDCP Drews et al. (2013) could only lead to marginal improvement or slight degradation. FUnIE-GAN Islam et al. (2020c) failed to generate reasonable enhanced image outputs with plausible textile information. There is an observable performance degradation compared with the "original" setting. In contrast, the proposed method can improve performance under all settings.

Furthermore, to verify that the yielded feature matching points are valid interior points, we conduct feature point matching evaluation through reprojection. In detail, the feature points extracted from the current frame are reprojected to the previous 20th image frame. We obtain the ground truth feature matching based on Structure-from-Motion. For defining accurate feature matching points; we choose a 3×3 pixel area:

- When the distance between the projected point (computed based on the estimated transformation matrix H and the intrinsic camera parameter K) and the detected feature

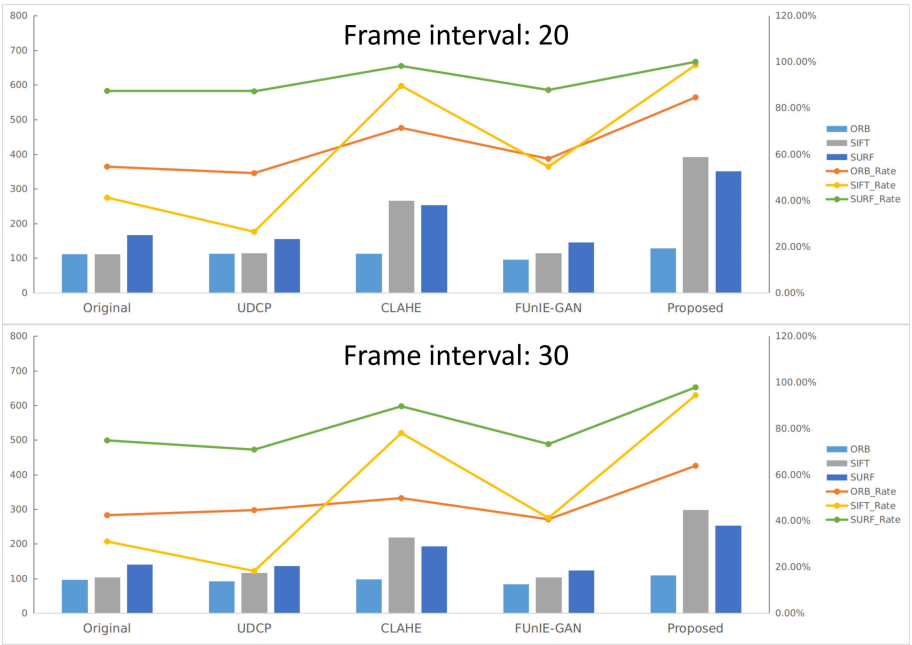


FIGURE 4
The qualitative feature matching results of various methods based on different feature descriptors. The lines and the bars indicate the feature matching success rates and average matching points based on various feature descriptors, respectively.

point is less than 3, such detected feature matching points are marked as tbfInner points (denoted as P_i),

- Other detected feature points are falsely matched as Outlier points (denoted as P_s).

We have provided the qualitative feature matching performance under three settings: 1) w/o underwater enhancement, 2) enhancement by CycleGAN, and 3) our method in Figure 1. As reported, CycleGAN adopted in Chen et al. (2019) a as a pre-processing module could increase the number of correct matching pairs. However, the number of incorrect matching pairs also increased. The proposed method can significantly increase the number of correct matching pairs with few errors.

For the quantitative comparison, we compute the error rate statistically based on 100 pairs as follows:

$$\text{Err}. = \frac{P_o}{P_i + P_o}. \tag{6}$$

The proposed method could achieve a matching error rate of 1.2%, significantly outperforming the error rate of 11.5% achieved by CycleGAN. The error rate of 10.1% under the setting without underwater enhancement is also reported for better comparison. As reported, the proposed method could effectively promote the feature matching performance. Finally, it is worth noting that Chen et al. (2019) did not conduct feature matching accuracy analysis.

4.3.3 Qualitative and quantitative results

In this section, we aim to provide both qualitative and quantitative underwater SLAM performance comparisons using real-world underwater datasets. Similarly, the qualitative image enhancement results on both URPC and OUC fisheye datasets are reported in Figure 5. Our method could effectively alleviate the over-under exposure problem and increases contrast and brightness. Besides, our enhancement module could render more details and utilize previous content representations from the original input images. We combine different image enhancement methods with ORB-SLAM2 to explore the improvement of underwater SLAM performance on the URPC dataset. Due to the fact that it is time-consuming to perform UDCP, we do not perform UDCP for the downstream underwater SLAM. The quantitative SLAM performance comparison can be found in Table 2. The proposed GAN-based underwater image enhancement method could heavily promote underwater SLAM performance with a real-time processing inference time. On the other hand, the FUnIE-GAN cannot synthesize enhanced outputs and there is a performance degradation compared to the SLAM performance conducted on the original underwater images.

Furthermore, we combine the GAN-based underwater image enhancement module with two SLAM systems: Dual-SLAM and ORB-SLAM3. The quantitative results are shown in Table 3. The estimated camera pose trajectory is more stable and the initial performance has been promoted heavily. The reasonable image

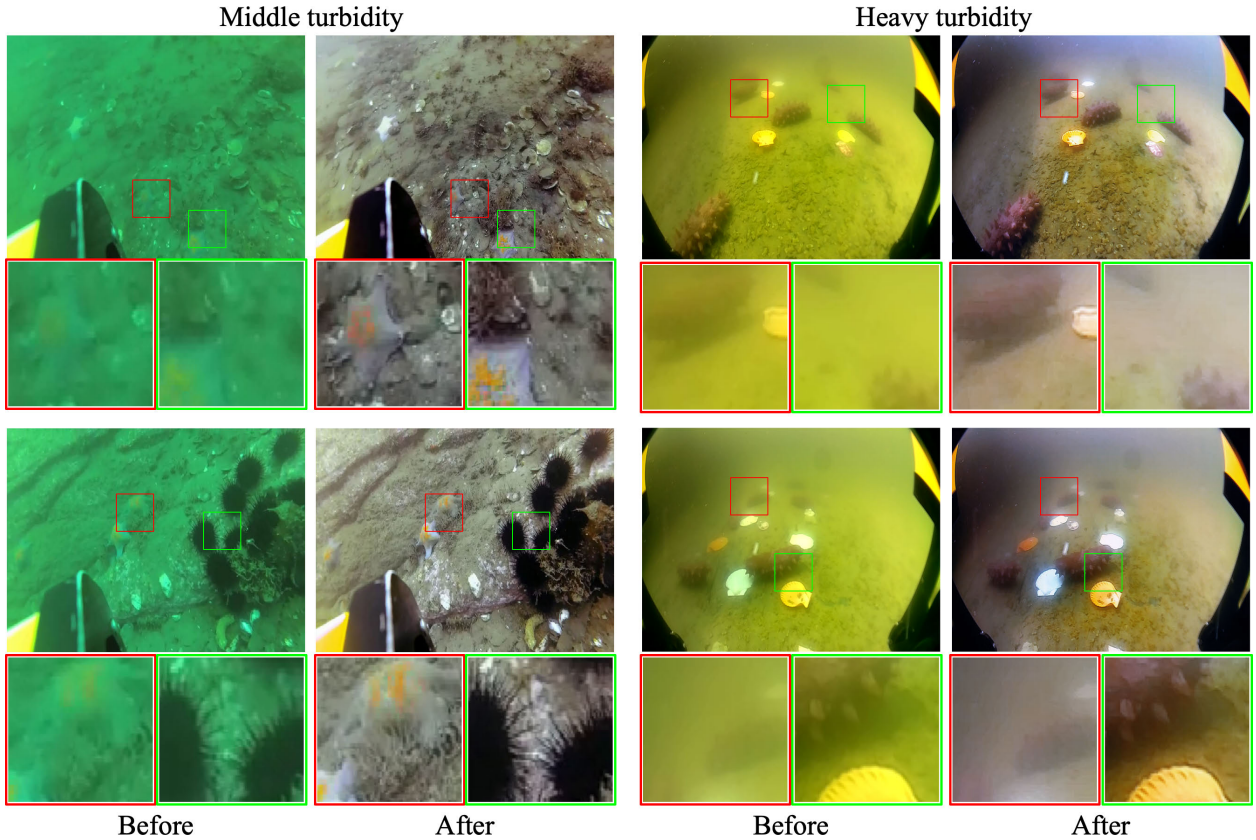


FIGURE 5
The qualitative results of our GAN-based underwater image enhancement on **(A)** URPC dataset and **(B)** OUC fisheye dataset. Best viewed in color.

TABLE 2 Quantization error ORB-SLAM2 baseline with different enhancement methods on the URPC dataset.

Method		ATE ↓	RMSE ↓	Initialization ↓
ORB-SLAM2	w/o	1.418	1.484	84
ORB-SLAM2	CLAHE	1.397	1.468	61
ORB-SLAM2	FUnIE-GAN	1.474	1.505	136
ORB-SLAM2	CycleGAN	1.501	1.565	159
ORB-SLAM2	SpiralGAN	1.348	**1.446**	24
ORB-SLAM2	Proposed	**1.344**	1.447	**23**

The best results are in bold.

enhancement could result in more reliable feature matching so that our method could achieve more stable and accurate outputs. The image enhancement module could promote the underwater SLAM performance in all metrics. Besides, the qualitative trajectory results are also included in Figure 6. The proposed framework outperforms current SLAM methods in both qualitative and quantitative evaluations.

4.3.4 Highly turbid setting

Comprehensive experiments have demonstrated that the proposed method can generate realistic enhanced images with high fidelity and image quality, which can be applied to promote underwater monocular SLAM performance. To further demonstrate the effectiveness and the generalization performance of the proposed framework, we perform experiments on OUC fisheye dataset Zhang et al. (2020). For better illustration, we provide the original underwater image and the enhanced output image in Figure 5B. Similarly, the quantitative and qualitative results under various settings are reported in Table 4 and Figure 7, respectively. The proposed framework can also promote SLAM performance under various challenging settings.

4.4 Ablation studies

4.4.1 Tradeoff between enhancement performance and inference speed

To better explore the performance-computation tradeoff, we have conducted experiments using different values of $c$in G_s. We report the computational costs, inference time, and SLAM results in Table 5. SpiralGAN Han et al. (2020) sets $c = 32$ and the proposed compressed method ($c = 16$) has achieved comparable or even better performance with higher speed. When $c = 8$, though it could perform real-time underwater image enhancement with a very high inference speed (FPS=47.54), there is a noticeable enhancement performance drop compared with the proposed method ($c = 16$).

5 Discussions

In this work, the GAN-based image enhancement module and the downstream visual SLAM are optimized separately. The image enhancement is only adopted as an effective image pre-processing module. We assume that the enhanced image could have higher image quality. However, if the GAN-based module cannot generate reasonable images, there would be performance degradation for the SLAM system. The wrong enhanced underwater outputs could lead to error accumulation. We target to optimize the two modules in a multi-task learning manner. The two modules could be mutually beneficial. Besides, we target to build a general open-source underwater SLAM framework which is robust to various underwater conditions. Furthermore, we also target integrating visual-inertial global odometry to combine the scale information into our system. We leave these as our future work.

Furthermore, we adopted the camera pose estimation results from the 3D reconstruction as the pseudo ground truth to evaluate

TABLE 3 Quantization error of different SLAM methods under two settings: 1) without and 2) with the proposed GAN-based underwater image enhancement on the URPC dataset.

Method		ATE ↓	RMSE ↓	Initialization ↓
ORB-SLAM2	w/o	1.418	1.484	84
ORB-SLAM2	w/	1.344	1.447	23
Dual-SLAM	w/o	1.438	1.502	49
Dual-SLAM	w/	1.350	1.444	6
ORB-SLAM3	w/o	1.405	1.472	69
ORB-SLAM3	w/	**1.332**	**1.433**	**3**

The best results are in bold.

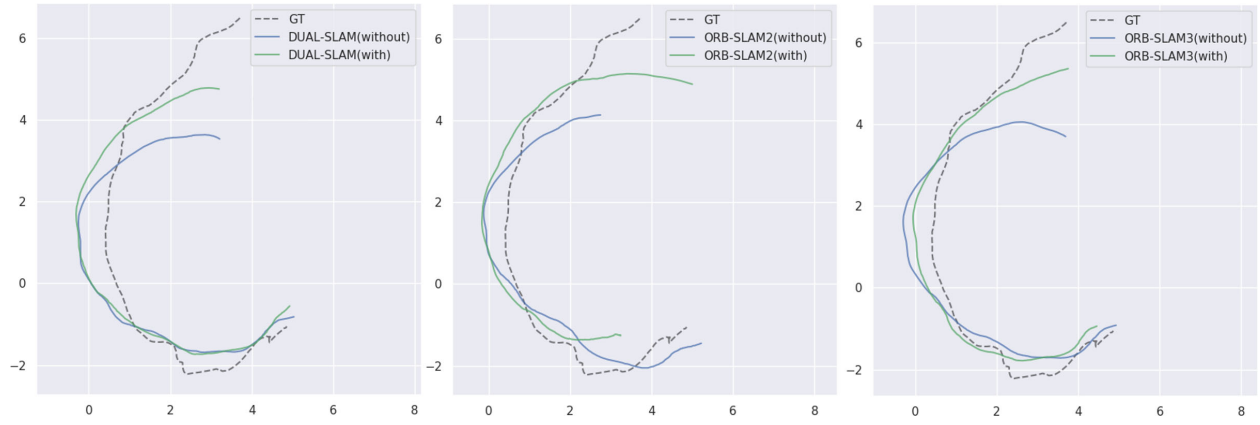


FIGURE 6
The qualitative results of different SLAM methods on URPC dataset under two settings: 1) without and 2) with the proposed GAN-based underwater image enhancement.

TABLE 4 Quantization error of different SLAM methods under two settings: 1) without and 2) with the proposed GAN-based underwater image enhancement on the OUC fisheye dataset.

Method		ATE ↓	RMSE ↓	Initialization ↓
ORB-SLAM2	w/o	2.655	2.700	153
ORB-SLAM2	w/	**2.410**	**2.450**	10
Dual-SLAM	w/o	2.676	2.688	59
Dual-SLAM	w/	2.586	2.520	**1**
ORB-SLAM3	w/o	2.654	2.667	33
ORB-SLAM3	w/	2.559	2.561	2

The best results are in bold.

SLAM performance since it is very challenging and difficult to obtain absolutely accurate ground truth in the underwater setting. To alleviate the ground truth acquisition, we utilize the Structure-from-Motion technique for more robust pose estimation Schönberger and Frahm (2016) in an offline manner since it combines the global bundle adjustment (BA) and pose-graph optimization for more effective and accurate state estimation. The SIFT feature point adopted in Schönberger and Frahm (2016) ¨ 354; Schönberger et al. (2016) is also more accurate than ORB which is widely used in SLAM systems. However, the reconstructed camera poses through 3D reconstruction may still have errors and cannot work under some adverse underwater

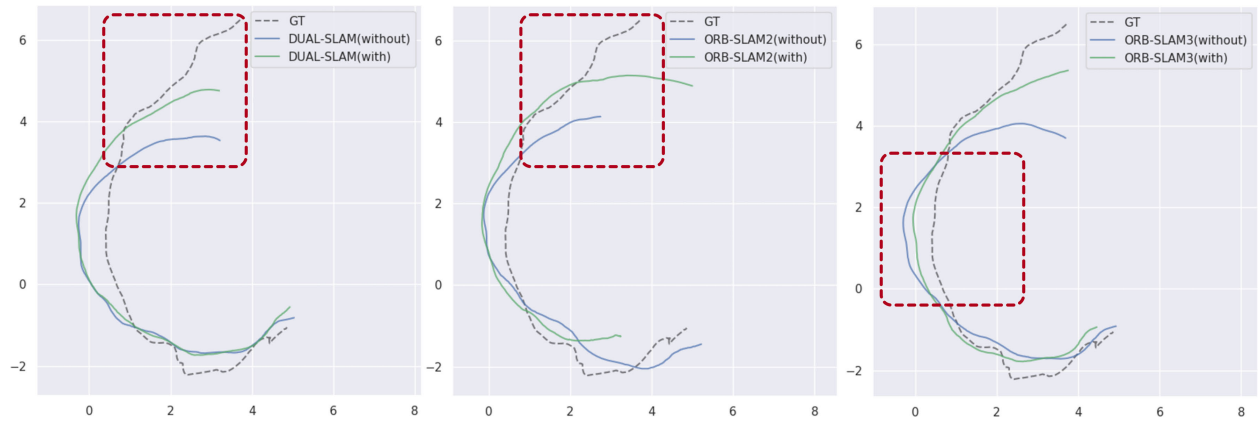


FIGURE 7
The qualitative results of different SLAM methods on the OUC fisheye dataset under two settings: 1) without and 2) with the proposed GAN-based underwater image enhancement.

TABLE 5 The ablation studies of various settings on the URPC dataset.

Settings	FPS ↑	MACs (G) ↓	Para. (M) ↓	ATE ↓	RMSE ↓	Initialization ↓
$c = 8$	**47.54**	**2.26**	**0.321**	1.371	1.466	60
$c = 16$	31.99	8.81	1.28	**1.344**	1.447	**23**
$c = 32$	17.63	34.75	4.99	1.348	**1.446**	24
w/o	–	–	–	1.418	1.484	84

We choose ORB-SLAM2 to evaluate the SLAM performance and the last column "w/o" represents the SLAM performance of the original ORB-SLAM2.
The best results are in bold. "-" means not applicable.

environments (*e.g.*, motion blur, camera shaking, an extensive range of rotation, and *etc.*).

6 Conclusion

This paper has proposed a generic and practical framework to perform robust and accurate underwater SLAM. We have designed a real-time GAN-based image enhancement module through knowledge distillation to promote underwater SLAM performance. With the adaptation of an effective underwater image enhancement as a pre-processing image module, we could synthesize enhanced underwater images with high fidelity for further underwater SLAM, leading to observable performance gains. The proposed framework can work effectively in an extensible way, in which external modifications can plug in the underwater monocular SLAM algorithms.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author.

Author contributions

ZZ and ZX did most of the work and contributed equally in this paper. ZY handled the work of revising the article, and S-KY provided guidance and funding for this research. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the Finance Science and Technology Project of Hainan Province of China under Grant Number ZDKJ202017, the Project of Sanya Yazhou Bay Science and Technology City (Grant No. SCKJ-JYRC-2022-102), the Innovation and Technology Support Programme of the Innovation and Technology Fund (Ref: ITS/200/20FP) and the Marine Conservation Enhancement Fund (MCEF20107).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1161399/full#supplementary-material

References

Aguinaldo, A., Chiang, P.-Y., Gain, A., Patil, A., Pearson, K., and Feizi, S. (2019). Compressing gans using knowledge distillation. doi: 10.48550/arXiv.1902.00159

Akkaynak, D., and Treibitz, T. (2019). "Sea-Thru: a method for removing water from underwater images," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA. (New York City, USA: IEEE), 1682–1691.

Anwar, S., and Li, C. (2020). Diving deeper into underwater image enhancement: a survey. *Signal Process. Image Commun.* 89, 115978. doi: 10.1016/j.image.2020.115978

Asmare, M. H., Asirvadam, V. S., and Hani, A. F. M. (2015). Image enhancement based on contourlet transform. *Signal Image Video Process* 9, 1679–1690. doi: 10.1007/s11760-014-0626-7

Bresson, G., Alsayed, Z., Yu, L., and Glaser, S. (2017). Simultaneous localization and mapping: a survey of current trends in autonomous driving. *IEEE Trans. Intelligent Vehicles* 2, 194–220. doi: 10.1109/TIV.2017.2749181

Buscher, E., Mathews, D. L., Bryce, C., Bryce, K., Joseph, D., and Ban, N. C. (2020). Applying a low cost, mini remotely operated vehicle (rov) to assess an ecological baseline of an indigenous seascape in canada. *Front. Mar. Sci.* 7, 669. doi: 10.3389/fmars.2020.00669

Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., and Tardós, J. D. (2021). Orb-slam3: an accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robotics* 37 (6), 1874–1890. doi: 10.1109/TRO.2021.3075644

Carreras, M., Hernández, J. D., Vidal, E., Palomeras, N., Ribas, D., and Ridao, P. (2018). Sparus ii auv–a hovering vehicle for seabed inspection. *IEEE J. Oceanic Eng.* 43, 344–355. doi: 10.1109/JOE.2018.2792278

Chen, W., Rahmati, M., Sadhu, V., and Pompili, D. (2019). "Real-time image enhancement for vision-based autonomous underwater vehicle navigation in murky waters," in *WUWNet '19: Proceedings of the 14th International Conference on Underwater Networks & Systems*, Atlanta, GA, USA. (USA: ACM Digital Library), 1–8.

Cho, Y., and Kim, A. (2017). "Visibility enhancement for underwater visual slam based on underwater light scattering model," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. (New York City, USA: IEEE), 710–717.

Drews, P., Nascimento, E., Moraes, F., Botelho, S., and Campos, M. (2013). "Transmission estimation in underwater single images," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Sydney, NSW, Australia. (New York City, USA: IEEE), 825–830.

Elvira, R., Tardós, J. D., and Montiel, J. M. (2019). "Orbslam-atlas: a robust and accurate multi-map system," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China. (New York City, USA: IEEE), 6253–6259.

Fabbri, C., Islam, M. J., and Sattar, J. (2018). "Enhancing underwater imagery using generative adversarial networks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia. (New York City, USA: IEEE), 7159–7165.

Ferrera, M., Creuze, V., Moras, J., and Trouvé-Peloux, P. (2019a). Aqualoc: an underwater dataset for visual–inertial–pressure localization. *Int. J. Robotics Res.* 38, 1549–1559. doi: 10.1177/0278364919883346

Ferrera, M., Moras, J., Trouvé-Peloux, P., and Creuze, V. (2019b). Real-time monocular visual odometry for turbid and dynamic underwater environments. *Sensors* 19, 687. doi: 10.3390/s19030687

García, S., López, M. E., Barea, R., Bergasa, L. M., Gómez, A., and Molinos, E. J. (2016). "Indoor slam for micro aerial vehicles control using monocular camera and sensor fusion," in *2016 international conference on autonomous robot systems and competitions (ICARSC)*, Bragança, Portugal. (New York City, USA: IEEE), 205–210.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial networks. *Commun ACM* (USA: ACM Digital Library) 63 (11), 139–144.

Han, R., Guan, Y., Yu, Z., Liu, P., and Zheng, H. (2020). *Underwater image enhancement based on a spiral generative adversarial framework* (IEEE Access), 8, 218838–218852.

He, K., Sun, J., and Tang, X. (2010). Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 2341–2353. doi: 10.1109/TPAMI.2010.168

Hoegh-Guldberg, O., Mumby, P. J., Hooten, A. J., Steneck, R. S., Greenfield, P., Gomez, E., et al. (2007). Coral reefs under rapid climate change and ocean acidification. *Science* 318, 1737–1742. doi: 10.1126/science.1152509

Huang, H., Lin, W.-Y., Liu, S., Zhang, D., and Yeung, S.-K. (2020). "Dual-slam: a framework for robust single camera navigation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA. (New York City, USA: IEEE), 4942–4949.

Huang, Z., Wan, L., Sheng, M., Zou, J., and Song, J. (2019). "An underwater image enhancement method for simultaneous localization and mapping of autonomous underwater vehicle," in *2019 3rd International Conference on Robotics and Automation Sciences (ICRAS)*, Wuhan, China. (New York City, USA: IEEE), 137–142.

Huvenne, V. A., Robert, K., Marsh, L., Iacono, C. L., Le Bas, T., and Wynn, R. B. (2018). Rovs and auvs. *Submarine Geomorphology*, 93–108.

Islam, M. J., Enan, S. S., Luo, P., and Sattar, J. (2020a). "Underwater image super-resolution using deep residual multipliers," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France. (New York City, USA: IEEE), 900–906. doi: 10.1109/ICRA40945.2020.9197213

Islam, M. J., Luo, P., and Sattar, J. (2020b). Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception. doi: 10.15607/RSS.2020.XVI.018

Islam, M. J., Xia, Y., and Sattar, J. (2020c). Fast underwater image enhancement for improved visual perception. *IEEE Robotics Automation Lett.* 5, 3227–3234. doi: 10.1109/LRA.2020.2974710

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). "Image-to-image translation with conditional adversarial networks," in *2017 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. (New York City, USA: IEEE), 1125–1134.

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. doi: 10.48550/arXiv.1412.6980

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., et al. (2017). "Photo-realistic single image super-resolution using a generative adversarial network," in *2017 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. (New York City, USA: IEEE), 4681–4690.

Li, C., Anwar, S., and Porikli, F. (2020a). Underwater scene prior inspired deep underwater image and video enhancement. *Pattern Recognition* 98, 107038. doi: 10.1016/j.patcog.2019.107038

Li, M., Lin, J., Ding, Y., Liu, Z., Zhu, J.-Y., and Han, S. (2020b). "Gan compression: efficient architectures for interactive conditional gans," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (New York City, USA: IEEE) 44 (12), 9331–9346. doi: 10.1109/TPAMI.2021.3126742

Li, N., Zheng, Z., Zhang, S., Yu, Z., Zheng, H., and Zheng, B. (2018). *The synthesis of unpaired underwater images using a multistyle generative adversarial network* (IEEE Access), 54241–54257.

Mirza, M., and Osindero, S. (2014). Conditional generative adversarial nets. doi: 10.48550/arXiv.1411.1784

Mur-Artal, R., and Tardós, J. D. (2017). Orb-slam2: an open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robotics* 33, 1255–1262. doi: 10.1109/TRO.2017.2705103

Qin, T., Li, P., and Shen, S. (2018). Vins-mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robotics* 34, 1004–1020. doi: 10.1109/TRO.2018.2853729

Qin, T., and Shen, S. (2018). "Online temporal calibration for monocular visual-inertial systems," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain. (New York City, USA: IEEE), 3662–3669.

Rahman, S., Li, A. Q., and Rekleitis, I. (2018). "Sonar visual inertial slam of underwater structures," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia. (New York City, USA: IEEE), 5190–5196.

Rahman, S., Li, A. Q., and Rekleitis, I. (2019a). "Contour based reconstruction of underwater structures using sonar, visual, inertial, and depth sensor," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China. (New York City, USA: IEEE), 8054–8059.

Rahman, S., Li, A. Q., and Rekleitis, I. (2019b). "Svin2: an underwater slam system using sonar, visual, inertial, and depth sensor," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China. (New York City, USA: IEEE), 1861–1868.

Reza, A. M. (2004). Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *J. VLSI Signal Process. Syst. signal image video Technol.* 38, 35–44. doi: 10.1023/B:VLSI.0000028532.53893.82

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). "Orb: an efficient alternative to sift or surf," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia. (New York City, USA: IEEE), 2564–2571.

Schönberger, J. L., and Frahm, J.-M. (2016). "Structure-from-motion revisited," in *2016 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA. (New York City, USA: IEEE), 4104–4113.

Schönberger, J. L., Zheng, E., Frahm, J.-M., and Pollefeys, M. (2016). "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on computer vision* (Springer), 501–518.

Zhang, X., Zeng, H., Liu, X., Yu, Z., Zheng, H., and Zheng, B. (2020). *In situ holothurian noncontact counting system: a general framework for holothurian counting* (IEEE Access) 8, 210041–210053.

Zhu, J., Park, T., Isola, P., and Efros, A. A. (2017). "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy. (New York City, USA: IEEE). 2242–2251. doi: 10.1109/ICCV.2017.244

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 11 July 2023
DOI 10.3389/fmars.2023.1132500

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Wenbo Ma,
Xiangtan University, China
Yan Song,
Shandong University, China

*CORRESPONDENCE
Naoki Saito
n.saito@aist.go.jp

RECEIVED 27 December 2022
ACCEPTED 04 May 2023
PUBLISHED 11 July 2023

CITATION
Saito N, Washburn TW, Yano S and Suzuki A (2023) Using deep learning to assess temporal changes of suspended particles in the deep sea.
Front. Mar. Sci. 10:1132500.
doi: 10.3389/fmars.2023.1132500

COPYRIGHT
© 2023 Saito, Washburn, Yano and Suzuki. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Using deep learning to assess temporal changes of suspended particles in the deep sea

Naoki Saito[1,2]*, Travis W. Washburn[1], Shinichiro Yano[3] and Atsushi Suzuki[1,4]

[1]Geological Survey of Japan, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan, [2]Department of Civil Engineering, Kyushu University, Fukuoka, Japan, [3]Department of Urban and Environmental Engineering, Kyushu University, Fukuoka, Japan, [4]Research Laboratory on Environmentally-conscious Developments and Technologies [E-code], National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan

While suspended particles play many important roles in the marine environment, their concentrations are very small in the deep sea, making observation difficult with existing methods: water sampling, optical sensors, and special imaging systems. Methods are needed to fill the lack of environmental baseline data in the deep sea, ones that are inexpensive, quick, and intuitive. In this study we applied object detection using deep learning to evaluate the variability of suspended particle abundance from images taken by a common stationary camera, "Edokko Mark 1". Images were taken in a deep-sea seamount in the Northwest Pacific Ocean for approximately one month. Using the particles in images as training data, an object detection algorithm YOLOv5 was used to construct a suspended particle detection model. The resulting model successfully detected particles in the image with high accuracy (AP50 > 85% and F1 Score > 82%). Similarly high accuracy for a site not used for model training suggests that model detection accuracy was not dependent on one specific shooting condition. During the observation period, the world's first cobalt-rich ferromanganese crusts excavation test was conducted, providing an ideal situation to test this model's ability to measure changes in suspended particle concentrations in the deep sea. The time series showed relatively little variability in particle counts under natural conditions, but there were two turbidity events during/after the excavation, and there was a significant difference in numbers of suspended particles before and after the excavation. These results indicate that this method can be used to examine temporal variations both in small amounts of naturally occurring suspended particles and large abrupt changes such as mining impacts. A notable advantage of this method is that it allows for the possible use of existing imaging data and may be a new option for understanding temporal changes of the deep-sea environment without requiring the time and expense of acquiring new data from the deep sea.

KEYWORDS

suspended particle, monitoring tools, machine learning, object detection, computer vision, YOLO, deep-sea mining, sediment plume

Introduction

Deep-sea environmental functions are influenced by suspended particle concentrations while animals here depend on these particles for survival, making the variability of these particles of geochemical, oceanographic and biological importance. Much of the suspended solids in the ocean exist as aggregate particles of detritus, microorganisms, and clay minerals. Particle concentrations decrease rapidly with depth as organisms feed on and decompose particles in the settling process. Suspended particle concentrations in the open ocean are very low (5-12 µg/L; Brewer et al., 1976; Biscaye and Eittreim, 1977; Gardner et al., 1985) at depths greater than 200 m, and most deep waters have low natural concentrations even near the sea floor (Gardner et al., 2018). These particles are responsible for much of the transport of elements to the deep-sea, are a major energy source for deep-sea biota, and form seafloor sediments (Lal, 1977; Alldredge and Silver, 1988).

Low concentrations make suspended particle abundance in the deep sea difficult to observe. Water sampling can detect minute quantities of suspended particles; however, it cannot be performed frequently due to the difficulty of collecting physical samples in the deep sea. Therefore, changes on fine time scales are difficult to observe with this method. Optical sensors, such as turbidimeters, can take continuous measurements to get better temporal understanding but their accuracy is low when particle concentrations are very low, such as in the deep sea, because the signal is lost in electronic noise due to low scattering intensity (Gardner et al., 1985; Omar and MatJafri, 2009). In fact, previous studies that have used optical sensors to examine suspended particles in the deep sea were focused on nepheloid layers which by definition have elevated concentrations of particles compared to the surrounding environment (Martín et al., 2014; Gardner et al., 2018; Haalboom et al., 2021). Special imaging systems that take pictures of particles or plankton as they pass through a known volume illuminated by a specific light source can both take continuous measurements and provide good accuracy when particle concentrations are very low. *In-situ* imaging systems include Video Plankton Recorder II (VPR) (Davis et al., 2005) and Underwater Vision Profiler 5 (UVP) (Picheral et al., 2010), which are primarily used as profilers. However, these systems are intended for small spatial sampling: the VPR uses approximately 1 – 350 ml of seawater while the UVP captures an approximate area 180 x 180 mm^2 in front of the camera. These systems also require large amounts of money, time, and expertise for installation and analysis. A general problem with deep-sea surveys is that they are difficult to access, expensive, and have limited space for equipment. An observation method that compensates for these shortcomings is needed because little data can be obtained in a single survey (Amon et al., 2022).

This study proposes a method to evaluate variation in suspended particle abundance by applying deep learning-based object detection to images from a common stationary camera. Object detection is a technique related to computer vision that detects the position and number of specific objects in images. In the last decade, accuracy has improved dramatically as deep learning techniques such as convolutional neural networks have been incorporated (Zhao et al., 2019; Zou et al., 2023). In particular, one-stage algorithms which perform object region estimation and classification of each candidate region within a single network, such as YOLO (Redmon et al., 2016), SSD (Liu et al., 2016), RetineNet (Lin et al., 2020), and EfficientDet (Tan et al., 2020), enable fast detection. In the marine field, studies have applied object detection to organisms (Ditria et al., 2020; Salman et al., 2020; Bonofiglio et al., 2022; Kandimalla et al., 2022; Knausgård et al., 2022) and debris (Fulton et al., 2019; Xue et al., 2021), obtaining high detection accuracy (e.g., >80% in F1 Score and Average Precision (AP50) indices). In underwater images, suspended particles scatter light from illumination and appear as circular white reflections. Image processing research often views particles as noise sources and remove them from images (Walther et al., 2004; Cyganek and Gongola, 2018; Wang et al., 2021). On the other hand, when they are targets for object detection, such characteristics may facilitate detection.

Taking advantage of the fact that particles appear in high luminosity, we hypothesized that applying object detection would allow us to evaluate the variation in particle abundance. In this study, fixed-point imaging was conducted for approximately one month on a seamount summit located in the Northwest Pacific Ocean. Using the particles in a subset of images as training data, a particle detection model using the object detection algorithm YOLOv5 was constructed to evaluate the variability in the amounts of suspended particles. During several days of the period, a small-scale excavation test of cobalt-rich ferromanganese crusts (hereafter referred to as "crusts"), which is a potential seafloor mineral resource (Hein, 2004), was also conducted. This activity provided us a test case to assess rapid, large changes in suspended particle abundance in the deep sea. Our proposed approach is intended for use as a simple and auxiliary monitoring tool for exploring temporal variations in the deep-sea environment. There is an increasing need to collect baseline data in the deep sea to assess environmental impacts of ever-expanding human activities there (Ramirez-Llodra et al., 2011; Amon et al., 2022). In particular, deep-sea mining can generate large amounts of resuspended particles, or sediment plumes, which can impact ecosystems (Washburn et al., 2019; Drazen et al., 2020). Understanding the variability of suspended particles in their natural state is essential for environmental impact assessments (Glover and Smith, 2003; Tyler, 2003).

Materials and methods

Study site

The study site was the flat summit of Takuyo-Daigo Seamount located in the northwestern Pacific Ocean (Figure 1A). The Takuyo-Daigo Seamount rises to a depth of approximately 900-1200 m, approximately 4500 m above the 5400 m deep-sea plain. The summit area is approximately 2220 km^2. The basement rocks on the summit are covered with crusts about 10 cm thick, and thin sediments are distributed on top. Most of the sediments are sand composed of planktonic and benthic foraminifera (Hino and Usui,

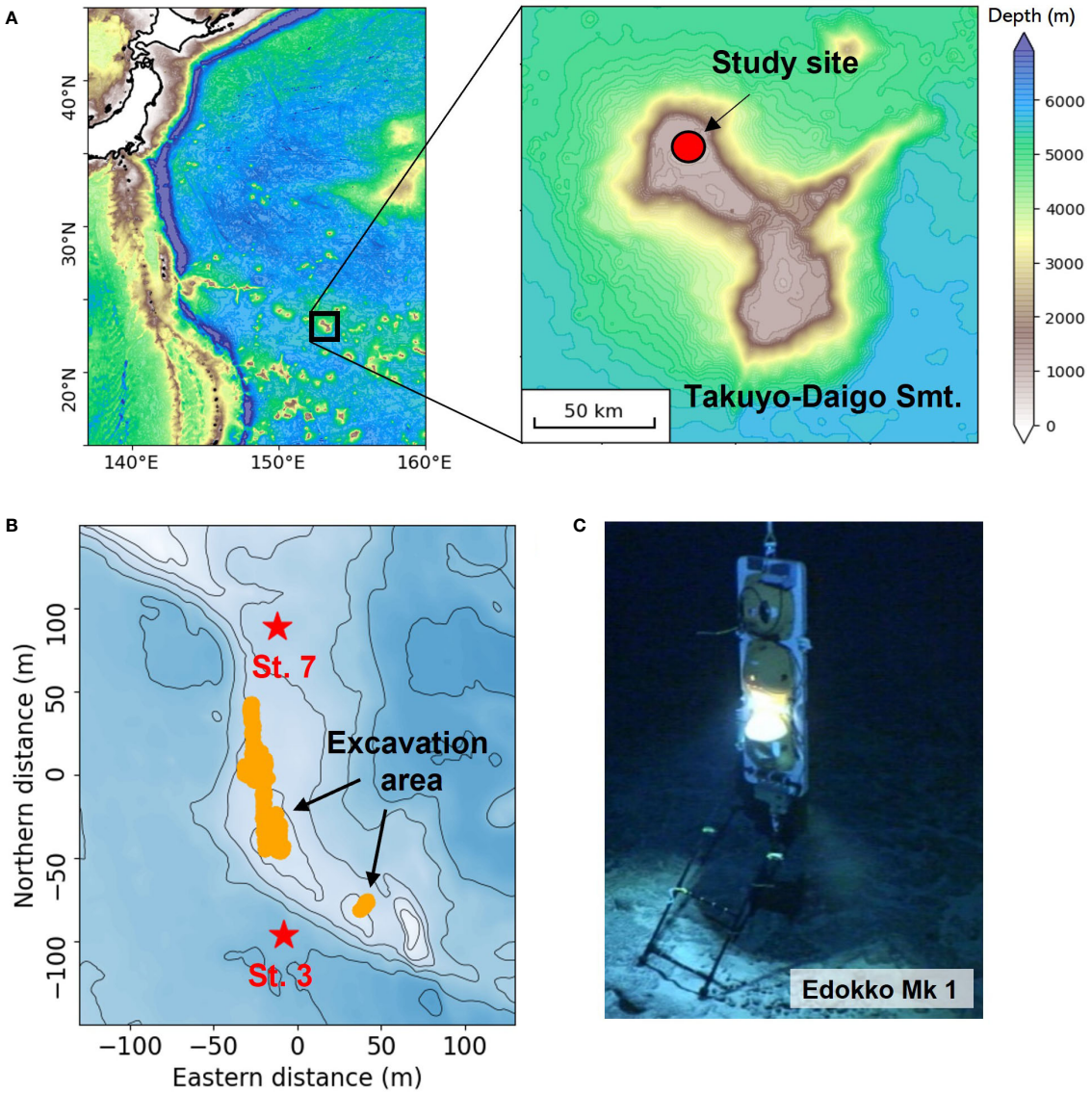


FIGURE 1
Study site **(A, B)** and the deep-sea bottom monitoring device "Edokko Mark 1 HSG type" **(C)**. In **(B)**, the red stars represent sites of image collection, the area in orange represents the location of the excavator operation during the excavation test. For bathymetry of the study site which was at ~950 m, contour lines are for every 2 meters with blue being deeper.

2022; Ota et al., 2022). Suzuki et al. (in review) sampled water in this area and reported a suspended solid concentration of about 20 µg/L.

Image collection

The deep-sea monitoring device "Edokko Mark 1 HSG type" (Okamoto Glass Co., Ltd.) was installed at two locations in the north and south of the study site (St. 3 and St. 7) to capture video (Figures 1B, C). The two locations were selected close (~50 – 100 m) to the excavation area to allow for comparison between sites and represent different levels of sediment deposition. Based on preliminary flow observations and sediment-plume modelling, the plume from the excavation was expected to flow primarily towards St.3 with relatively little towards St.7 (Suzuki et al, in review). The video recording period was from June 23 to July 30, 2020. The shooting time was set to 1 minute every 4 hours from June 23 to July 2 to extend battery life, and 1 minute every hour from July 3 to July 30 for detailed observation. The 2 seconds between when the lights were turned on until the brightness of the lights stabilized was removed from all videos before analysis. The camera was approximately 1.2 m from the bottom, at an angle of approximately 64° to the bottom, and with a horizontal angle of view of approximately 110° (in air). The screen resolution was 1080 p/30 fps. Illumination was approximately 1.6 m above the bottom, at an angle of approximately 30° to the bottom, and at a half illumination angle of ±60° (in air). The total luminous flux was approximately 4000 lumens (in air). An example of the acquired images is shown in Figure 2. Suspended particles were white or translucent and were around ten pixels in size.

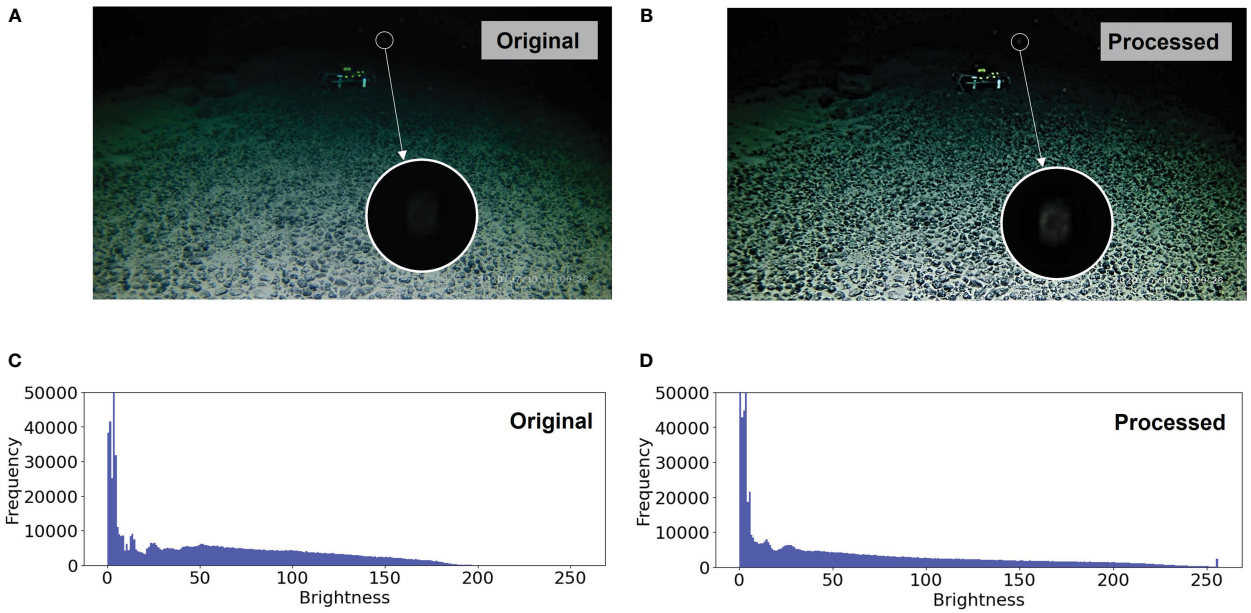


FIGURE 2
Examples of images at St. 3 which are original **(A)** and pre-processed with edge-preserving smoothing filter **(B)**. **(C, D)** are histograms of the HSB color model with pixel brightness (range 0-255) on the horizontal axis, **(C)** for the original image and **(D)** for the processed image. The objects in the upper center of the screen are instruments that are not relevant to this study.

Suspended particles detection

Pre-processing of image

When analyzing underwater images, pre-processing is performed to facilitate the identification of objects. In this study, an edge-preserving smoothing filter was used as a processing method to emphasize suspended particles. In water, light absorption by water and scattering of light by suspended particles and plankton cause image degradation such as color distortion, contrast reduction, and blurring. In previous studies, underwater image preprocessing methods by pixel values correction, physical modeling (Ancuti et al., 2018; Dai et al., 2020; Li et al., 2020; Zhang et al., 2022), and deep learning (Islam et al., 2019; Wang Y. et al., 2019; Anwar and Li, 2020; Li et al., 2020; Jian et al., 2022) were proposed. The goal of these methods is to make the target, such as seafloor or organisms, more visible by restoring color and removing haze. However, suspended particles are considered as noise that should be removed, making existing pre-processing methods for underwater images likely counterproductive in this study. The edge-preserving smoothing filter is a process that preserves the contour lines of the object while smoothing the rest of the image as noise. Therefore, it can be useful in both enhancing the contours of suspended particles and removing blurring. Typical examples include median filter and bilateral filter (Tomasi and Manduchi, 1998; Zhu et al., 2019; Chen et al., 2020). In this study, we used the domain transform filter by Gastal and Oliveira (2011), which is based on a transform that defines an isometry between curves on the 2D image manifold in 5D and the real line. This filter is implemented as a "detail enhancement" function in OpenCV (Intel), a Python library for computer vision, for easy and quick processing. Figure 2 shows the original and processed images and their brightness histograms. The filter processing enhanced the light and dark parts of the images and made the particles sharper.

Model training and validation

An object detection algorithm YOLOv5 (Ultralytics, https://github.com/ultralytics/yolov5) was used to create the suspended particle detection model. YOLOv5 is the fifth generation of You Only Look Once (YOLO) (Redmon et al., 2016), released in June 2020. YOLO performs one-stage object detection using convolutional neural networks. YOLOv5 has four training models (s, m, l, x) with different computational load and detection accuracy. In this study, YOLOv5x, which has the highest computational load and detection accuracy, was selected since the particles targeted have few features and are likely difficult to detect. The training and validation data were images captured every 1 second on July 3, 7, 11, 14, and 20 at St. 3. These days were selected because they contained a relatively large number of particles, with the goal of increasing the number and variation of data. The training data consisted of 1028 images containing a total of 3484 particles, and the validation data consisted of 255 images containing a total of 958 particles. The ratio of training data to validation data was distributed approximately 8:2 for both the number of images and the number of classes. St. 7 was not used as training data, only for accuracy verification using the validation data. This allows us to examine whether the detection model works accurately when the location (background of the image) is changed. As with St. 3, the validation data for St. 7 consisted of images captured on July 3, 7, 11, 14, and 20. There was a total of 255 images, containing 575 particles. The hyperparameters were the default settings of YOLOv5. The number of epochs, indicating the number of training iterations, was set to 100, and the batch size was set to 4. The input image size was 1280 × 720

pixels. The loss function was the bounding box regression loss with mean squared error. The loss function is a measure of the magnitude of the discrepancy between the correct value (validation data) and the predicted value (detection result), which is used to optimize the model.

The detection accuracy of the model was evaluated based on intersection over union (IOU), a measure of the overlap of the area of the rectangles of the annotations of the correct and predicted values. Assuming that the validation data are ground truth, the rectangle of the validation data is R_v, and the rectangle of the detection results is R_d, IOU is defined as follows.

$$IOU = \frac{area(R_d \cap R_v)}{area(R_d \cup R_v)}$$

The IOU was compared to the threshold value t. When IOU ≥ t, the detection result was considered correct. In this study, the commonly used value t = 50% was used.

Precision (P), which indicates the percentage of detected rectangles that are correct, and recall (R), which indicates the percentage of detected rectangles that should be detected, are defined as follows.

$$P = \frac{True\ \ Positive}{True\ \ Positive + False\ \ Positive}$$

$$R = \frac{True\ \ Positive}{True\ \ Positive + False\ \ Negative}$$

Then, the average precision (AP), a measure of the model's detection accuracy, is defined as follows.

$$AP = \int_0^1 P(R)dR$$

In this study, AP50, which means the threshold for IOUs is 50%, was used. AP50 is one of the most common performance indicators for object detection accuracy (Padilla et al., 2020). In addition, the F1 Score, an index that shows the balance between precision and recall, was used to confirm model's performance:

$$F1 = \frac{2}{P^{-1} + R^{-1}}$$

For both AP50 and F1 Score, the closer to 100 on the percentage scale, the better the model's accuracy.

Suspended particles detection

Particle detection was performed on captured images at 5-second intervals for each video. Only the upper 40% of the image was used to assess the temporal changes of the particles. The upper 40% of the viewing area was chosen because this was the portion of the image that did not overlap with the seafloor, and similarities in properties between the seafloor and suspended particles hindered detection. The complexity of the seafloor also appeared to cause some areas of false positives in particle detection at the location not used to train the model (see "Results" chapter for details). The average number of particles for each video (particle numbers counted every 5 seconds averaged over 1 minute) was defined as N40, and was used to evaluate time-series changes. N40 was square-root transformed before statistical analysis (2-way ANOVA and Tukey's HSD test). To focus on rapid increases in suspended particles (described below), we defined a "turbidity event" as a period when N40 was observed to be more than 10x the pre-excavation average. The time required to detect a single image was about 1.5 seconds when a CPU (Intel Core i9-10850K, 3.6 GHz) was used, which was roughly the same whether there were zero or more than 200 particles.

Excavation test

During image collection, the world's first small-scale excavation test of crusts was conducted (Japan Oil, Gas and Metals National Corporation, 2020). The test period was July 9-16, 2020, and a total of seven dredging excavations were conducted. The total excavation distance was 129 m, the excavation width was 0.5 m, and the total dredging time was 109 minutes (Figure 1B). The excavation area was located on top of a 5-7 m high hill, surrounded by a seafloor at a depth of ~950 m. The excavator moved along the seafloor with a crawler, excavated the crusts with a cutterhead, and collected the excavated material by a dredge hose to supplement the cyclone tank. For further details please see Suzuki et al. (in review).

Result

Detection accuracy

The highest AP50 in the learning process was 85.8%, which occurred at 96 epochs (Figure 3A; Table 1). Therefore, the model trained up to 96 epochs was used in this study. The loss function trend (Figure 3B) showed that the error decreased as the model was trained, and no overlearning occurred. The values converged after approximately 30 epochs, indicating that the number of training iterations was sufficient. For St. 7, which was not used to train the model, the validation results showed an accuracy of AP50 = 87.9% (Table 1). The F1 Scores were >80% for both St. 3 (82.1%) and St. 7 (86.1%) (Table 1).

Examples of model detection results are shown in Figures 4, 5. The sizes of the particles detected ranged from approximately 5 to 20 pixels (Figure S1). Particles were mainly detected in the upper 40% of image where the background was blackish water; in St. 3, the percentage of particles located in the upper 40% was 99%, and in St. 7, it was 97% (Figure 6). On the lower 60% of the image field, where whitish sandy seafloor was the primary background, similar whitish particles were difficult to identify and were rarely detected. Suspended particles that appeared blurred and elliptical due to the fast flow were not detected. The reason for these non-detections was that particles with indistinct contours were not included in the training data in order to avoid false positives for the seafloor and organisms. In the lower part of St. 7, there were two areas of false positives, which corresponded to whitish sediment patches (Figures 6B, C). Other factors that could contribute to false detections include the appearance of organisms such as shrimp and fish, or the slight swaying of the camera system itself due to the

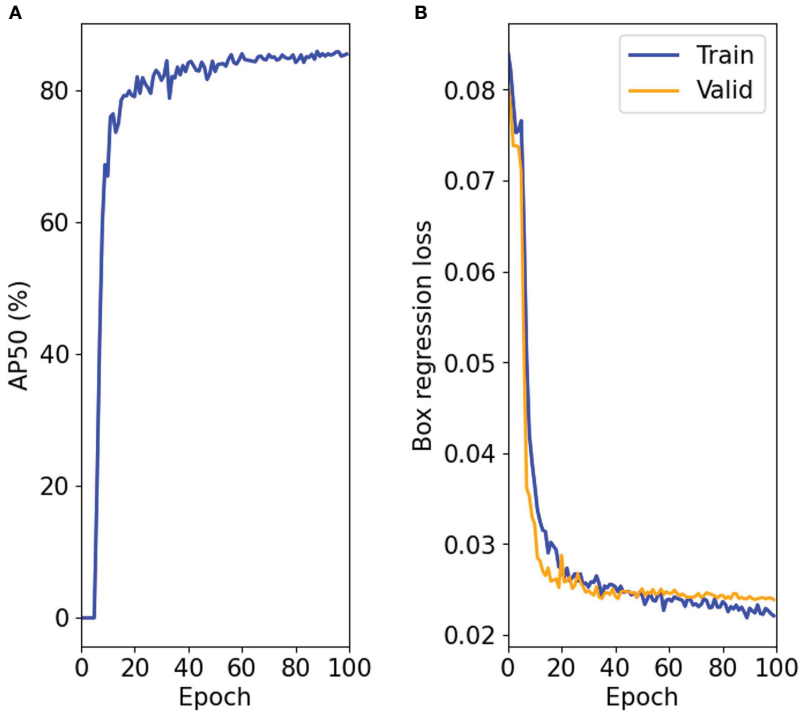


FIGURE 3
Model training transition. **(A)** average precision (AP) with 50% thresholds for correct detection, AP50, and **(B)** box regression loss. The blue line in **(B)** shows transition of training while the orange line shows transition of validation.

current, but manual visual inspection of the images confirmed that these were not an issue with our dataset (Figure S2).

Fluctuation in suspended particle abundance

The 2-way ANOVA found statistically significant differences for the number of particles detected in the upper 40% of images, N40, between St. 3 and St. 7 ($F_{1, 1390} = 106.44$, $p < 0.001$) and among times (i.e., before, during and after the excavation) ($F_{2, 1390} = 7.51$, $p < 0.001$), while the interaction term for station and time was not significant ($F_{2, 1390} = 0.01$, $p = 0.988$). Spectral analysis including the entire duration of the study revealed no tidal (diurnal or half-diurnal) variation in the time series of N40 (Figure S4).

Natural conditions

Under natural conditions (before the excavation test), N40 had mean values of 3.6 and 2.3 with maximum values of 18.5 and 15.8 for St. 3 and 7, respectively (Table 2, Figure S3). There was a significant difference between St. 3 and St. 7 (Tukey's HSD test, $p < 0.001$). Standard deviation was half of the mean for each station before excavation (Table 2).

Conditions during and after the excavation test

During the excavation test N40 had mean values of 4.7 and 2.3 with maximum values of 248.0 and 4.0 for St. 3 and 7, respectively. After the test, N40 had mean values of 4.8 and 2.9 with maximum values of 88.7 and 46.7 for St. 3 and 7, respectively (Table 2, Figure S3). There was a significant difference between St. 3 and St. 7 both during ($p < 0.001$) and after ($p < 0.001$) the excavation. During excavation standard deviation was ~4 times the mean for St. 3, but only 22% of the mean for St. 7. After excavation standard deviation was roughly the mean at both stations (Table 2).

At St. 3, there was no significant difference between before, during, and after excavation ($p > 0.1$); however, at St. 7, the number of particles after excavation was significantly larger than the number of particles both before ($p < 0.01$) and during ($p < 0.01$) excavation.

TABLE 1 Accuracy validation of the detection model.

	Model training	Precision (%)	Recall (%)	AP50 (%)	F1 (%)
St. 3	Used	85.6	78.8	85.8	82.1
St. 7	Unused	87.4	84.8	87.9	86.1

The results for St. 3 and St. 7 are described. "Model raining" means whether the image of the stations was used for model training. AP50 means average precision (AP) with 50% thresholds for correct detection, and F1 means F1 Score. The closer to 100 for both AP50 and F1 Score, the better the model's performance.

10.3389/fmars.2023.1132500

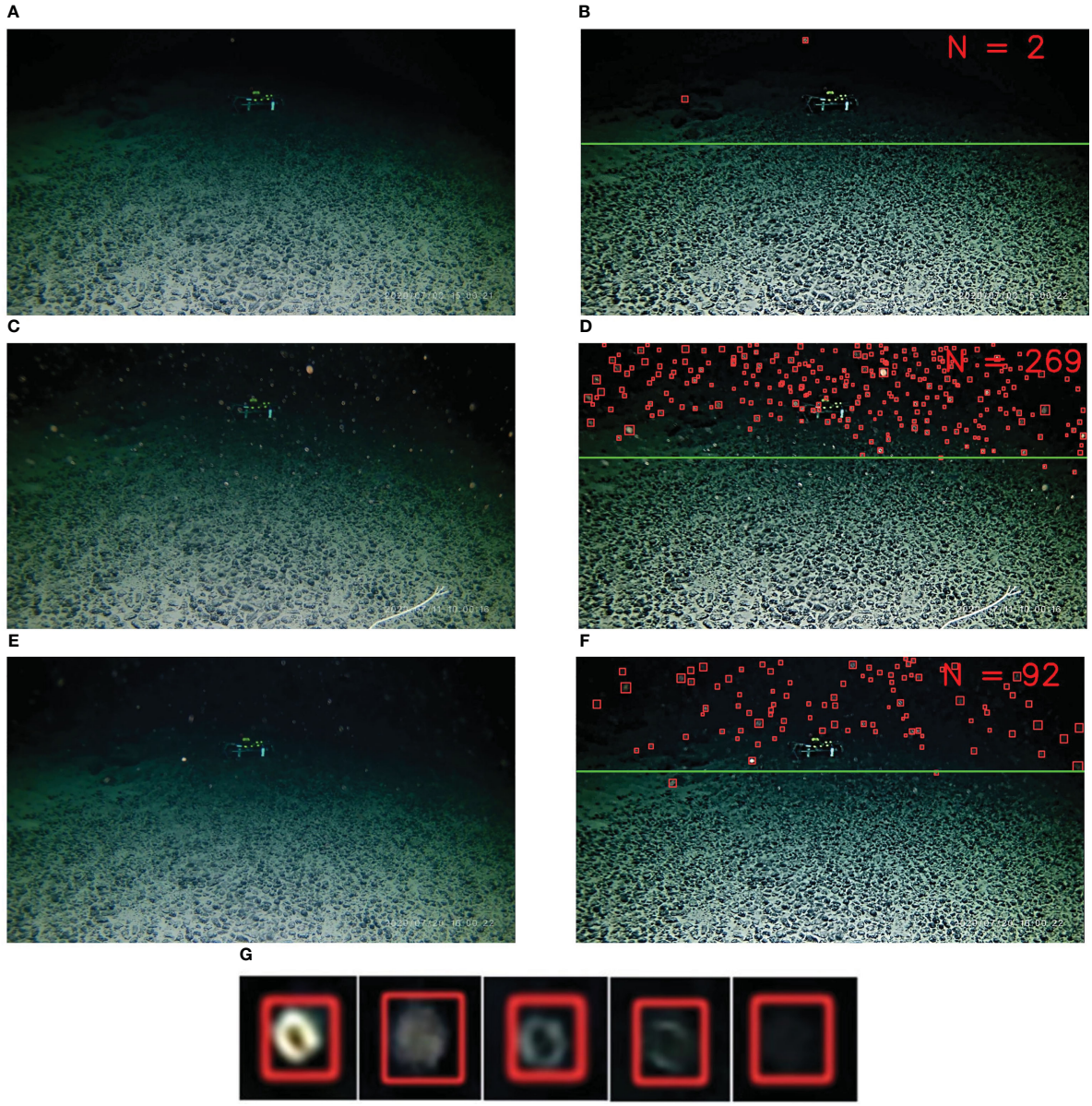


FIGURE 4
Examples of original images (left column) and particle detection results (right column) in St. 3. Detected particles are surrounded by red rectangles. The number in the upper right corner of the detected image represents the number of particles. The green lines crossing the images show the upper 40%. The images in the right column were pre-processed to enhance light and dark areas. Images were taken at **(A, B)** 15:00 on July 3 (before excavation test), **(C, D)** 10:00 on July 11 (turbidity event during excavation test), and **(E, F)** 16:00 on July 20 (turbidity event after excavation test). **(G)** examples of detected particles. The objects in the upper center of the screen are instruments that are not relevant to this study.

Turbidity events

The N40 showed 3 turbidity events during the observation period, two at St. 3 and one at St. 7, which all occurred either during or after the excavation (Figure 7). For St. 3, the first event was on July 11 at 10:00 during excavation (maximum N40 = 248.0) and was observed at only this time. The second event occurred four days after the end of the excavation test on July 20 and was observed from 13:00 to 20:00 (maximum N40 = 88.7). For St. 7, the turbidity event occurred on July 20 and was observed from 14:00 to 19:00 (maximum N40 = 46.7). For both St. 3 and St. 7, the maximum N40 after the excavation test occurred at 16:00 on July 20 (Figure 7).

Discussion

The results of this study suggest that object detection with deep learning may serve as a valuable tool for assessing suspended particle abundance in the deep sea using image datasets. The detection model could detect particles in images with high accuracy at locations used for both model training and those not used (Table 1; Figures 4, 5). The model enabled us to assess temporal changes of particles, including natural small-scale variability and rapid increases possibly caused by anthropogenic disturbance (i.e., small-scale crusts excavation test) (Figure 7).

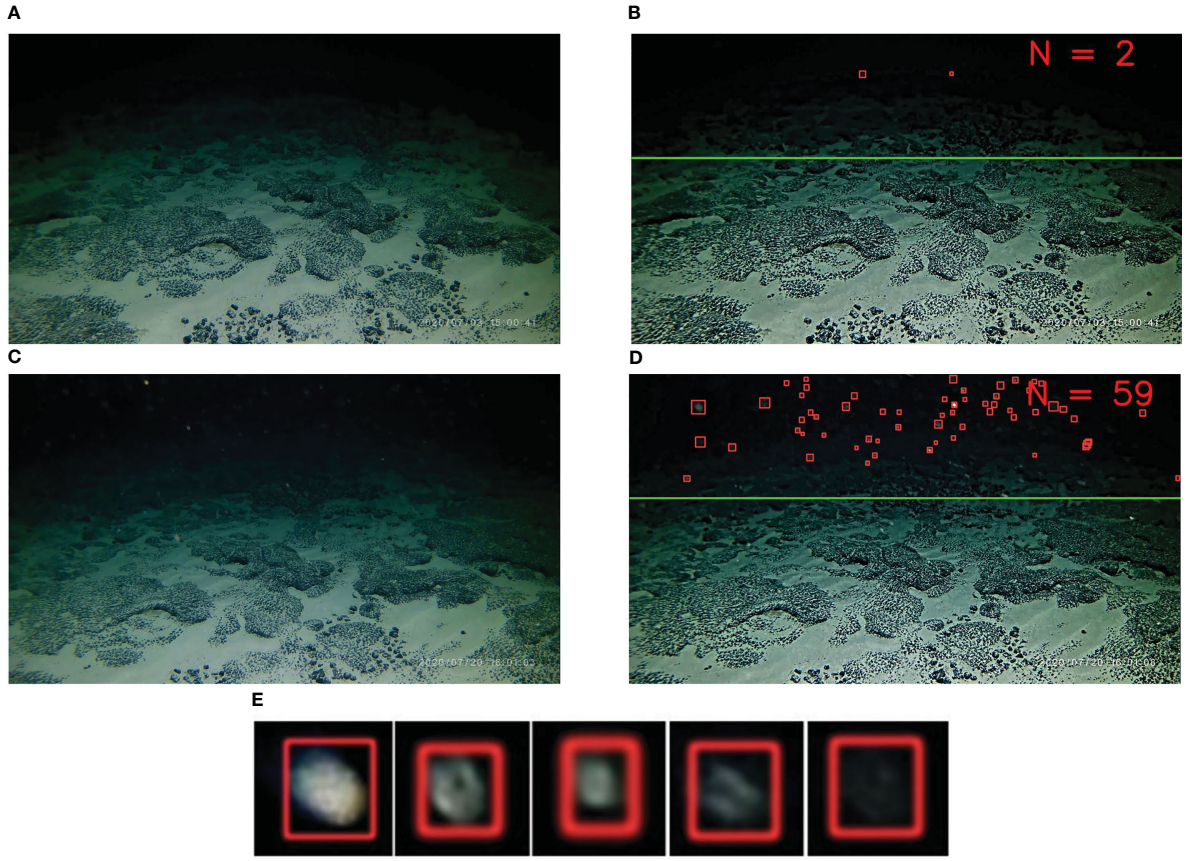


FIGURE 5
Examples of original images (left column) and suspended particle detection results (right column) at St. 7. Images were taken at **(A, B)** 15:00 on July 3 (before excavation test) and **(C, D)** 16:00 on July 20 (turbidity event after excavation test). **(E)** examples of detected particles.

The detection model's wide measurement range combined with the ease of eliminating artifacts and possibility of examining both short and long time scales suggest that our method for examining deep-sea suspended particle concentrations can compensate for many shortcomings of existing methods. The detection model was able to measure from zero to hundreds of particles in an image, which may help overcome the detection limits of optical sensors (Gardner et al., 1985; Omar and MatJafri, 2009). To measure low concentrations by optical sensors, it is useful to narrow the measurement range to a higher sensitivity. However, Baeye et al. (2022) measured seafloor disturbance tests with turbidimeters and found that low range turbidimeters are often saturated. Also, measuring low turbidity with optical sensors can often produce electronic noise (Omar and MatJafri, 2009). A detection model that can easily visually identify whether noise is artificial or not (see Figure 6) may be useful as a reference for optical sensors. The fine time scale measurements of the detection model can also complement the sparseness of the measurements generally associated with water sampling. In our study, the measurement interval was 1 – 4 hours, but it can be further fine-tuned according to the interval of image capture. Because detection models can cover a large area, they may be better suited as a monitoring tool than specialized camera systems which generally examine trace amounts of seawater, such as VPR (Davis et al., 2005) or UVP (Picheral et al., 2010). Since one of the objectives of special camera systems is to observe the morphology of plankton and particles, there is a tradeoff between the delicacy of image quality and the narrowness of the measurement space (Lombard et al., 2019). The basic principle of the method in this study is the same as that of the special camera system in the sense that it measures particles in the image. However, the general stationary camera used in this study captured reflected light over a wider area, allowing it to measure sparsely distributed particles, as shown in Figures 4B, 5B. As a bonus, general stationary cameras are much cheaper and user-friendly than specialized camera systems and are commonly used in various deep-sea studies.

Our study is the first that we know of to attempt to use deep learning to quantify suspended particle abundance. While other computational methods exist besides deep learning which may serve useful in quantifying suspended particles, such as binary processing and motion detection, these methods have inherent characteristics that may lead to false measurements. Binary processing, which separates images into background and target objects, may be able to measure particles that stand out against a black background, but if objects other than particles, such as organisms, are captured in the image, they too will be separated from the background and subject to measurement. Motion detection, which detects moving objects against a fixed background, may also be an option for observation of flowing

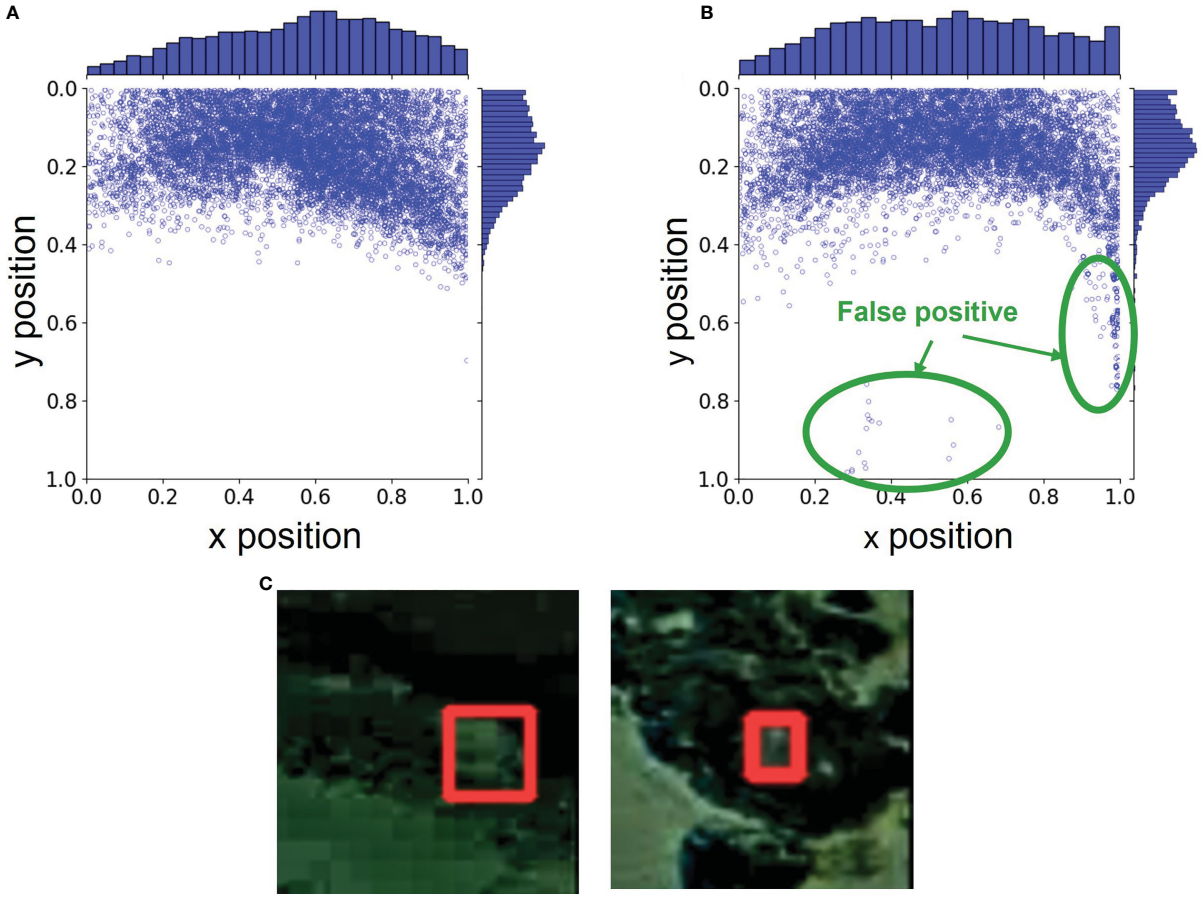


FIGURE 6
Positions of the detected particles. The vertical and horizontal axes represent the x and y coordinates on the image, normalized from 0 to 1, respectively. **(A)** St. 3 and **(B)** St. 7. **(C)** example of an area of false positives caused by whitish sediment in St. 7.

particles (Neri et al., 1998); however, in our study, the video (images) included mobile shrimp and fish while motion was also created by the slight swaying of the camera system itself caused by the current. The use of motion detection would also prevent the use of the vast amounts of video data collected during ROV dives. In general, using deep learning to train a system with target examples is much easier than manually programming the process to predict and avoid all possible false positive targets as described above (Jordan and Mitchell, 2015), greatly reducing the need for manual visual confirmation and additional processing. One remaining challenge is that false positives occurred in certain areas of the seafloor at the station not used for model training (Figure 6), but this can be addressed by increasing the diversity of the dataset used for training (e.g., variations in the environment and shooting conditions).

Our model results suggest that similar evaluations using this method can be made for image data from various locations and also areas where no trained data are used. Most of the particles detected were in the portion of the image where the background was blackish

TABLE 2 The values for the number of particles detected in the upper 40% of images, N40, for the entire observation period divided into before, during, and after the excavation.

	Excavation test	Count	Mean	Std	Mdn	Min	Max
St. 3	Before	203	3.6	1.7	3.1	0.0	18.5
	During	168	4.7	18.9	3.0	0.0	248.0
	After	327	4.8	6.6	4.0	0.0	88.7
St. 7	Before	203	2.3	1.1	2.3	0.0	15.8
	During	168	2.3	0.5	2.3	0.0	4.0
	After	327	2.9	2.9	2.4	0.0	46.7

Count represents the number of 60-second observations (i.e., samples), Std represents standard division, Mdn represents the median, Min represents the minimum value, and Max represents the maximum value.

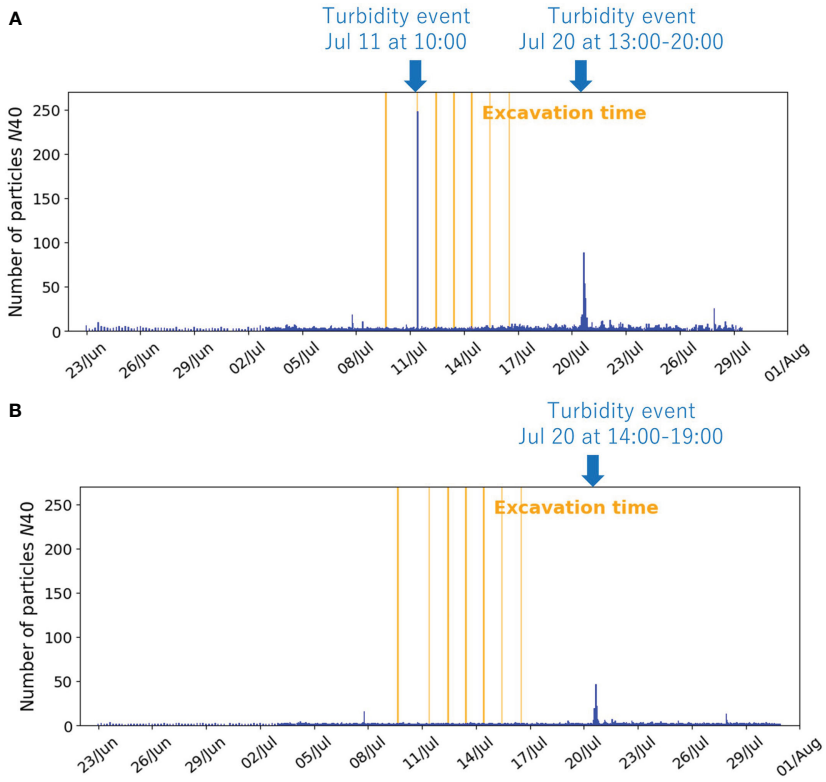


FIGURE 7
Temporal changes of the number of detected particles in the upper 40% of images, N40 for **(A)** St. 3 and **(B)** St. 7. The orange vertical lines indicate the times of excavation.

water, and by extracting only the detection results from that part, the possibilities of false positives were greatly reduced. In deep water, where no sunlight reaches, the background is always black water at any location unless the seafloor is captured, and turbidity is generally low, so the environmental conditions affecting the images are fairly similar regardless of specific habitat. Therefore, the model may be similarly accurate for any deep-sea image data set. However, it should be noted that the image dataset used in this study is for only two sites, and could be insufficient in terms of quantity and diversity. It is still necessary to test the model's performance using data sets with a greater variety of shooting and environmental conditions. Much of the work on underwater object detection has been done on fish (Ditria et al., 2020; Salman et al., 2020; Bonofiglio et al., 2022; Kandimalla et al., 2022; Knausgård et al., 2022), which, although they look and behave differently from suspended particles, could be a useful reference for dataset collection. Ditria et al. (2020), which targeted one type of fish for detection, tested the model's performance accuracy on images from the same estuarine region as the training data and on images from a different estuarine region, and found similarly high accuracy (> 92% for F1 Score and AP50). Salman et al. (2020), which proposed a method to detect moving fish, demonstrated that the approach is robust to image variability using a large underwater video repository containing diverse environments and fish species (> 80% for F1 Score).

Future work required to improve our particle-detection method includes extending the diversity of image datasets used for accuracy validation and identifying the limits of applicability of the model. Examples of future datasets to explore include images from habitats with a wide range of environmental conditions including particle size, suspended particle concentration, and flow velocity (how fast flowing blurry particles can be detected). In terms of imaging conditions, particular attention may need to be paid to lighting, which affects the visibility of suspended particles (Walther et al., 2004; Cyganek and Gongola, 2018). The detection results also need to be calibrated with physical collections of suspended particles to convert what is essentially qualitative data into actual quantitative data. Otherwise, they cannot be compared with observations from other studies (e.g., Biscaye and Eittreim, 1977; Gardner et al., 2018). Laboratory dilution methods that convert turbidimeter readings (formazin turbidity units, FTU) to concentrations (mg/L) may be a reference for calibration. For example, Spearman et al. (2020) diluted sediment samples with seawater from the field to create suspensions of known concentrations. Optical sensors were then immersed in these suspensions and their FTU readings were recorded, and this process was repeated over a range of concentrations. For future work, a similar calibration may be possible by replacing the optical sensor with a camera and using a water tank. Furthermore, even if abrupt changes due to anthropogenic impacts are measured, it is still remains largely unknown what thresholds of suspended particles will be ecologically relevant (Washburn et al., 2019; Drazen et al., 2020), although this work is not directly related specifically to our methods.

Our model may provide new insights into temporal changes of suspended particles. The extremely low N40 values before the excavation highlight the difficulties of measurement by previous methods. But the fact that these particles constitute the primary food source of organisms in the deep sea (Lal, 1977; Alldredge and Silver, 1988) suggest that changes in observed particles from, for example, N40 = 1 to N40 = 10 would constitute a possible 900% increase in food supply. Thus, even "small" temporal variability may be of large importance in the deep sea, and our detection model may be able detect these miniscule changes.

The observations following the excavation test also have interesting implications on future impacts of deep-sea mining. The cause of differences in average N40 among time periods and the rapid increases of particles, or turbidity events, may be a sediment plume of broken crust particles, a large amount of resuspended sediment generated by disturbance, or resuspension of natural sediment or sediment deposited from the plume after excavation (Sharma et al., 2001; Aleynik et al., 2017). The fact that for N40 at St. 7, there was no difference before and during the excavation test, but there were differences before and after and during and after may suggest that once deposited, the particles from excavation increased the amount of suspended particles in the surrounding area over time due to resuspension (Sharma et al., 2001; Aleynik et al., 2017). However, human disturbance is often associated with increased variability, and the extremely large standard deviation during the excavation at St. 3 compared to other times suggests that there may have been alterations in suspended particle concentrations during the test as well (Table 2). Much remains unexplored about dynamics of sediment plumes (Washburn et al., 2019; Drazen et al., 2020) and resuspension in deep-sea seamounts (Turnewitsch et al., 2013). These likely causes are not discussed in detail because they are beyond the scope of this paper which is focused on methodology. For further details please see Suzuki et al. (in review). If turbidity events were indeed caused by the excavation test, one would expect there to be plumes generated during each of the 7 excavations. A likely reason why only one event was observed during excavation is that the excavation time was too short to be captured by the one-minute-per-hour video recording. Due to the limitations in our dataset, we chose to use the excavation test as an example of high particle concentrations for our model rather than attempt to focus on and define the extent of impacts from excavation itself. This highlights the importance of carefully considering sampling intervals to ensure the ability to examine particular hypotheses.

A notable advantage of our detection model is that it can be adapted to observational data acquired for other purposes, even opening up the possibility of providing new insights from the thousands of hours of data collected in the past. The detection model is likely to be applicable to any deep-sea region and camera system, as long as the entire image does not show the seafloor. Monitoring deep-sea environments with imagery is a common research topic (Bicknell et al., 2016); therefore, there is already an abundance of image data to which the detection model could potentially be applied. A fundamental challenge for ocean observations is to reduce costs (Wang Z. A. et al., 2019). This challenge is particularly acute in deep-sea surveys where access to the field is difficult (Amon et al., 2022). Leveraging existing imaging data may reduce the need for new surveys and the need for familiarization and installation of specialized equipment, and may allow for rapid data collection at a lower cost. Detection models can be a new option to make better use of existing data and improve our understanding of suspended particles in the deep sea.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author.

Author contributions

Conceptualization: NS. Methodology: NS and TW. Writing—original draft preparation: NS and TW. Writing—review and check: SY and AS. Image collection: AS. Training and validation of object detection model: NS. Statistical analysis: NS and TW. All authors have read and agreed to the published version of the manuscript. All authors contributed to the article and approved the submitted version.

Acknowledgments

This project was commissioned by the Agency for Natural Resources and Energy in the Japanese Ministry of Economy, Trade and Industry and the Japan Organization for Metals and Energy Security (JOGMEC). The authors express their appreciation to Yoshiaki Igarashi and Jumpei Minatoya (JOGMEC), and Kazumasa Ikeda (Okamoto Glass Co., Ltd.), and others involved in this project. This study was also supported by Research Laboratory on Environmentally-conscious Developments and Technologies (E-code) at AIST.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or

claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1132500/full#supplementary-material

SUPPLEMENTARY FIGURE 1
Size of the detected particles. The vertical and horizontal axes indicate the size in pixels along the x- and y-axes, respectively. Note that many points are plotted overlapping each other.

SUPPLEMENTARY FIGURE 2
Example of images showing possible false positive targets. Shrimp, fish, and a rope used to secure the camera system were captured. **(A, B)** are from St. 3 and **(C, D)** are from St. 7. The number in the upper right corner of the images represents the number of particles detected by the model.

SUPPLEMENTARY FIGURE 3
Box-and-whisker plots of suspended particle counts detected in the upper 40% of images (N40). Plotted separately before, during, and after excavation test at St. 3 and St. 7.

SUPPLEMENTARY FIGURE 4
Results of spectral analysis on the number of suspended particles detected in the upper 40% of images (N40). **(A)** St. 3 and **(B)** St. 7. The data used were taken from July 3, 2020, when the image taking interval was 1 hour.

References

Aleynik, D., Inall, M. E., Dale, A., and Vink, A. (2017). Impact of remotely generated eddies on plume dispersion at abyssal mining sites in the pacific. *Sci. Rep.* 7, 16959. doi: 10.1038/s41598-017-16912-2

Alldredge, A. L., and Silver, M. W. (1988). Characteristics, dynamics and significance of marine snow. *Prog. Oceanogr.* 20 (1), 41–82. doi: 10.1016/0079-6611(88)90053-5

Amon, D. J., Gollner, S., Morato, T., Smith, C. R., Chen, C., Christiansen, S., et al. (2022). Assessment of scientific gaps related to the effective environmental management of deep-seabed mining. *Mar. Policy* 138, 105006. doi: 10.1016/j.marpol.2022.105006

Ancuti, C. O., Ancuti, C., De Vleeschouwer, C., and Bekaert, P. (2018). Color balance and fusion for underwater image enhancement. *IEEE Trans. Image Process.* 27 (1), 379–393. doi: 10.1109/TIP.2017.2759252

Anwar, S., and Li, C. (2020). Diving deeper into underwater image enhancement: a survey. *Signal Process. Image Commun.* 89, 115978. doi: 10.1016/j.image.2020.115978

Baeye, M., Purkiani, K., Stigter, H., Gillard, B., Fettweis, M., and Greinert, J. (2022). Tidally driven dispersion of a deep-Sea sediment plume originating from seafloor disturbance in the DISCOL area (SE-pacific ocean). *Geosci* 12 (1), 8. doi: 10.3390/geosciences12010008

Bicknell, A. W. J., Godley, B. J., Sheehan, E. V., Votier, S. C. V., and Witt, M. J. (2016). Camera technology for monitoring marine biodiversity and human impact. *Front. Ecol. Environ.* 14 (8), 424–432. doi: 10.1002/fee.1322

Biscaye, P. E., and Eittreim, S. L. (1977). Suspended particulate loads and transports in the nepheloid layer of the abyssal Atlantic ocean. *Mar. Geol.* 23 (1-2), 155–172. doi: 10.1016/0025-3227(77)90087-1

Bonofiglio, F., De Leo, F. C., Yee, C., Chatzievangelou, D., Aguzzi, J., and Marini, S. (2022). Machine learning applied to big data from marine cabled observatories: a case study of sablefish monitoring in the NE pacific. *Front. Mar. Sci.* 9. doi: 10.3389/fmars.2022.842946

Brewer, P. G., Spencer, D. W., Biscaye, P. E., Hanley, A., Sachs, P. L., Smith, C. L., et al. (1976). The distribution of particulate matter in the Atlantic ocean. *Earth Planet. Sci. Lett.* 32 (2), 393–402. doi: 10.1016/0012-821X(76)90080-7

Chen, B., Tseng, Y., and Yin, J. (2020). Gaussian-Adaptive bilateral filter. *IEEE Signal Process. Lett.* 27, 1670–1674. doi: 10.1109/LSP.2020.3024990

Cyganek, B., and Gongola, K. (2018). Real-time marine snow noise removal from underwater video sequences. *J. Electron. Imaging* 27 (4), 43002. doi: 10.1117/1.JEI.27.4.043002

Dai, C., Lin, M., Wu, X., Wang, Z., and Guan, Z. (2020). Single underwater image restoration by decomposing curves of attenuating color. *Opt. Laser Technol.* 123, 105947. doi: 10.1016/j.optlastec.2019.105947

Davis, C. S., Thwaites, F. T., Gallager, S. M., and Hu, Q. (2005). A three-axis fast-tow digital video plankton recorder for rapid surveys of plankton taxa and hydrography. *Limnol. Oceanogr.: Methods* 3 (2), 59–74. doi: 10.4319/lom.2005.3.59

Ditria, E. M., Lopez-Marcano, S., Sievers, M., Jinks, E. L., Brown, C. J., and Connolly, R. M. (2020). Automating the analysis of fish abundance using object detection: optimizing animal ecology with deep learning. *Front. Mar. Sci.* 7. doi: 10.3389/fmars.2020.00429

Drazen, J. C., Smith, C. R., Gjerde, K. M., Haddock, S. H. D., Carter, G. S., Choy, C. A., et al. (2020). Midwater ecosystems must be considered when evaluating environmental risks of deep-sea mining. *PNAS* 117 (30), 17455–17460. doi: 10.1073/pnas.2011914117

Fulton, M., Hong, J., Islam, M. J., and Sattar, J. (2019). "Robotic detection of marine litter using deep visual detection models," in *2019 International Conference on Robotics and Automation (ICRA)*. (Montreal, QC, Canada: IEEE), 5752–5758. doi: 10.1109/ICRA.2019.8793975

Gardner, W. D., Biscaye, P. E., Zaneveld, J. R. V., and Richardson, M. J. (1985). Calibration and comparison of the LDGO nephelometer and the OSU transmissometer on the Nova scotian rise. *Mar. Geol.* 66 (1-4), 323–344. doi: 10.1016/0025-3227(85)90037-4

Gardner, W. D., Richardson, M. J., Mishonov, A. V., and Biscaye, P. E. (2018). Global comparison of benthic nepheloid layers based on 52 years of nephelometer and transmissometer measurements. *Prog. Oceanogr.* 168, 100–111. doi: 10.1016/j.pocean.2018.09.008

Gastal, E. S. L., and Oliveira, M. M. (2011). Domain transform for edge-aware image and video processing. *ACM Trans. Graph* 30 (4). doi: 10.1145/2010324.1964964

Glover, A., and Smith, C. (2003). The deep-sea floor ecosystem: current status and prospects of anthropogenic change by the year 2025. *Environ. Conserv.* 30 (3), 219–241. doi: 10.1017/S0376892903000225

Haalboom, S., de Stigter, H., Duineveld, G., van Haren, H., Reichart, G., and Mienis, F. (2021). Suspended particulate matter in a submarine canyon (Whittard canyon, bay of Biscay, NE Atlantic ocean): assessment of commonly used instruments to record turbidity. *Mar. Geol.* 434, 106439. doi: 10.1016/j.margeo.2021.106439

Hein, J. R. (2004). "Cobalt-rich ferromanganese crusts: global distribution, composition, origin and research activities," in *Minerals other than polymetallic nodules of the international seabed area* (Kingston, Jamaica: International Seabed Authority), 188–256.

Hino, H., and Usui, A. (2022). Regional and fine-scale variability in composition and structure of hydrogenetic ferromanganese crusts: geological characterization of 25 drill cores from the Marcus-wake seamounts. *Mar. Georesources Geotechnol.* 40 (4), 415–437. doi: 10.1080/1064119X.2021.1904066

Islam, M. J., Xia, Y., and Scttar, J. (2019). Fast underwater image enhancement for improved visual perception. *IEEE Robot. Autom. Lett.* 5 (2), 3227–3234. doi: 10.1109/LRA.2020.2974710

Japan Oil, Gas and Metals National Corporation (2020) *News release: JOGMEC conducts world's first successful excavation of cobalt-rich seabed in the deep ocean; excavation test seeks to identify best practices to access essential green technology ingredients while minimizing environmental impact*. Available at: http://www.jogmec.go.jp/english/news/release/news_01_000033.html (Accessed November 4, 2022).

Jian, Q., Gu, Y., Li, C., Cong, R., and Shao, F. (2022). Underwater image enhancement quality evaluation: benchmark dataset and objective metric. *IEEE Trans. Circuits Syst. Video Technol.* 32 (9), 5959–5974. doi: 10.1109/TCSVT.2022.3164918

Jordan, M. I., and Mitchell, T. M. (2015). Machine learning: trends, perspectives, and prospects. *Science* 349 (6245), 255–260. doi: 10.1126/science.aaa8415

Kandimalla, V., Richard, M., Smith, F., Quirion, J., Torgo, L., and Whidden, C. (2022). Automated detection, classification and counting of fish in fish passages with deep learning. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.823173

Knausgård, K. M., Wiklund, A., Sørdalen, T. K., Halvorsen, K. T., Kleiven, A. R., Jiao, L., et al. (2022). Temperate fish detection and classification: a deep learning based approach. *Appl. Intell.* 52, 6988–7001. doi: 10.1007/s10489-020-02154-9

Lal, D. (1977). The oceanic microcosm of particles: suspended particulate matter, about 1 gram in 100 tons of seawater, plays a vital role in ocean chemistry. *Science* 198 (4321), 997–1009. doi: 10.1126/science.198.4321.997

Li, A., Anwar, S., and Porikli, F. (2020). Underwater scene prior inspired deep underwater image and video enhancement. *Pattern Recognit.* 98, 107038. doi: 10.1016/j.patcog.2019.107038

Lin, T., Goyal, P., Girshick, R., He, K., and Dollár, P. (2020). Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2), 318–327. doi: 10.1109/TPAMI.2018.2858826

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: Single shot MultiBox detector," in *Proceedings ofthe computer vision —ECCV 2016* (Cham, Swizerland: Springer), 21–37. doi: 10.1007/978-3-319-46448-0_2

Lombard, F., Boss, E., Waite, A. M., Vogt, M., Uitz, J., Stemmann, L., et al. (2019). Globally consistent quantitative observations of planktonic ecosystems. *Front. Mar. Sci.* 6. doi: 10.3389/fmars.2019.00196

Martín, J., Puig, P., Palanques, A., and Ribó, M. (2014). Trawling-induced daily sediment resuspension in the flank of a Mediterranean submarine canyon. *Deep-Sea Res. II* 104, 174–183. doi: 10.1016/j.dsr2.2013.05.036

Neri, A., Colonnese, S., Russo, G., and Talone, P. (1998). Automatic moving object and background separation. *Signal Process.* 66 (2), 219–232. doi: 10.1016/S0165-1684(98)00007-3

Omar, A. F. B., and MatJafri, M. Z. B. (2009). Turbidimeter design and analysis: a review on optical fiber sensors for the measurement of water turbidity. *Sensors* 9, 8311–8335. doi: 10.3390/s91008311

Ota, Y., Suzumura, M., Tsukasaki, A., Suzuki, A., Seike, K., and Minatoya, J. (2022). Sediment accumulation rates and particle mixing at northwestern pacific seamounts. *J. Mar. Syst.* 229, 103719. doi: 10.1016/j.jmarsys.2022.103719

Padilla, R., Netto, S. L., and Da Silva, E. A. B. (2020). "A survey on performance metrics for object-detection algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*. (Niteroi, Rio de Janeiro, Brazil: IEEE), 237–247. doi: 10.1109/IWSSIP48289.2020.9145130

Picheral, M., Guidi, L., Stemmann, L., Karl, D. M., Iddaoud, G., and Gorsky, G. (2010). The underwater vision profiler 5: an advanced instrument for high spatial resolution studies of particle size spectra and zooplankton. *Limnol. Oceanogr.: Methods* 8 (9), 462–473. doi: 10.4319/lom.2010.8.462

Ramirez-Llodra, E., Tyler, P. A., Baker, M. C., Bergstad, O. A., Clark, M. R., Escobar, E., et al. (2011). Man and the last great wilderness: human impact on the deep Sea. *PloS One* 6 (8), e22588. doi: 10.1371/journal.pone.0022588

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Las Vegas, NV, USA: IEEE), 779–788. doi: 10.1109/CVPR.2016.91

Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M. R., Khurshid, K., et al. (2020). Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES J. Mar. Sci.* 77 (4), 1295–1307. doi: 10.1093/icesjms/fsz025

Sharma, R., Nagender Nath, B., Parthiban, G., and Jai Sankar, S. (2001). Sediment redistribution during simulated benthic disturbance and its implications on deep seabed mining. *Deep-Sea Res. II* 48 (16), 3363–3380. doi: 10.1016/S0967-0645(01)00046-7

Spearman, J., Taylor, J., Crossouard, N., Cooper, A., Turnbull, M., Manning, A., et al. (2020). Measurement and modelling of deep sea sediment plumes and implications for deep sea mining. *Sci. Rep.* 10, (5075). doi: 10.1038/s41598-020-61837-y

Suzuki, A., Minatoya, J., Fukushima, T., Yokooka, H., Kudo, K., Sugishima, H., et al. (in review). Environmental impact assessment for small-scale excavation test of cobalt-rich ferromanganese crusts of a seamount in the northwestern pacific.

Tan, M., Pang, R., and Le, Q. V. (2020). "EfficientDet: scalable and efficient object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. (Seattle, WA, USA: IEEE), 10778–10787. doi: 10.1109/CVPR42600.2020.01079

Tomasi, C., and Manduchi, R. (1998). "Bilateral filtering for gray and color images," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*. (Bombay, India: IEEE), 839–846. doi: 10.1109/ICCV.1998.710815

Turnewitsch, R., Falahat, S., Nycander, J., Dalea, A., Scottc, R. B., and Furnival, D. (2013). Deep-sea fluid and sediment dynamics—Influence of hill- to seamount-scale seafloor topography. *Earth Sci. Rev.* 127, 203–241. doi: 10.1016/j.earscirev.2013.10.005

Tyler, P. (2003). Disposal in the deep sea: analogue of nature or faux ami? *Environ. Conserv.* 30 (1), 26–39. doi: 10.1017/S037689290300002X

Walther, D., Edgington, D. R., and Koch, C. (2004). "Detection and tracking of objects in underwater video," in *Proceedings of the 2004 IEEE Computer Society Conference*. (Washington, D.C., USA: IEEE), doi: 10.1109/CVPR.2004.1315079

Wang, Z. A., Moustahfid, H., Mueller, A. V., Michel, A. P. M., Mowlem, M., Glazer, B. T., et al. (2019). Advancing observation of ocean biogeochemistry, biology, and ecosystems with cost-effective *in situ* sensing technologies. *Front. Mar. Sci.* 6. doi: 10.3389/fmars.2019.00519

Wang, Y., Song, W., Fortino, G., Qi, L., Zhang, W., and Liotta, A. (2019). An Experimental-based review of image enhancement and image restoration methods for underwater imaging. *IEEE Access* 7, (99). doi: 10.1109/ACCESS.2019.2932130

Wang, Y., Yu, X., An, D., and Wei, Y. (2021). Underwater image enhancement and marine snow removal for fishery based on integrated dual-channel neural network. *Comput. Electron. Agric.* 186, 106182. doi: 10.1016/j.compag.2021.106182

Washburn, T. W., Turner, P. J., Durden, J. M., Jones, D. O. B., Weaver, P., and Van Dover, C. L. (2019). Ecological risk assessment for deep-sea mining. *Ocean Coast. Manage.* 176 (15), 24–39. doi: 10.1016/j.ocecoaman.2019.04.014"

Xue, B., Huang, B., Wei, W., Chen, G., Li, H., Zhao, N., et al. (2021). An efficient deep-Sea debris detection method using deep neural networks. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 14, 12348–12360. doi: 10.1109/JSTARS.2021.3130238

Zhang, W., Zhuang, P., Sun, H., Li, G., Kwong, S., and Li, C. (2022). Underwater image enhancement *via* minimal color loss and locally adaptive contrast enhancement. *IEEE Trans. Image Process.* 31, 3997–4010. doi: 10.1109/TIP.2022.3177129

Zhao, Z., Zheng, P., Xu, S., and Wu, X. (2019). Object detection with deep learning: a review. *IEEE Trans. Neural Netw. Learn. Syst.* 30 (11), 3212–3232. doi: 10.1109/TNNLS.2018.2876865

Zhu, F., Liang, Z., Jia, X., Zhang, L., and Yu, Y. (2019). A benchmark for edge-preserving image smoothing. *IEEE Trans. Image Process.* 28 (7), 3556–3570. doi: 10.1109/TIP.2019.2908778

Zou, Z., Chen, K., Shi, Z., Guo, Y., and Ye, J. (2023). Object detection in 20 years: a survey. *Proc. IEEE* 111 (3), 257–276. doi: 10.1109/JPROC.2023.3238524

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 17 July 2023
DOI 10.3389/fmars.2023.1197695

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Matteo Zucchetta,
National Research Council (CNR), Italy
Li Jiang,
Old Dominion University, United States

*CORRESPONDENCE
Lucas A. Langlois
lucas.langlois@jcu.edu.au

RECEIVED 31 March 2023
ACCEPTED 27 June 2023
PUBLISHED 17 July 2023

CITATION
Langlois LA, Collier CJ and McKenzie LJ (2023) Subtidal seagrass detector: development of a deep learning seagrass detection and classification model for seagrass presence and density in diverse habitats from underwater photoquadrats. *Front. Mar. Sci.* 10:1197695. doi: 10.3389/fmars.2023.1197695

COPYRIGHT
© 2023 Langlois, Collier and McKenzie. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Subtidal seagrass detector: development of a deep learning seagrass detection and classification model for seagrass presence and density in diverse habitats from underwater photoquadrats

Lucas A. Langlois*, Catherine J. Collier and Len J. McKenzie

Centre for Tropical Water and Aquatic Ecosystem Research (TropWATER), James Cook University, Cairns, QLD, Australia

This paper presents the development and evaluation of a Subtidal Seagrass Detector (the Detector). Deep learning models were used to detect most forms of seagrass occurring in a diversity of habitats across the northeast Australian seascape from underwater images and classify them based on how much the cover of seagrass was present. Images were collected by scientists and trained contributors undertaking routine monitoring using drop-cameras mounted over a 50 x 50 cm quadrat. The Detector is composed of three separate models able to perform the specific tasks of: detecting the presence of seagrass (*Model #1*); classify the seagrass present into three broad cover classes (low, medium, high) (*Model #2*); and classify the substrate or image complexity (simple of complex) (*Model #3*). We were able to successfully train the three models to achieve high level accuracies with 97%, 80.7% and 97.9%, respectively. With the ability to further refine and train these models with newly acquired images from different locations and from different sources (e.g. Automated Underwater Vehicles), we are confident that our ability to detect seagrass will improve over time. With this Detector we will be able rapidly assess a large number of images collected by a diversity of contributors, and the data will provide invaluable insights about the extent and condition of subtidal seagrass, particularly in data-poor areas.

KEYWORDS

seagrass, Great Barrier Reef, deep learning, image classification, underwater

1 Introduction

Seagrasses are one of the most valuable marine ecosystems on the planet, with their meadows estimated to occupy 16 - 27 million ha globally across a variety of benthic habitats within the nearshore marine photic zone (Mckenzie et al., 2020). Seagrass meadows are an integral component of the northeast Australian seascape that includes: the Great Barrier Reef, Torres Strait, and the Great Sandy Marine Park. Seagrass ecosystems in these marine domains are ecologically, socially and culturally connected and contain values of national and international significance (Johnson et al., 2018).

The Great Barrier Reef (the Reef) is the most extensive reef system in the world, in which seagrass is estimated to cover approximately 35,679 km^2 (Mckenzie et al., 2022b). Over 90% of the Reef's seagrass meadows occur in subtidal waters, with the deepest record to 76 m (Carter et al., 2021c), although most field surveys are in depths shallower than 15 m (Mckenzie et al., 2022b). There are 15 seagrass species reported within the Reef, occurring in estuaries, coastal, reef and deep water habitats and forming meadows comprised of different mixes of species (Carter et al., 2021a). Seagrass ecosystems of the Reef support a range of goods and benefits to species of conservation interest and society. The seagrass habitats of Torres Strait to the north are also of national significance due to their large extent, diversity and the vital role they play to ecology and the cultural economy of the region (Carter et al., 2021b). Similarly, the seagrasses within the Great Sandy Marine Park to the south support internationally important wetlands, highly valued fisheries and the extensive subtidal meadows in Hervey Bay are critical for marine turtles and the second largest dugong population in eastern Australia (Preen et al., 1995; Mckenzie et al., 2000). Catchment and coastal development, climate change and extreme weather events threaten seagrass ecosystem resilience and drive periodic decline. Maintaining up-to-date information on the distribution and condition of seagrass meadows is needed to protect and restore seagrass ecosystems.

A wide range of methods have been applied to assess and monitor changes in subtidal seagrass, including free-diving, SCUBA diving, towed camera, towed sled, grabs or drop–camera (Mckenzie et al., 2022b). Most of these techniques rely on trained scientists to visually confirm, quantify and identify the presence of seagrass in situ. This labour-intensive work, combined with the tremendously large area of the Reef, makes assessing the state (extent and condition) of subtidal seagrass prohibitively time consuming and expensive.

In recent years, the use of digital cameras and autonomous underwater vehicles (AUVs) has led to an exponential increase in availability of underwater imagery. When this imagery is geotagged or geolocated, it provides an invaluable resource for spatial assessments, and when collected by a range of providers and the wider community who are accessing the Reef for a range of other activities (tourism, Reef management), is highly cost effective. For example, the Queensland Parks and Wildlife Service uses drop-cameras to collect photoquadrats of the benthos within seagrass habitats for processing by and inclusion in the Inshore Seagrass component of the GBR Marine Monitoring Program (MMP). Recent projects such as The Great Reef Census (greatreefcensus.org) aim at tapping into the power of citizen science to collect images and provide new sources of information about the Reef. A similar approach could be applied to seagrass. This digital data can be analysed automatically if the workflows are in place to deal with structured big data streams.

Deep learning technology provides potentially unprecedented opportunities to increase efficiency for the analysis of underwater images. Deep learning models or Deep Neural Networks (DNNs) are being used for counting fish (Sheaves et al., 2020), identifying species of plankton (Schröder et al., 2020) and estimating macroalgae (Balado et al., 2021) or coral cover (Beijbom et al., 2015). Few studies explored their application for seagrass coverage estimation (Reus et al., 2018) as well as detection and classification (Moniruzzaman et al., 2019; Raine et al., 2020; Noman et al., 2021). While these showed interesting technical methods, they were not necessarily developed specifically for operational applications. An operational model that can detect seagrass within the Reef will improve our capability to rapidly assess and easily provide data critical for large scale assessments. In particular, there is a need for a model that can detect seagrass presence even with diverse physical appearances among the 15 species in the Reef, and in a range of habitat types with variable benthic substrates. As seagrass can also be very sparse in the Reef, with an historic baseline of 22.6 ± 1.2% cover (Mckenzie et al., 2015) and subtidal percent covers frequently less than 10%, a detector is needed to cope with such circumstances.

In this paper we detail the development of a Subtidal Seagrass Detector (the Detector) using a DNN to analyse underwater images to detect and classify seagrasses. This enables rapid processing of many images. It will form an integral step in workflow from image capture to provision of rapidly and easily accessed information. Up-to-date information on the extent and condition of seagrass is required for marine spatial planning and for the implementation of other management responses to protect Reef and seagrass ecosystems.

2 Material and methods

2.1 Detector model datasets

Our subtidal image dataset was composed of 7440 photoquadrats collected by drop-camera and SCUBA divers as part as the MMP (Mckenzie et al., 2022a), the Seagrass-Watch Global Seagrass Observing Network (Seagrass-Watch, 2022) and the Torres Strait Ranger Subtidal Monitoring Program (Carter et al., 2021b). Images were captured between 2014 and 2021 from 28 sites across 18 unique locations within the coastal and reef subtidal habitats from Torres Strait to Hervey Bay (Figure 1; Supplementary Table S1). Images were annotated by assessing: (1) the percent cover of seagrass (Mckenzie et al., 2003), (2) the seagrass morphology of the dominant species based on largest percent cover (straplike, oval–shaped or fernlike), (3) percent cover of algae, (4) substrate complexity (simple or complex), and (5) quality of the photo (0=photo unusable, 1=photo clear with more than 90% of quadrat in the frame, 2=photo with bad visibility with more than 90% of quadrat in the frame, 3= photo clear with quadrat partially not visible, 4= photo oblique with quadrat not totally on the bottom). Only photos with a rating of 1 (5782 in total)

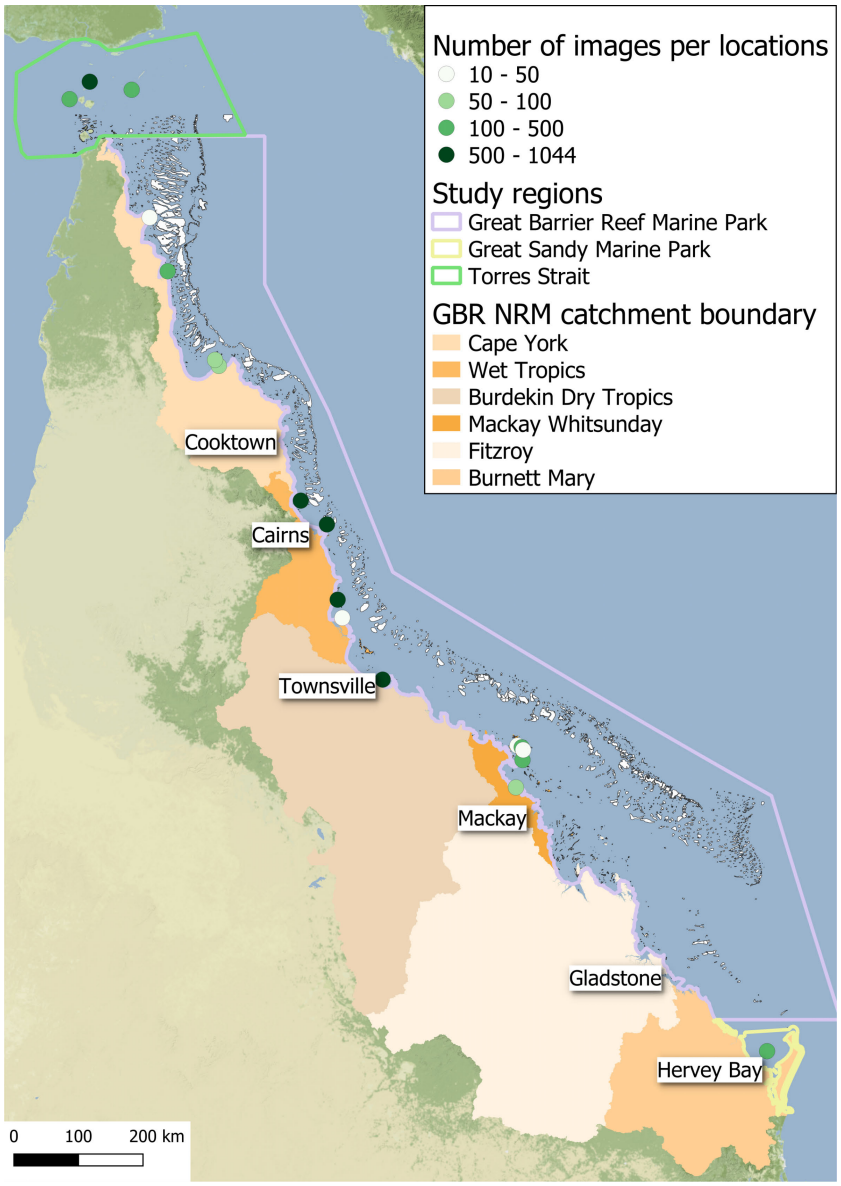


FIGURE 1
Map showing the location and number of images used for the Subtidal Seagrass Detector in the Torres Strait, the Great Barrier Reef World Heritage Area and Great Sandy Marine Park.

were retained to ensure optimal performance. All images were cropped to the outer boundary of the quadrat and standardised to a 1024 × 1024 pixel size.

2.1.1 Seagrass presence detector (*Model #1*)

We defined seagrass presence as an area of the seafloor, also known as benthos, spatially dominated by seagrass, which we classed as ≥3% cover (sensu Mount et al., 2007). Images with seagrass cover less than 3% were excluded, resulting in the removal of an additional 819 images from the analysis. This maximised the power of detection to levels where seagrass was clearly visible. There were 1727 images with seagrass absent and 3236 with seagrass present. To ensure a balance dataset of the two classes, 1727 images were chosen at random out of the 3236 while ensuring the inclusion of all images from the minor seagrass morphology classes oval-shaped (522) and fernlike (165). The remaining images with seagrass present (1509) were retained for further testing.

2.1.2 Seagrass cover category classifier (*Model #2*)

Cover categories were first established based on four cover quantiles, which were equivalent to seagrass percent cover categories of; ≥3 <9%, ≥9 <15%, ≥15 <30% and ≥30%. However, the resulting model did not adequately distinguish between the two middle categories (less than 60% accuracy). Therefore, those two classes were merged resulting in three main classes used in *Model #2*: (1) low seagrass cover (≥3 <10%), (2) medium seagrass cover

(≥10 <30%), and (3) high seagrass cover (≥30%) (Figure 2). The classes were somewhat unbalanced with 1082, 1509 and 644 images respectively. However, more images in the medium class were beneficial as it helped improve accuracy for that class which is the most commonly occurring at MMP sites (long term mean of 14% seagrass cover for coastal and reef subtidal sites where seagrass is present). When we ran the same model on a down-sampled version of the dataset (644 images for each class) the overall accuracy was lower (-3.3%): accuracy for the low and high cover class increased (+12.5% and +7.9% respectively), while the accuracy for the medium class significantly decreased (-26.3%).

2.1.3 Substrate complexity classifier (*Model #3*)

The substrate complexity classifier was applied to all images without any seagrass present. Those images were labelled either as 'simple substrate' or as 'complex substrate'. The 'simple' category was assigned to clear images with mostly sandy bottoms while the 'complex' category was assigned to images that met at least one of the following conditions:

- had consolidated substrates, such as rock, live coral or coral rubble
- had a visually significant amount of macroalgae
- labelling was difficult (e.g. poor visibility, small seagrass species, poor image contrast).

Out of the 1727 images without seagrass, 1129 had simple substrate and 598 had complex substrate. Similar to *Model #1*, a random 531 simple substrate images were excluded and retained for further testing to unsure a balance dataset during training. This classifier can provide a potential reason for the absence of seagrass as well as highlighting potential shortfall in the seagrass detection from *Model #1*. In complex substrate habitats, seagrass could be present, however, percent cover is most likely to be low (<10%) and particularly difficult to detect by the model. Images predicted into this category can be later manually inspected to confirm the absence of seagrass.

All three final datasets were split 60-20-20 into a training, validation and test set.

2.2 Deep neural network modelling

2.2.1 Image classification workflow

Our overall aim for this study was to develop a Detector that would be able to achieve three separate classification tasks: (1) detect the presence/absence of seagrass, (2) estimate the seagrass cover (low, medium or high), and (3) identify the level of complexity of the substrate (simple or complex). Separate deep learning models were developed to execute each of these tasks independently which maximised model accuracy and reduced category imbalance (Figure 3). All model training and testing was conducted in Python using Keras (Chollet, 2015) on a local machine (Intel Core i9-10900KF CPU 3.70GHz, 3696 Mhz, 10 Cores, 20 Logical Processors, 64GB 3200 MHz, GPU NVIDIA GeForce RTX 3090).

2.2.2 Model architecture

The classification models were composed of a binary classification model for *Model #1* and *Model #3* and multiclass classification for *Model #2*. The classification employed deep learning also known as DNNs. Training a neural network can be a protracted process and requires a large number of images to achieve satisfactory results. Transfer learning has been developed where an already successfully trained network such as VGG16 can be used as a feature extractor and coupled with a new classifier trained for the new specific task (Tammina, 2019). Our initial network was composed of a VGG16 model pre-trained on the

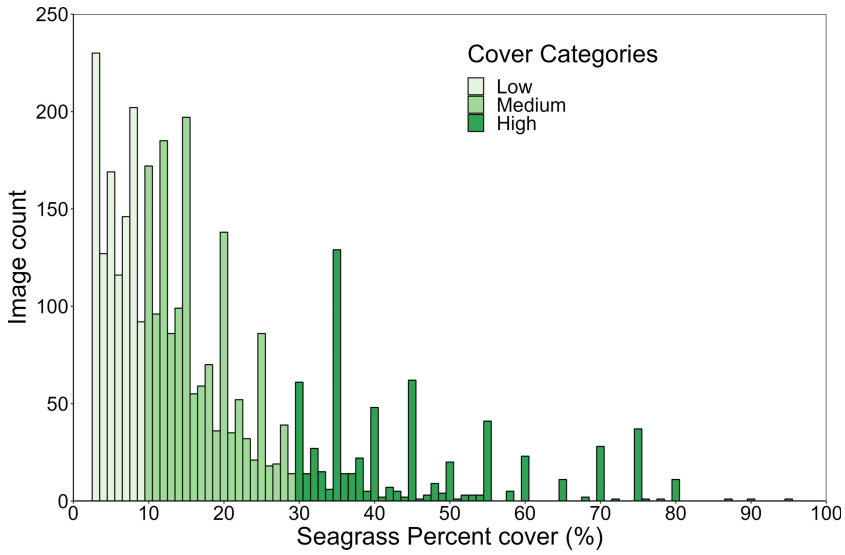


FIGURE 2
Distribution of seagrass percent cover in the image dataset used for Model #2.

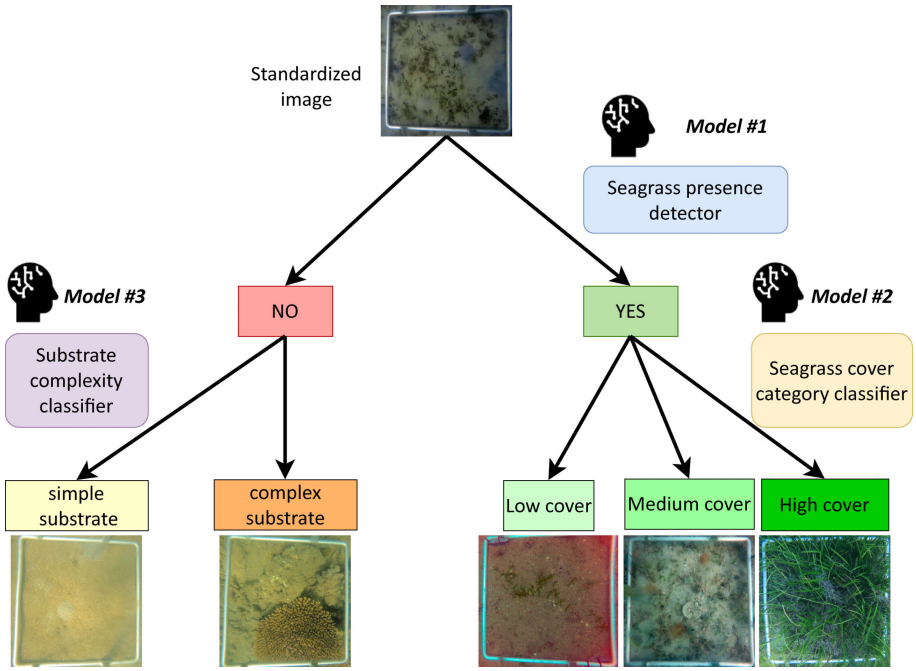


FIGURE 3
Diagram detailing the image classification workflow process of the Detector with the three deep learning models involved.

ImageNet classification tasks (Zhang et al., 2015). Instead of the final dense layer from the original VGG16 model, we created our own custom classier composed of a sequence of two fully connected layers with 512 nodes and ReLU activation (Agarap, 2018), two consecutive dropout (Srivastava et al., 2014) with probability of 0.05 and 0.5 to prevent overfitting and a final dense layer with one node for each of predicted class activated by either the Sigmoid or Softmax function (Figure 4).

Contrary to other studies (Raine et al., 2020) we chose not to split our original images as it would have meant having to create new labels for thousands of sub-images. Instead, the input image size was increased. After multiple trials we found that optimal results were achieved for the input size of 1024x1024 pixels. We also tried more complex networks for feature extraction such as Resnet50 and EfficientNet but they did not perform as well overall (-1.7 and -5.2% in overall accuracy respectively).

2.2.3 Model training

The DNNs were all trained independently on batches of eight random images per training iteration. When the DNN has gone through as many iterations as needed to process the full training image set, this constituted an epoch. Throughout the whole training process, the progress of the learning is monitored by evaluating the model performance on the validation image set.

We started with an initial training phase where only the final classification layers (custom classifier part) were trainable and the rest of the VGG16 layers were frozen. During this phase the Adam optimizer (Kingma and Ba, 2014) was used with an initial learning rate of 0.001. If the loss on the validation image set did not improve after 10 epochs the learning rate was reduced by half up to four times after which the training was stopped. That process lasted 60 to 68 epochs. A fine-tuning training phase followed, where the VGG16 layers were unfrozen and set as trainable. This was done over 100 epochs and with the RMSprop optimizer (Tieleman and Hinton, 2014) and a much slower learning rate of 0.00001. The fine-tuning is meant to ensure the feature extraction is optimised for our input size as well as increasing performance of the models.

To further prevent overfitting and best capture, the potential illumination and turbidity variations of underwater images, colour-based data augmentation was applied where brightness (-70 to 70), contrast (0.1 to 0.3), blur (sigma 0 to 0.5) and the red channel (-50 to 50) were randomly altered at each training iteration.

2.2.4 Model evaluation (testing)

The training process stopped once all the DNNs have reached a plateau where further training did not further improve performances on the validation set.

We then conducted final evaluation of the model performances on the test image set (20% of the total) where accuracy was assessed in detail. For *Model #1* and *Model #3*, further testing was conducted by running the model on the remaining images not included in the training, validation and test sets.

3 Results

3.1 Model #1

Model #1 achieved 97.0% accuracy (Supplementary Table S2) on the test image set (691). We had 3 false positive and 18 false negative

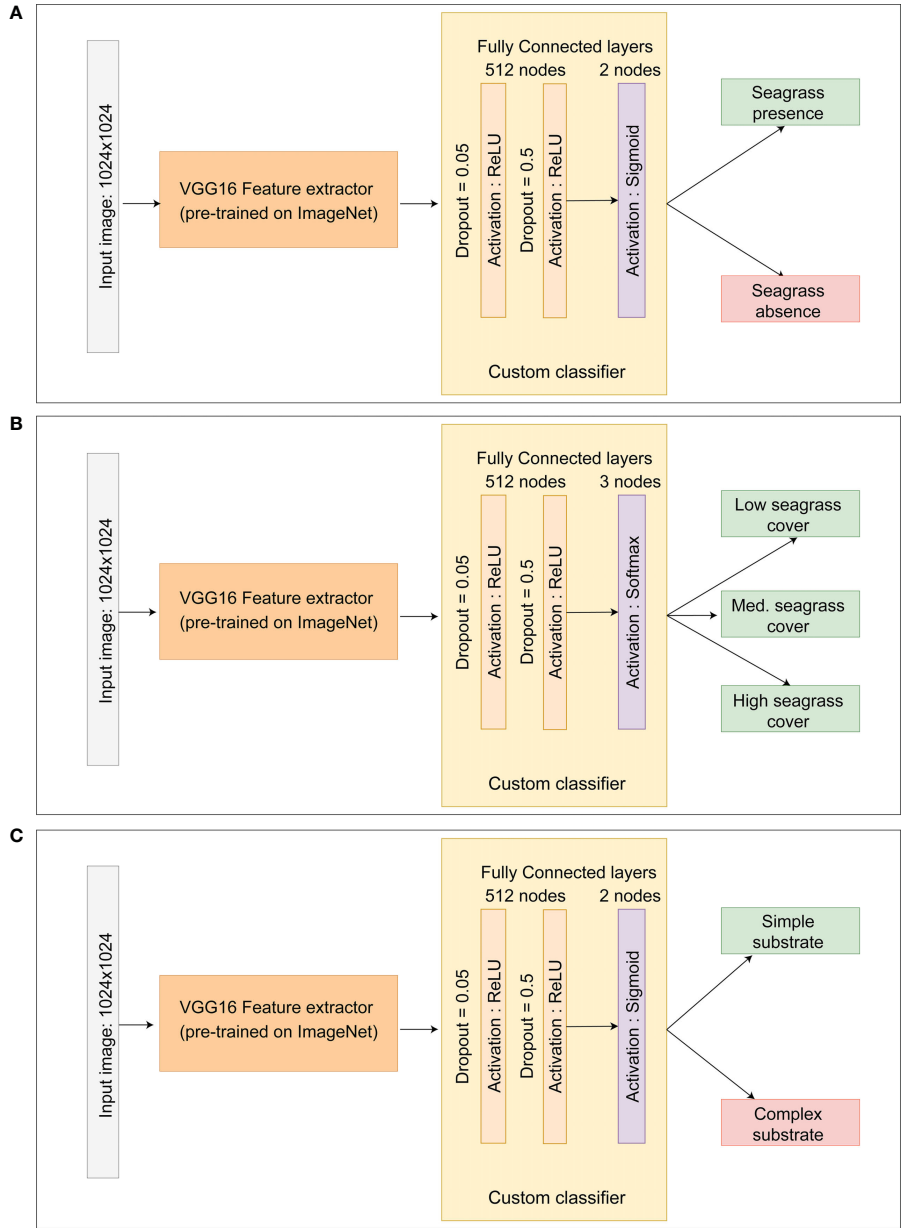


FIGURE 4
Convolution neural network architecture of: **(A)** Model #1, **(B)** Model #2 and **(C)** Model #3.

classifications (Figure 5; Supplementary Table S3A). The false positives were all images from Low Isles and taken on SCUBA. We suspect that the presence of turf algae and the low image quality could be the source of the misclassification. The small number of false positives suggests the model was not overestimating seagrass presence.

Of the false negative images, 16 had a percent cover lower than 10% and in nine of these percent cover was lower than 5% (Figure 6). In addition, 14 of the false negative images had a complex substrate with seven having more than 15% algae cover. This was further confirmed by running the model on the remaining seagrass photos not included in the training, validation and test sets. The model failed to detect seagrass in 38 out of 1509 images, achieving 97.4% accuracy. A similar pattern was observed where 31 of the misclassified images had less than 10% seagrass cover and 33 had complex substrate (Figure 6).

3.2 Model #2

Model #2 had an overall accuracy of 80.7% (Supplementary Table S2) on the test image set (647). The highest accuracy was achieved for the medium cover class (84.3%), followed by the low cover class (78.5%) and the high cover class (75.9%). However, these differences in accuracies were marginal and most likely a consequence of the unbalanced nature of the cover classes image dataset (Figure 7; Supplementary Table S3C).

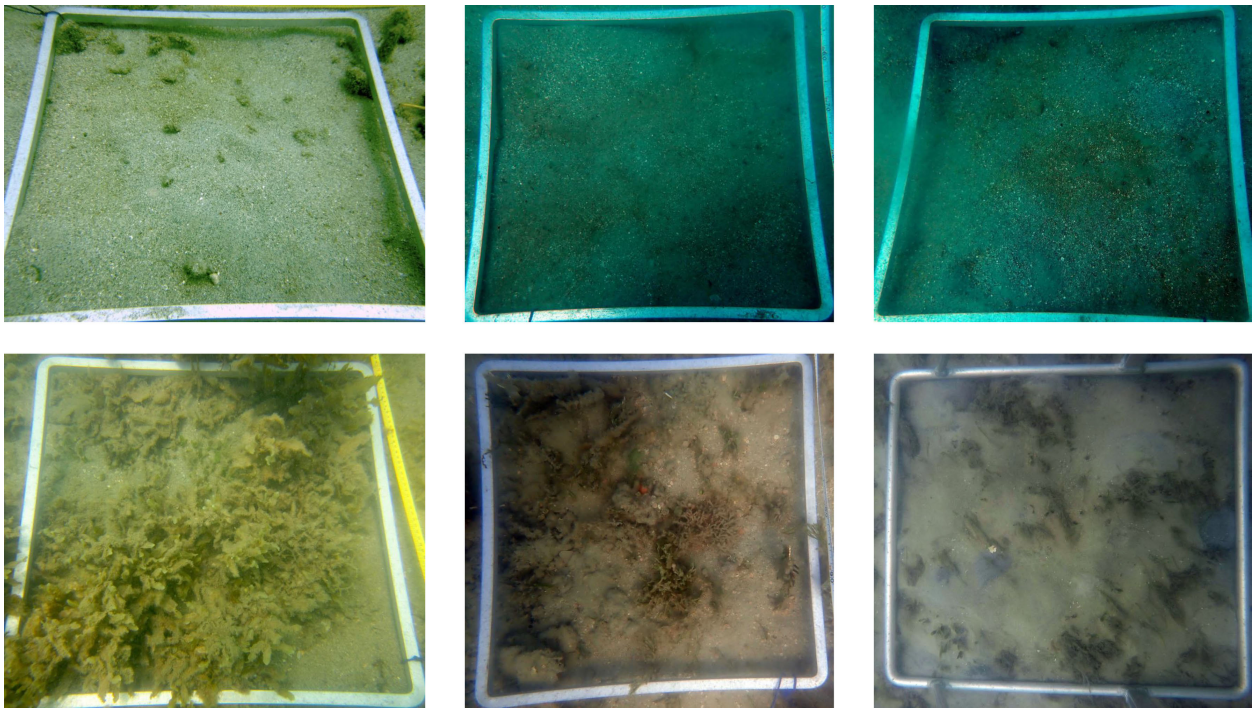


FIGURE 5
Examples of images misclassified by Model #1 with false positives on top row and false negative on the bottom row.

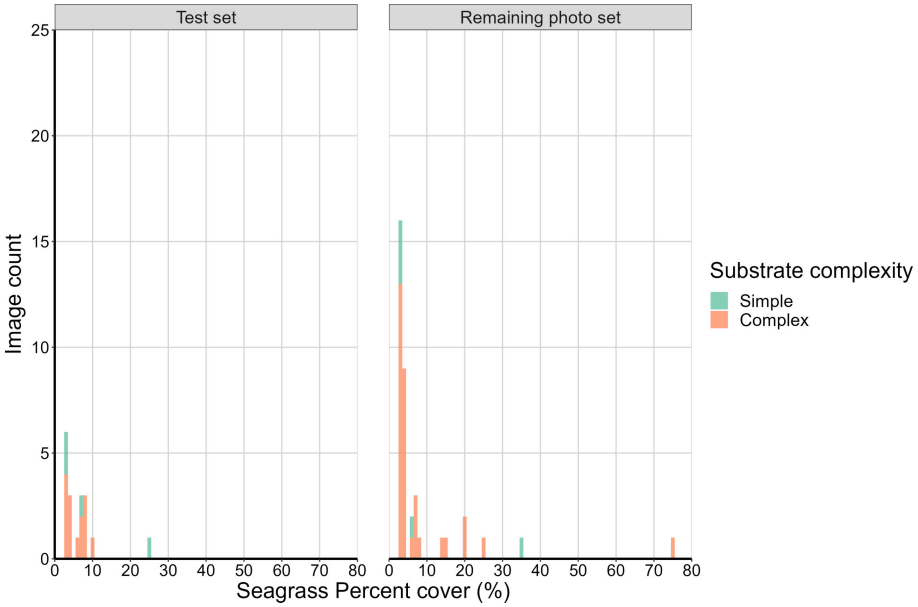


FIGURE 6
Histogram of the distribution of the seagrass percent cover and substrate complexity present in the images misclassified (false negative) by Model #1 from the test set (18) and the remaining seagrass photo set (38).

All the misclassified images of the low cover classes (45) were incorrectly predicted to be in the medium cover category. Misclassification occurred for images with percent cover between 7 and 9% (31) (Figure 8A). Furthermore, 32 of which also had a complex substrate, further highlighting the difficulty categorising images close to the threshold of 10%, especially for complex substrates where algae for example could be biasing the predictions.

There were 48 misclassified images of the medium cover classes, of which 31 were predicted as low cover and 17 as high cover. The false low cover images were mostly close to the 10% threshold with

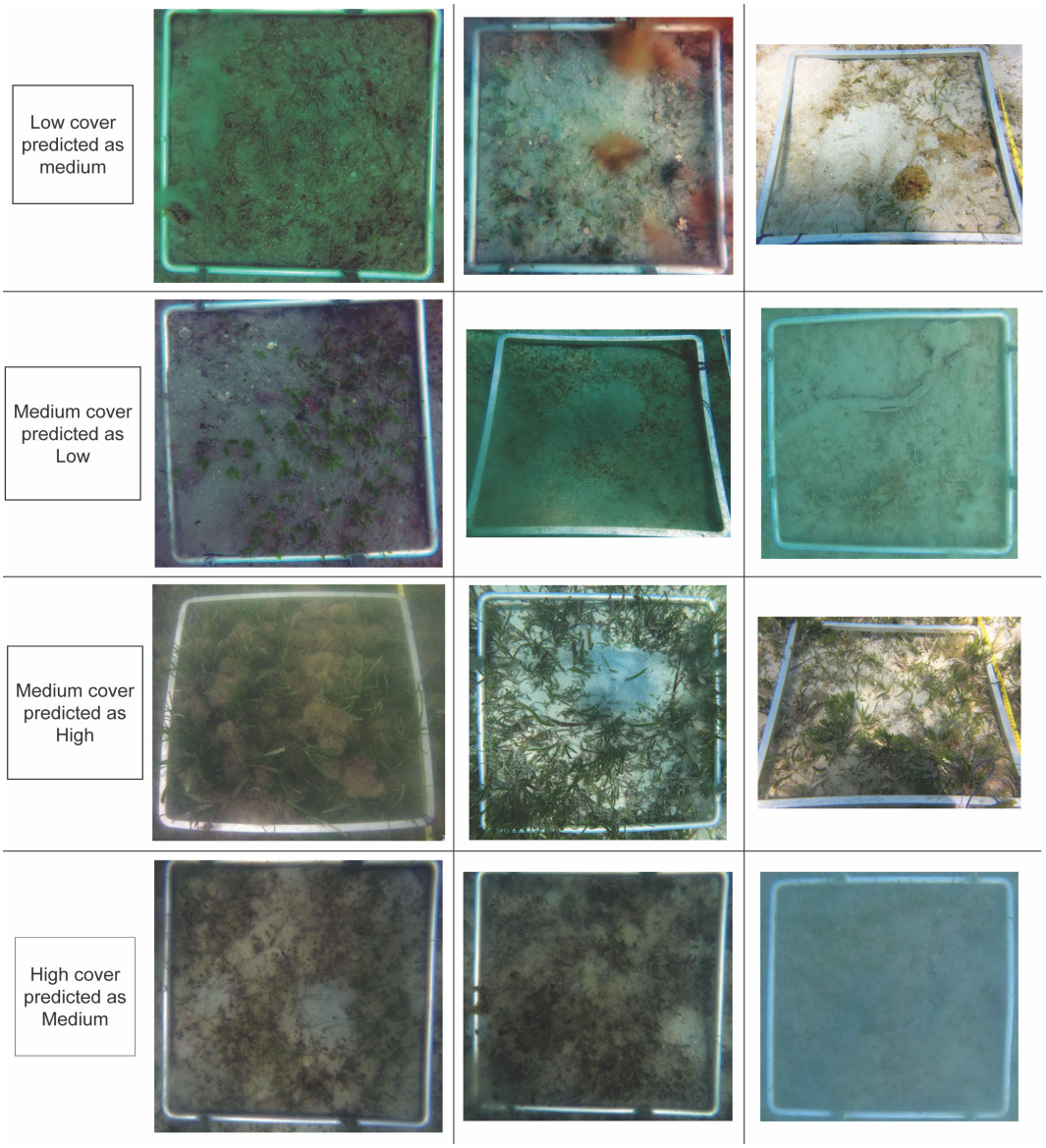


FIGURE 7
Examples of images misclassified by Model #2 from the low, medium and high cover categories.

27 of these images being between 10 and 15% seagrass cover (Figures 8B, C). Images dominated by smaller seagrass species with rounded and fernlike morphology were also a source of misclassification. The false high classifications were solely dominated by straplike species and 10 images had a seagrass cover between 20 and 30% (Figures 8D, E).

There were 32 misclassified images of the high cover class, which were all predicted as a medium cover. Similar to the previous classes, a vast majority of these were close to the adjacent cover category threshold with 28 of these images having less than 38% seagrass cover (Figures 8F, G). Straplike morphology dominated in 27 of the misclassified images except for those with percent cover of more than 40% which were dominated by rounded and fernlike morphology.

The type of substrate was not a significant driver of prediction errors for the medium and high cover class.

3.3 Model #3

Our subtidal substrate complexity classifier (*Model #3*) achieved an accuracy of 97.9% (Supplementary Table S2) on the test image set (240) and on the simple substrate only images remaining (531). There were two images misclassified as complex and three images were misclassified as simple instead of complex out of the test image set (Figure 9; Supplementary Table S3D). These images were also difficult to manually classify because they were mostly composed of a simple sandy substrate with some additional features such as algae or soft coral, or have poor visibility.

There were 11 images misclassified as complex instead of simple out of the simple substrate images remaining. These had 7% algae cover on average and 10 had more than 3%. This may be a consequence of the arbitrary binary classification used during the

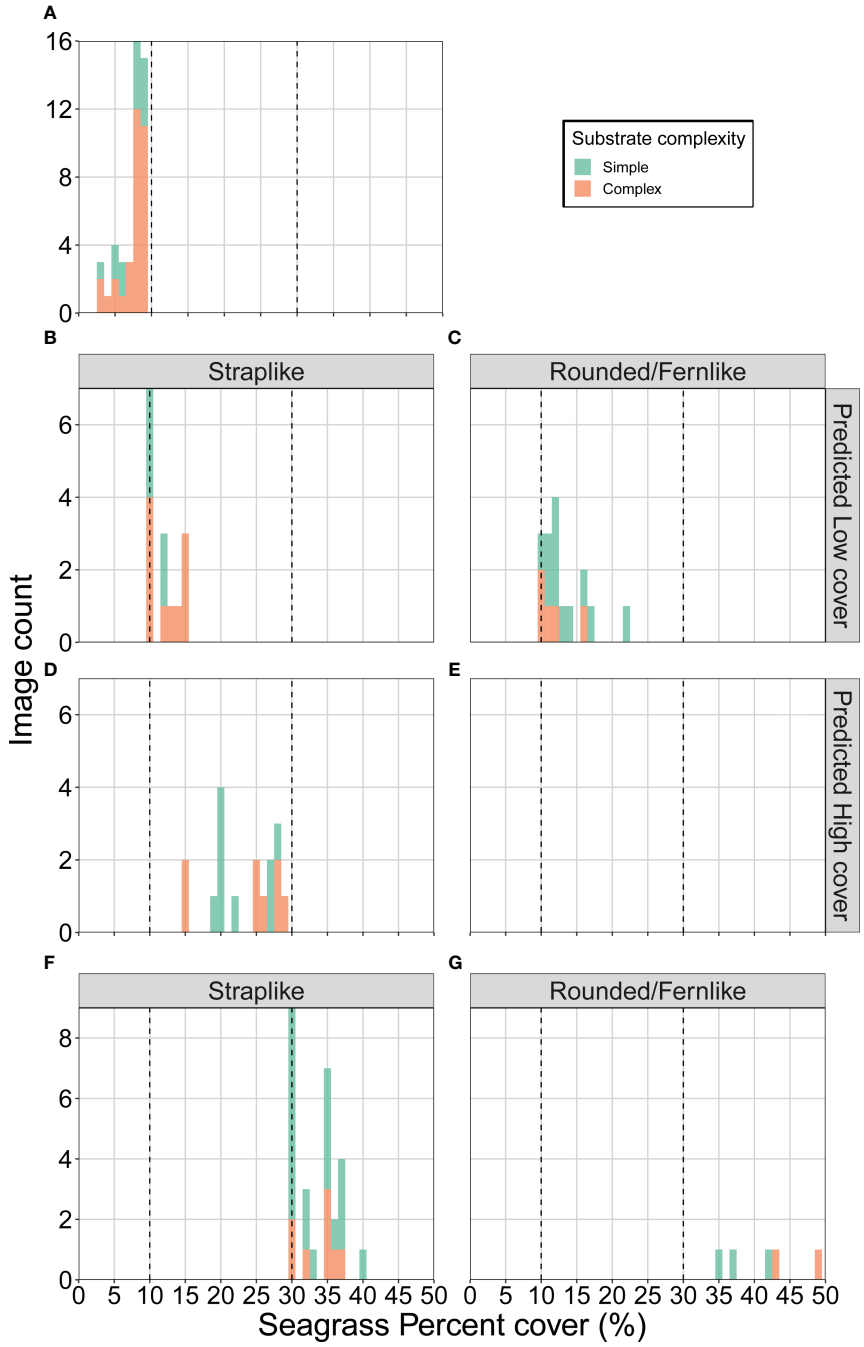


FIGURE 8
Histogram of the distribution of the seagrass percent cover and substrate complexity present in the misclassified images by Model #2 of **(A)** the low cover category (false medium), the medium cover category with **(B, C)** false low and **(D, E)** false high for straplike and rounded/fernlike species, and the high cover category (false medium) for **(F)** straplike and **(G)** rounded/fernlike.

labelling process. It is very difficult to establish a clear difference between a quadrat with a simple sandy substrate with some algae or other features like coral and a complex substrate. These instances are uncommon within the dataset, with 82 images labelled as simple substrate and more than 3% algae cover and occurred mainly only at the Dunk Island and Low Isles sites (36 and 30 images respectively). This could be easily refined further by increasing the image dataset and by setting clearer thresholds or rules to define the substrate complexity classes.

4 Discussion

4.1 Method performance and limitation

The main goal of this research was to determine the potential for deep learning models to detect the presence of seagrass within underwater photos. Seagrass was identified in images containing a mix of seagrass species, seagrass morphologies and from a range of habitats/substrates with a very high level of accuracy (97%). This

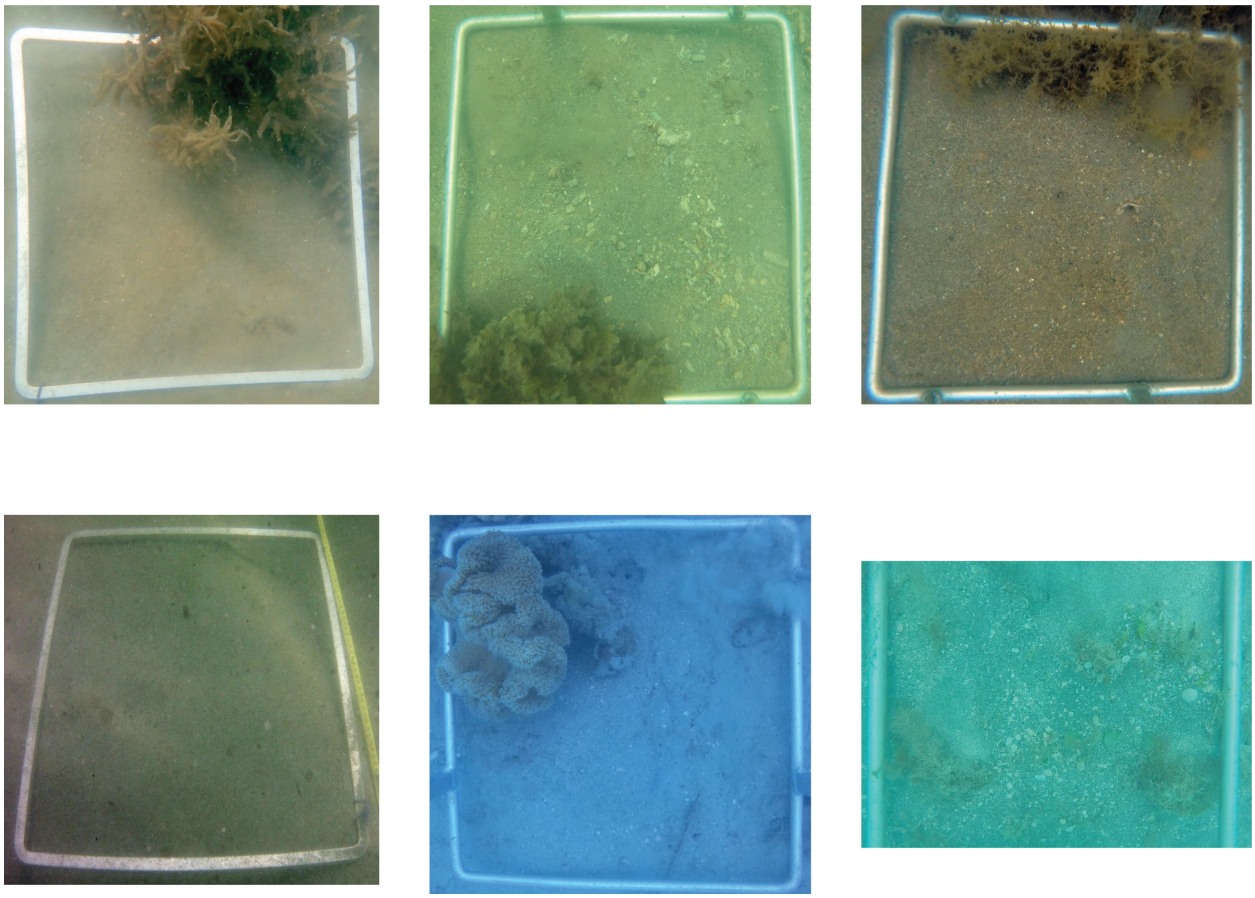


FIGURE 9
Examples of images misclassified by Model #3 with false complex on top row and false simple on the bottom row.

was achieved using a simple neural network architecture. The performance of *Model #1* was higher than previously published deep learning seagrass detection models (Raine et al., 2020). However, a direct comparison between the accuracies is difficult due to differences in image dataset size and classifiers for seagrass morphology between studies.

We found that most of the misclassification occurred for images with complex substrate especially those with high algae percent cover. This is typical for deep learning classification models that are still lacking the ability to apply extreme generalization the way humans do (Chollet, 2017). Differentiating among well-defined objects is usually straight forward with numerous documented examples on image datasets such as ImageNet (Krizhevsky et al., 2017). The model outcomes for complex substrate, could possibly be improved by increasing the overall number of images, but also by having a balanced number of images with the same level of algae with and without seagrass. Indeed, deep learning models can continue to "learn" with additional imagery, so as new images are being collected, our models can be further trained which will lead to improved performance over time.

We also demonstrated it was possible to categorise seagrass cover into three broad classes with an accuracy of 80.7%. The choice of category boundaries was crucial in the model performance. Most of the classification errors happened around these boundaries and resulted in an image being placed into the adjacent category, rather than for example two categories away (i.e. a high being classed as low or vice versa). This needs to be considered when applying the model. For instance, the medium seagrass cover category was defined as ≥10 <30% during the labelling process, however the percent cover range of the images predicted in that class ranged from 7 to 35%. Despite these misclassification potential errors, using broad seagrass cover categories is sufficient in the context of mapping. At the scale of the photoquadrat used in this study, it is currently more accurate and easier to assess seagrass cover with a classification model rather than a regression model via segmentation of the image. Because of the very small morphology of the seagrass species in the Reef and the high level of complexity in the background (e.g. macroalgae, rubbles, turf algae), automated segmentation or even manual annotation of seagrass leaves is incredibly difficult in particular for strap-like species.

Seagrass percent cover estimates can be difficult to assign for low densities. Except for a few structurally large species, individual seagrass leaves are very small and therefore may not be easy to identify. A study from Moniruzzaman et al. (2019) developed deep learning models to detect single leaves of *Halophila ovalis*. This was effective for oblique close-up images with a sand background, but is likely to be less effective with nadir quadrat images as used in this study. Photoquadrats are used so that cover can be easily quantified

in a standardised manner. While it would require a significant effort to label a photoquadrat dataset with individual bounding boxes, it might be the best way to detect very low seagrass density (<3%) and deserves further investigation.

An alternative method to estimate percent cover of benthic taxa (e.g. coral, algae, seagrass) and substrate (e.g. sand, rock) is using a point annotation system. This method has been successfully used for coral reefs and invertebrate communities (González-Rivero et al., 2016) and is publicly available through platforms such as CoralNet or ReefCloud. In seagrass habitats, the point annotation method is only able to detect seagrass when cover is above 25% (Kovacs et al., 2022). This is because the method relies on classifying an area (224x224 pixels) around the annotated point. The dimension of the annotation area is not visible through the labelling interface and the person conducting the labelling is expected to label only what is directly under the point. This approach is appropriate for well–defined and larger objects like coral, however, it is not well adapted to scattered, low and sparse seagrass cover where there could be seagrass within the classifying area but not directly under the point, resulting in a high level of misclassification. By classifying the patches directly, others studies have shown very high overall accuracy for multi-species seagrass detection (Raine et al., 2020) and even the addition of semi-supervised learning to reduce labelling effort (Noman et al., 2021). However, this was achieved on a dataset composed of images from Moreton Bay (Queensland, Australia), which does not encompass all species present within our study area and does not include complex substrate background.

While we acknowledge the limitations of our models, especially *Model #2*, we believe to have developed the most operationally relevant subtidal seagrass detection deep learning model for the Reef to date with a lot of potential for future improvements.

4.2 Operationalisation and mainstreaming

This study was undertaken to demonstrate the feasibility of a subtidal seagrass detection model as a step towards operationalisation and mainstreaming of big data acquisition and analysis (Dalby et al., 2021).

Traditional direct field observations provide instantaneous data, but need to be performed or overseen by formally trained scientists, and the data requires time consuming transcription into a database. Images (e.g. photoquadrats), however, can be collected by a variety of contributors such as environmental practitioners, Indigenous ranger groups or members of the public without a formal scientific background (i.e. citizen scientists), requiring less capacity and resources. For example, rangers from the Queensland Park and Wildlife Services (QPWS) conduct subtidal seagrass monitoring using drop cameras that is currently integrated into the MMP (Mckenzie et al., 2021). Citizen scientists, QPWS Rangers and Indigenous rangers frequently access the Reef and seagrass habitats of northern Australia. Simplifying the methods and minimising the time required to capture data by using photoquadrats can vastly increase the volume, velocity, variety and geographic spread of image data collection. The models presented in this study facilitate the ability to mainstream data capture and increase the rate of image processing, enabling scientists to maximise big data analysis and reporting. With our current computer, the models are able to process and produce predictions for 1500 images in under two minutes. In our experience it would take approximately 12 to 25 hours for a trained person to manually label that number of images depending on their complexity. Scaling up the process will require some specific infrastructure to store data and powerful cloud computing capacity (CPU and GPU) on platforms such as AWS or Azure to handle on-demand inference of new data. In addition of the deep learning models, we aim to grow our capacity for image data handling. In parallel with the development of the models presented here we have been working on streamlining a higher efficiency image processing workflow. This includes handling either time-lapse or video (e.g. GoPro) input sources and a DDN model (YOLOv5) to generate deep learning ready standardized quadrat images via detecting quadrat metal frame and cropping the image.

The operational applications for the subtidal seagrass detector are wide-ranging, including mapping and monitoring of the vast and remote northern Australian and global seagrass habitats. Image collection combined with a geotagging/geolocation, will enable the production of spatially explicit maps of subtidal areas. Our models are most adapted to this application as maps tend to only need simple information like seagrass presence/absence. However, we have also shown potential for monitoring with the ability to detect broad seagrass cover categories which with further refinement could enable temporal changes in seagrass abundance to be assessed.

4.3 Future directions

While the findings in this study are encouraging, we very much intend to further refine and improve those models and the associated data processing workflow over time. One of the main advantages of using DNNs is their capacity to incrementally improve when additional training data is provided. Therefore, as more and more diverse images are supplied it will help us build more robust models and give greater confidence in the predictions. Our models are currently limited to be used on subtidal nadir photoquadrats captured using a drop-camera. However, with the increasing popularity of Autonomous Underwater Vehicles (AUVs), our DNNs would need to be trained to accept more versatile image inputs (e.g. oblique and without guiding bounds).

5 Conclusion

In this study, we developed a Subtidal Seagrass Detector capable of detecting the presence of seagrass as well as classifying seagrass cover and substrate complexity in underwater photoquadrats by using Deep Neural Networks. The three subsequent models achieved high level accuracies with 97%, 80.7% and 97.9%, respectively. This demonstrates great potential towards the

operationalisation of the Detector for accurate automated seagrass detection over a wide range of subtidal seagrass habitats.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

Conceptualisation, LL, CC and LM. Methodology, LL, CC and LM. Validation, LL, CC and LM. Formal analysis, LL. Investigation, LL, CC and LM. Resources, LL, CC and LM. Data curation, LL. Writing—original draft preparation, LL, CC and LM. Writing—review and editing, LL, CC and LM. Visualisation, LL, CC and LM. All authors contributed to the article and approved the submitted version.

Funding

This research was internally funded by the Centre for Tropical Water and Aquatic Ecosystem Research (TropWATER), James Cook University, Cairns, QLD, Australia.

Acknowledgments

We thank Rudi Yoshida, Haley Brien, Abby Fatland, Jasmina Uusitalo and Miwa Takahashi assistance with data collection. We acknowledge the Australian Government and the Great Barrier Reef Marine Park Authority (the Authority) for support of the collection of the Great Barrier Reef data under the Marine Monitoring Program. We thank Sascha Taylor and the QPWS rangers who conducted the subtidal drop camera field assessments in Cape York, southern Wet Tropics and Mackay–Whitsunday regions of the Great Barrier Reef. Torres Strait seagrass images were collected by the Sea Team of the Land and Sea Management Unit as part of seagrass monitoring funded by Torres Strait Regional Authority. We thank Damien Burrows for his support for this research and Mohammad Jahanbakht for help reviewing the manuscript. We also thank all Traditional Owners of the sea countries we visited to conduct our research: Wuthathi, Uutaalnganu, Yiithuwarra, Eastern Kuku Yalanji, Gunggandji, Djiru, Bandjin, Wulgurukaba, Ngaro, Gia and Butchulla. Parts of this manuscript has been released as a technical report at TropWATER, (Langlois et al., 2022).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1197695/full#supplementary-material

References

Agarap, A. F. (2018). Deep learning using rectified linear units (ReLU). *arXiv*. doi: 10.48550/arxiv.1803.08375

Balado, J., Olabarria, C., Martínez-Sánchez, J., Rodríguez-Pérez, J. R., and Pedro, A. (2021). Semantic segmentation of major macroalgae in coastal environments using high-resolution ground imagery and deep learning. *Int. J. Remote Sens.* 42, 1785–1800. doi: 10.1080/01431161.2020.1842543

Beijbom, O., Edmunds, P. J., Roelfsema, C., Smith, J., Kline, D. I., Neal, B. P., et al. (2015). Towards automated annotation of benthic survey images: variability of human experts and operational modes of automation. *PloS One* 10, e0130312. doi: 10.1371/journal.pone.0130312

Carter, A. B., Collier, C., Lawrence, E., Rasheed, M. A., Robson, B. J., and Coles, R. (2021a). A spatial analysis of seagrass habitat and community diversity in the great barrier reef world heritage area. *Sci. Rep.* 11, 22344. doi: 10.1038/s41598-021-01471-4

Carter, A. B., David, M., Whap, T., Hoffman, L. R., Scott, A. L., and Rasheed, M. A. (2021b). *Torres Strait seagrass 2021 report card. TropWATER report no. 21/13* (Cairns: TropWATER, James Cook University).

Carter, A. B., Mckenna, S. A., Rasheed, M. A., Collier, C., Mckenzie, L., Pitcher, R., et al. (2021c). Synthesizing 35 years of seagrass spatial data from the great barrier reef world heritage area, Queensland, Australia. *Limnol. Oceanogr. Lett.* 6, 216–226. doi: 10.1002/lol2.10193

Chollet, F. (2015). keras, GitHub. Available at: https://github.com/fchollet/keras.

Chollet, F. (2017). *Deep learning with python* (Shelter Island, NY: Manning Publications).

Dalby, O., Sinha, I., Unsworth, R. K. F., Mckenzie, L. J., Jones, B. L., and Cullen-Unsworth, L. C. (2021). Citizen science driven big data collection requires improved and inclusive societal engagement. *Front. Mar. Sci.* 8. doi: 10.3389/fmars.2021.610397

González-Rivero, M., Beijbom, O., Rodriguez-Ramirez, A., Holtrop, T., González-Marrero, Y., Ganase, A., et al. (2016). Scaling up ecological measurements of coral reefs using semi-automated field image collection and analysis. *Remote Sens.* 8, 30. doi: 10.3390/rs8010030

Johnson, J. E., Welch, D. J., Marshall, P., Day, J., Marshall, N., Steinberg, C., et al. (2018). *Characterising the values and connectivity of the northeast Australia seascape: great barrier reef, Torres strait, coral Sea and great sandy strait. report to the national environmental science program* (Cairns: Rainforest Research Centre Limited).

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv*. doi: 10.48550/arxiv.1412.6980

Kovacs, E. M., Roelfsema, C., Udy, J., Baltais, S., Lyons, M., and Phinn, S. (2022). Cloud processing for simultaneous mapping of seagrass meadows in optically complex and varied water. *Remote Sens.* 14, 609. doi: 10.3390/rs14030609

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM* 60, 84–90. doi: 10.1145/3065386

Langlois, L. A., Collier, C. J., and Mckenzie, L. J. (2022). *Subtidal seagrass detector: development and preliminary validation* (Cairns: Centre for Tropical Water & Aquatic Ecosystem Research, James Cook University). Available at: https://bit.ly/3WqYAQN.

Mckenzie, L. J., Campbell, S. J., and Roder, C. A. (2003). *Seagrass-watch: manual for mapping & monitoring seagrass resources* (Cairns: QFS, NFC).

Mckenzie, L. J., Collier, C., Langlois, L., Yoshida, R., Smith, N., and Waycott, M. (2015). *Reef rescue marine monitoring program - inshore seagrass, annual report for the sampling period 1st June 2013 – 31st may 2014* (Cairns: TropWATER, James Cook University).

Mckenzie, L. J., Collier, C. J., Langlois, L. A., Yoshida, R. L., Uusitalo, J., and Waycott, M. (2021). *Marine monitoring program: annual report for inshore seagrass monitoring 2019–20* (Townsville, Australia: James Cook University).

Mckenzie, L. J., Collier, C. J., Langlois, L. A., Yoshida, R. L., and Waycott, M. (2022a). *Marine monitoring program: annual report for inshore seagrass monitoring 2020–21. report for the great barrier reef marine park authority* (Townsville: Great Barrier Reef Marine Park Authority).

Mckenzie, L. J., Langlois, L. A., and Roelfsema, C. M. (2022b). Improving approaches to mapping seagrass within the great barrier reef: from field to spaceborne earth observation. *Remote Sens.* 14, 2604. doi: 10.3390/rs14112604

Mckenzie, L. J., Nordlund, L. M., Jones, B. L., Cullen-Unsworth, L. C., Roelfsema, C., and Unsworth, R. K. F. (2020). The global distribution of seagrass meadows. *Environ. Res. Lett.* 15, 074041. doi: 10.1088/1748-9326/ab7d06

Mckenzie, L. J., Roder, C. A., Roelofs, A. J. , and Lee Long, W. J. (2000). "Post-flood monitoring of seagrasses in hervey bay and the great sandy strait 1999: implications for dugong, turtle and fisheries management," in *Department of primary industries information series QI00059* (Cairns: Queensland Department of Primary Industries, NFC).

Moniruzzaman, M., Islam, S. M. S., Lavery, P., and Bennamoun, M. (2019). "Faster r-CNN based deep learning for seagrass detection from underwater digital images," in *2019 digital image computing: techniques and applications (DICTA)* (Perth, WA, Australia), 1–7. doi: 10.1109/DICTA47822.2019.8946048

Mount, R., Bricher, P., and Newton, J. (2007). *National Intertidal/Subtidal benthic (NISB) habitat classification scheme, version 1.0, October 2007* (Hobart, Tasmania: National Land & Water Resources Audit & School of Geography and Environmental Studies, University of Tasmania).

Noman, M. K., Islam, S. M. S., Abu-Khalaf, J., and Lavery, P. (2021). "Multi-species seagrass detection using semi-supervised learning," in *2021 36th International Conference on Image and Vision Computing New Zealand (IVCNZ)* (Tauranga, New Zealand). 1–6. doi: 10.1109/IVCNZ54163.2021.9653222

Preen, A. R., Lee Long, W. J., and Coles, R. G. (1995). Flood and cyclone related loss, and partial recovery, of more than 1000 km2 of seagrass in hervey bay, Queensland, Australia. *Aquat. Bot.* 52, 3–17. doi: 10.1016/0304-3770(95)00491-H

Raine, S., Marchant, R., Moghadam, P., Maire, F., Kettle, B., and Kusy, B. (2020). Multi-species seagrass detection and classification from underwater images. *Computer Vision and Pattern Recognition*. doi: 10.1109/DICTA51227.2020.9363371

Reus, G., Möller, T., Jäger, J., Schultz, S. T., Kruschel, C., Hasenauer, J., et al. (2018). "Looking for seagrass: deep learning for visual coverage estimation," in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)* (Kobe, Japan). 1–6. doi: 10.1109/OCEANSKOBE.2018.8559302

Schröder, S.-M., Kiko, R., and Koch, R. (2020). MorphoCluster: efficient annotation of plankton images by clustering. *arXiv*. doi: 10.48550/arxiv.2005.01595

Seagrass-Watch (2022) *Hervey bay* (Clifton Beach: Seagrass-Watch Ltd). Available at: https://www.seagrasswatch.org/burnettmary/#HV (Accessed 22 June 2022).

Sheaves, M., Bradley, M., Herrera, C., Mattone, C., Lennard, C., Sheaves, J., et al. (2020). Optimizing video sampling for juvenile fish surveys: using deep learning and evaluation of assumptions to produce critical fisheries parameters. *Fish Fish.* 21, 1259–1276. doi: 10.1111/faf.12501

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958. doi: 10.5555/2627435.2670313

Tammina, S. (2019). Transfer learning using VGG-16 with deep convolutional neural network for classifying images. *Int. J. Sci. Res. Publications (IJSRP)* 9(10). doi: 10.29322/IJSRP.9.10.2019.p9420

Tieleman, T., and Hinton, G. (2014)RMSprop gradient optimization. Available at: http://www.cs.toronto.edu/tijmen/csc321/slides/lecture_slides_lec6.pdf.

Zhang, X., Zou, J., He, K., and Sun, J. (2015). Accelerating very deep convolutional networks for classification and detection. *Computer Vision and Pattern Recognition*. doi: 10.48550/arXiv.1505.06798

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 19 July 2023
DOI 10.3389/fmars.2023.1149895

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Salah Bourennane,
Centrale Marseille, France
Syed Agha Hassnain Mohsan,
Zhejiang University, China

*CORRESPONDENCE
Min Fu
fumin@ouc.edu.cn

RECEIVED 23 January 2023
ACCEPTED 27 June 2023
PUBLISHED 19 July 2023

CITATION
Huo H, Fu M, Liu X and Zheng B (2023) DCC-GAN-based channel emulator for underwater wireless optical communication systems.
Front. Mar. Sci. 10:1149895.
doi: 10.3389/fmars.2023.1149895

COPYRIGHT
© 2023 Huo, Fu, Liu and Zheng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

DCC-GAN-based channel emulator for underwater wireless optical communication systems

Huanxin Huo[1], Min Fu[1,2]*, Xuefeng Liu[3] and Bing Zheng[1,2]

[1]College of Electronic Engineering, Ocean University of China, Qingdao, China, [2]Sanya Oceanographic Institution, Ocean University of China, Sanya, China, [3]College of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao, China

The complex and variable oceanic environment challenges channel modeling of Underwater Wireless Optical Communication (UWOC) systems. Most of the classical modeling methods focus mainly on the water environment and ignore the effect of communication equipment on signal transmission, thus making it difficult to model the UWOC channel's complicated characteristics comprehensively. In this work, a UWOC channel emulator based on Deep Convolutional Conditional Generative Adversarial Networks is established and verified to address the challenge, which can effectively learn the characteristics of channel response and generate emulated signals with randomness like a real UWOC system in a practical application environment. Compared with the approaches based on multi-layer perceptron and convolutional neural network, the experimental results of the proposed method indicate outstanding performances in time domain, frequency domain and universality with different turbidity levels, respectively. This approach provides a new idea for applying deep learning techniques to the field of UWOC channel modeling.

KEYWORDS

Underwater Wireless Optical Communication, Generative Adversarial Networks, deep learning, UWOC, GAN, channel modeling

1 Introduction

Nowadays, with the gradual deepening of marine research, there has been a significant increase in underwater activities such as marine environment monitoring, offshore oil exploration, underwater archaeology, and underwater experimental data collection, so a reliable and high transmission rate underwater wireless communication technique is urgently needed (Zeng et al., 2017). Acoustic communication can no longer meet the growing demand for high-speed rates due to its low bandwidth and high transmission delay. Additionally the transmission distance of underwater radio frequency (RF) communication is suppressed because of the skin effect of radio waves (Kaushal and Kaddoum, 2016; Miramirkhani and Uysal, 2018). While utilizing the Underwater Wireless

Optical Communication (UWOC) technique, which features high bandwidth, fast transmission rate, good confidentiality and low cost, is gaining more and more attention and has broad application prospects (Chi et al., 2015).

When light propagates through seawater, photons will randomly collide with water molecules or other particles in seawater and deviate from the original propagation direction, leading to a phenomenon of beam divergence, thus causing a loss of optical power at the receiving end. Meanwhile, the farther the distance of light transmission, the more severe the beam divergence becomes, while the loss of the received optical signal directly affects the communication distance and transmission rate of the UWOC system (Mobley, 1994; Gabriel et al., 2013). Furthermore, the attenuation characteristics of the UWOC channel to the optical signal vary with different marine environmental parameters, such as depth and water quality. The complicated absorption and scattering characteristics of the same channel for various wavelengths of light also vary greatly (Zeng et al., 2017). Therefore, the complexity of the UWOC channel poses considerable difficulties for channel modeling.

In recent years, with the development of theory, optoelectronic technology and the improvement of computer performance, the research on UWOC channels has made remarkable progress. Sermsak Jaruwatanadilok modeled the impulse response of the UWOC channel using vector radiative transfer theory which includes multiple scattering effects and polarization. And the scattering effects were quantified as a function of distance and bit error rate (BER) (Jaruwatanadilok, 2008). Brandon M. Cochenour et al. proposed the Beam-Spread Function (BSF) to estimate the impact of scattering effects on the received signal power in the underwater light propagation process (Cochenour et al., 2008). Chadi Gabriel et al. quantified the UWOC channel impulse response for different water types and link distances using the Monte Carlo approach (Gabriel et al., 2013). Shijian Tang et al. presented a closed-form expression of double Gamma functions to model the UWOC channel impulse response, which fits well with the Monte Carlo simulation results (Tang et al., 2014). Also using numerical Monte Carlo simulations, Sanjay Kumar Sahu and Palanisamy Shanmugam obtained a more accurate UWOC channel model by improving the scattering phase function(Sahu and Shanmugam, 2018).

The aforementioned studies mainly focus on the loss of optical signals during the propagation in different water types. Actually, in the process of signal transmission, it is inappropriate to ignore the effect of optical and electrical devices at the transmitter and receiver ends, such as the dark current noise of the photomultiplier tube (PMT), the impulse response of the electronic amplifier, the nonlinear response of the laser, the errors in digital-to-analog (D/A) or analog-to-digital (A/D) conversion and so on. Therefore, a realistic and reliable UWOC channel model is required to completely capture the effects of all parts of the communication system on signal transmission, which is a complex process that neural networks are ideally suited to emulate. Yiheng Zhao et al. proved the feasibility of utilizing neural networks for UWOC channel modeling by proposing a channel emulator called two tributaries heterogeneous neural network (TTHnet) (Zhao et al., 2019), which is based on a combined design of multi-layer perceptron (MLP) and convolutional neural network (CNN). The 1.2m saltwater channel experiments verified the TTHnet regarding both spectrum and BER mismatch, realizing more accurate performance than other channel emulators.

Generative Adversarial Networks (GAN) (Goodfellow et al., 2014), composed of a generator and a discriminator, is one of the most critical research directions in deep learning. Owing to its outstanding data generation capability, GAN has been widely used in computer vision and natural language processing (Pan et al., 2019). In order to generate samples with specific properties, Mhdi Mirza and Simon Osindero proposed a Conditional Generative Adversarial Network (CGAN), where conditional information is added to guide the GAN generator to generate samples (Mirza and Osindero, 2014). The content and structure of the conditional information can be flexibly changed according to the application scenario. For instance, CGAN has been utilized for image resolution enhancement (Ledig et al., 2017) and semantic segmentation of images (Souly et al., 2017), as well as for generating images from text descriptions (Reed et al., 2016; Liang et al., 2017). Apart from applications in computer vision, previous studies in the field of communication have proved that GAN is an effective approach for channel modeling. Davide Righini et al. proposed an approach to generate channel transfer functions for power line communication using Mixture Generative Adversarial Nets (Hoang et al., 2018), which outperforms traditional modeling methods (Righini et al., 2019). In Ref. (Ye et al., 2020), CGAN was employed to model channel effects in end-to-end wireless communication system, and simulation results show that the CGAN approach is effective in additive white Gaussian noise (AWGN) channels, Rayleigh fading channels, and frequency-selective channels. Yudi Dong et al. also developed a CGAN-based channel estimation method for multiple-input multiple-output (MIMO) mmWave wireless communication systems, which has better robustness and reliability compared with conventional methods and other deep learning methods (Dong et al., 2021).

In this article, a Deep Convolutional Conditional Generative Adversarial Networks (DCC-GAN) method for modeling UWOC channels is developed and experimentally tested at different turbidity waters and various transmission rates. The performance is evaluated by spectrum mismatch, BER mismatch and correlation coefficient. The experimental results show that the generator can generate emulated signals with randomness like the real UWOC channel, proving that our proposed model can learn and analyze the characteristics of the channel well. To the best of our knowledge, this is the first study to apply GAN to emulate UWOC channels, which has great potential for exploration in channel modeling.

The rest of the paper is organized as follows. In Section 2, the theoretical principle of the proposed channel emulator is presented, and then the architecture of DCC-GAN is described in detail. In Section 3, the experimental setup for making UWOC datasets is introduced. In Section 4, a series of experiments are carried out to demonstrate the effectiveness of the proposed method. Finally, a brief conclusion is given in Section 5.

2 The proposed channel emulator based on DCC-GAN

2.1 GAN and CGAN

GAN, as its name implies, is a generative network model for learning data distribution in the way of adversarial training, where the aim is to learn a model that can produce samples close to the target distribution. In this article, the DCC-GAN is applied to model the distribution of the UWOC channel output based on a GAN.

The GAN system consists of two parts, namely the generator G and the discriminator D. The input to the generator is a noise sample z which is subject to a specific prior distribution p_z, e.g., Gaussian distribution. Then, the generator transforms the noise sample z into a generated sample $G(z)$. The discriminator takes either a real sample x from the target distribution p_{data} or a generated sample as input and returns the probability that the input comes from the target distribution rather than the generator. During the training stage, the objective of the discriminator is to learn to distinguish whether the current sample is from the real dataset or the data generated by the generator, while the objective of the generator is to generate fake samples that are as similar as possible to the real samples to fool the discriminator. If the discriminator can successfully distinguish between the two types of samples, then this information is fed back to the generator so that the generator can learn to generate samples more like the real samples. As the number of adversarial training epoch increases, the learning ability of the generator and the discriminating ability of the discriminator become stronger and stronger. Finally, the training progress ends when the discriminator can no longer discriminate between the real samples and the generated fake ones better than random guessing.

Generally, denote the parameter sets of the generator and discriminator as θ_G and θ_D, respectively, the objective functions of the generator and discriminator can be mathematically expressed as follows:

$$L_G = \min_{\theta_G} \quad E_{z\sim p_z(z)}[\log\ (1 - D(G(z)))] \tag{1}$$

$$L_D = \max_{\theta_D} \quad E_{x\sim p_{data}(x)}[\log\ D(x)] + E_{z\sim p_z(z)}[\log\ (1 - D(G(z)))] \tag{2}$$

The objective of G is to maximize the output of D when the input to D is $G(z)$, while the objective of D is to return a high value when the input is a real sample x and a low one when the input is $G(z)$, thus forming an adversarial training mechanism.

As shown in Figure 1, the GAN can be extended to a CGAN model if a conditional information y is imposed on the generator and discriminator. The conditional information attaches constraints to the original GAN so that the generator can generate data under the guidance of the conditional information, which addresses the issue of uncontrollable sample categories generated by the original GAN. Then, the optimization functions of the generator and discriminator become:

$$L_G = \min_{\theta_G} \quad E_{\tilde{x}\sim p_g(\tilde{x})}[\log\ (1 - D(\tilde{x}|y))] \tag{3}$$

$$L_D = \max_{\theta_D} \quad E_{x\sim p_{data}(x)}[\log\ D(x|y)] + E_{\tilde{x}\sim p_g(\tilde{x})}[\log\ (1 - D(\tilde{x}|y))] \tag{4}$$

Where, p_g is the generator model distribution implicitly defined by $\tilde{x} = G(z|y)$, $z \sim p_z(z)$.

CGAN is employed in the proposed UWOC channel emulator to simulate the output signal with the given conditioning information on the transmitted signal.

2.2 Architecture of DCC-GAN

Although the original GAN is a powerful generative model, it always suffers from difficulties in training and poor quality of the generated results. By combining GAN with CNN, Deep

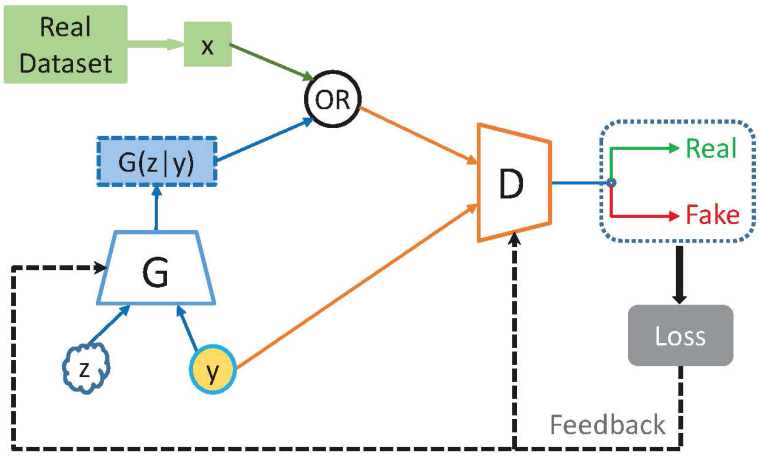


FIGURE 1
Structure of CGAN.

Convolutional Generative Adversarial Networks (DCGAN) (Radford et al., 2016) can significantly improve the quality of the generated results by exploiting the powerful feature extraction ability of two-dimensional convolutional layer and also find an appropriate network structure for stable training by improving CNN, thus remarkably overcoming the shortcomings of the original GAN. In this work, the hierarchical one-dimensional convolutional layers are used to replace the original MLP. Therefore, the proposed method is called "Deep Convolutional Conditional GAN". It is appropriate to employ convolutional layers since convolutional operations between signals can represent the channel response. The specific parameters of DCC-GAN are shown in Table 1, where K denotes the batch size, and the dimension of the noise sample z is 8. Every 100 adjacent signals are sequentially input to the network, with each signal containing 20 sampling points.

2.3 Improvement of the objective function

According to Ref. (Goodfellow et al., 2014), minimizing the original GAN's loss function is equivalent to minimizing the Jensen–Shannon (JS) divergence between the target distribution p_{data} and the generator model distribution p_g, which tends to cause the gradients to vanish when the discriminator saturates. This training difficulty arises because the JS divergence is potentially not continuous for the generator's parameters (Arjovsky et al., 2017). So, the Earth-Mover (also called Wasserstein-1) distance $W(q, p)$ is introduced to replace JS divergence in Wasserstein GAN (Arjovsky et al., 2017), where the discriminator is also called a critic. Using the Kantorovich-Rubinstein duality, the critic loss function can be obtained as Eq. (5) where $\mathbb{Z}$ is the set of 1-Lipschitz functions.

$$\min_{\theta_D \in \mathbb{Z}} \quad L_{critic} = \min_{\theta_D \in \mathbb{Z}} \quad \left\{ E_{\tilde{x} \sim p_g(\tilde{x})}[D(\tilde{x}|y)] - E_{x \sim p_{data}(x)}[D(x|y)] \right\} \tag{5}$$

In this case, minimizing L_{critic} is equivalent to minimizing the Wasserstein-1 distance $W(p_{data}, p_g)$ by optimizing the generator's parameters. Nevertheless, to enforce the Lipschitz constraint on the critic, Wasserstein GAN suggests clipping the weights of the critic to a compact space, which may result in either vanishing or exploding gradients when the clipping threshold is not tuned carefully. To avoid undesirable behaviors, a soft version of the constraint called the Wasserstein GAN Gradient Penalty (WGAN-GP) algorithm (Gulrajani et al., 2017) is proposed as an alternative way to enforce the Lipschitz constraint, and the gradient penalty metric L_{gp} is defined as:

$$L_{gp} = \lambda E_{\hat{x} \sim p_{\hat{x}}(\hat{x})}[(\nabla_{\hat{x}} D(\hat{x}|y)_2 - 1)^2] \tag{6}$$

Where, $p_{\hat{x}}$ is defined as sampling uniformly along straight lines between pairs of points sampled from p_{data} and p_g, λ denotes penalty coefficient.

In this article, the WGAN-GP algorithm is introduced to improve the training instability of DCC-GAN. The objective function of D is then reformulated as a combination of critic loss and gradient penalty metric, which is described as:

$$L_D = \min_{\theta_D} \quad \{L_{critic} + L_{gp}\} \tag{7}$$

And the objective function of G is modified as:

$$L_G = \min_{\theta_G} \quad E_{\tilde{x} \sim p_g(\tilde{x})}[-D(\tilde{x}|y)] \tag{8}$$

In the experiments, λ is set to 5, which works well on the proposed DCC-GAN and the UWOC datasets.

TABLE 1 Model parameters of DCC-GAN.

Type of layer	Activation function	Kernel size	Output shape
Generator			
Input	–	–	K × 100 × (20 + 8)
Conv1D	ReLU	5	K × 100 × 64
Conv1D	ReLU	3	K × 100 × 32
Conv1D	ReLU	3	K × 100 × 16
Conv1D	Tanh	3	K × 100 × 20
Discriminator			
Input	–	–	K × 100 × (20 + 20)
Conv1D	ReLU	5	K × 100 × 64
Conv1D	ReLU	3	K × 100 × 32
Conv1D	ReLU	3	K × 100 × 16
Conv1D	–	3	K × 100 × 8
Dense	ReLU	–	64
Dense	–	–	1

2.4 Training details of DCC-GAN

The improvement in training instability not only allows us to enhance sample quality by experimenting with a broader range of network architectures but also requires little hyperparameter tuning. The training procedure of DCC-GAN is illustrated in **Algorithm 1** in detail. The training process aims to obtain an ideal generator architecture, which can model the distribution of the UWOC channel output, that is, to realize the function of the channel emulator. In each iteration, the generator and discriminator training processes are carried out alternately. When one model is trained, the other one is fixed. The real data can be obtained from the transmitted signal through the real channel, while the fake data is obtained from the transmitted signal through the generator. The loss function of Eq. (8) is utilized to update the generator's parameters. The real, fake, and true-fake joint distribution data are fed into the discriminator, respectively, with the transmitted signal as conditional information. The parameters of the discriminator are updated according to the loss function of

Eq. (7). Both models are optimized by the Adam optimizer using stochastic gradient descent, with an initial learning rate α of 0.0002. The initial hyperparameter values in **Algorithm 1** are derived from a previous study on RF channel modeling(Ye et al., 2020). In order to ensure that the algorithm can handle a wide range of input data while still converging within a reasonable number of epochs, extensive experimentation is carried out to fine-tune these values. Ultimately, experimental results indicate that setting λ to 5 and using a batch size m of 20 yield the best performance. Assuming that k represents the number of discriminator iterations per generator iteration, the optimal convergence can be obtained in the experiments when k is set to 6. The number of training epochs is set to at least 200.

Require: The number of discriminator iterations per generator iteration k, the batch size m, the gradient penalty coefficient λ and the learning rate α.

for number of training epochs **do**
  for number of training iterations **do**
    for k steps **do**
      for $i = 1, \ldots, m$ **do**
        Sample real signal $x \sim p_{data}(x)$, noise variable $z \sim p_z(z)$
        Sample a random number $\delta \sim U[0,1]$
        Get the transmitted signal y as condition information.
        $\tilde{x} \leftarrow G(z|y)$
        $\hat{x} \leftarrow \delta x + (1-\delta)\tilde{x}$
        $L^{(i)} \leftarrow D(\tilde{x}|y) - D(x|y) + \lambda(\nabla_{\hat{x}} D(\hat{x}|y)_2 - 1)^2$
      end for
      $\theta_D \leftarrow \mathrm{Adam}(\nabla_{\theta_D} \frac{1}{m}\sum_{i=1}^{m} L^{(i)})$
    end for
    Sample minibatch of noise variables $\{z^{(i)}\}_{i=1}^{m} \sim p_z(z)$
    Sample minibatch of transmitted signal $\{y^{(i)}\}_{i=1}^{m}$
    $\{\tilde{x}^{(i)}\}_{i=1}^{m} \leftarrow \{G(z^{(i)}|y^{(i)})\}_{i=1}^{m}$
    $\theta_G \leftarrow \mathrm{Adam}(\nabla_{\theta_G} \frac{1}{m}\sum_{i=1}^{m} - D(\tilde{x}|y))$
  end for
end for

ALGORITHM 1 Minibatch stochastic gradient descent training of DCC-GAN. Assume the generator parameter θ_G and the discriminator parameter θ_D. The default values of k=6, m=20, λ=5 and α=0.0002 are used.

3 Experimental setup and details

This section describes the experimental procedure for making the UWOC dataset in detail. To simulate the characteristics of the UWOC channels, a 35 meters underwater laser communication system was built. The entire experimental setup of the UWOC system is shown in Figure 2. The main components of the UWOC system are shown in Figure 3.

3.1 Experimental setup

3.1.1 Transmitter

As shown in Figure 3B, a semiconductor laser (OXXIUS, LaserBoxx-488) with an emission peak wavelength of 488 nm is employed at the transmitter end, which meets the requirements of blue-green light in the 450 nm to 550 nm band where the attenuation of seawater is much less than that of other wavelengths(Duntley, 1963). The laser has built-in driver circuitry, which can accept analog signal input directly and adjust the emitted optical power according to the application scenario.

3.1.2 Establishment of UWOC channel

The underwater channel for light transmission is built in a 5 m × 1 m × 1 m water tank, shown in Figure 3A, which is filled with clear tap water or artificial turbid water. The communication distance of 35 m is achieved by using six reflective mirrors fixed to the tank's inner walls on both sides by cardan joints. The reflection of each 5 m light path is realized by fine-tuning the angle of the mirrors. A total propagation distance of 35 m (5 m × 7) can be obtained, through six reflections, as shown in Figure 3D.

3.1.3 Receiver

As shown in Figure 3C, a PMT (Hamamatsu, R1527) is employed at the receiver end, which has a spectral response in the range 185 nm-680 nm, with an optimal spectral response of about 400 nm, and works well with the 488 nm laser. Although the PMT has the advantages of low noise and high gain, the output current is still shallow, so a signal amplifier unit (AMP) (Hamamatsu, C11184) is needed. They are powered by high voltage and DC power, respectively.

3.2 Experimental procedure

An integral experimental setup is demonstrated in Figure 2. At first, the non-return-to-zero on-off-keying (NRZ-OOK) modulated signals are loaded into an arbitrary waveform generator (AWG) as the transmitted signal. Then, the AWG performs D/A conversion of the signals and outputs analog electrical signals to drive the laser for intensity modulation. Thus, optical signals for underwater transmission can be generated. After passing through a 35m underwater channel, the PMT detects and the AMP amplifies the optical signals, and then the received photons are converted back into electrical signals. Afterward, a memory oscilloscope (OSC) is employed to sample and record the corresponding digital signals. Finally, the offline processing operations are performed, including synchronization, demodulation, and the BER calculation. Meanwhile, the synchronized signals are adopted as the received signal, and the received signal and the corresponding transmitted signal are collected to make the UWOC dataset.

According to the setup of the network architecture in Figure 2, the received and the corresponding transmitted signals are combined to train the proposed DCC-GAN. In detail, as conditional information, the transmitted signal is fed into the generator along with a noise sample. Then, the generator outputs a fake received

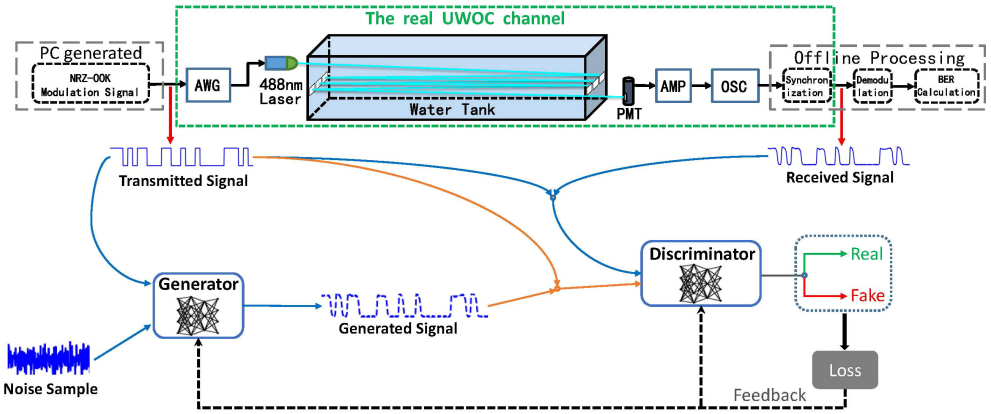


FIGURE 2
Experimental setup and block diagram of the UWOC system and channel emulator.

signal, and the discriminator will decide whether the input signal is a real signal or a fake one from the generator under the guidance of the transmitted signal. The generator and discriminator can find their optimal parameters individually according to the training strategy in **Algorithm 1**. When the training process finishes, the function of emulating the UWOC channel is realized. Finally, to evaluate the model's generalization ability, independent samples from the test set are employed to estimate the trained channel emulator.

This work produces a series of datasets under different experimental conditions, and the experiment parameters are shown in Table 2. Besides the tap water channel, two artificial turbid water channels are also created by adding a specific quantity of Aluminum Hydroxide ($Al(OH)_3$), which is commonly used as a scattering agent. The attenuation coefficient at wavelength 488 nm is measured to be 0.1169 m^{-1} (tap water), 0.2318 m^{-1} and 0.471 m^{-1} in three types of water, respectively. Correspondingly, the transmitting optical powers are also finely adjusted to obtain the optimal received signal.

4 Experiment results and analysis

In this section, the performance of the proposed channel emulator is demonstrated and analyzed concerning different types of water and various transmission rates. In detail, metrics such as

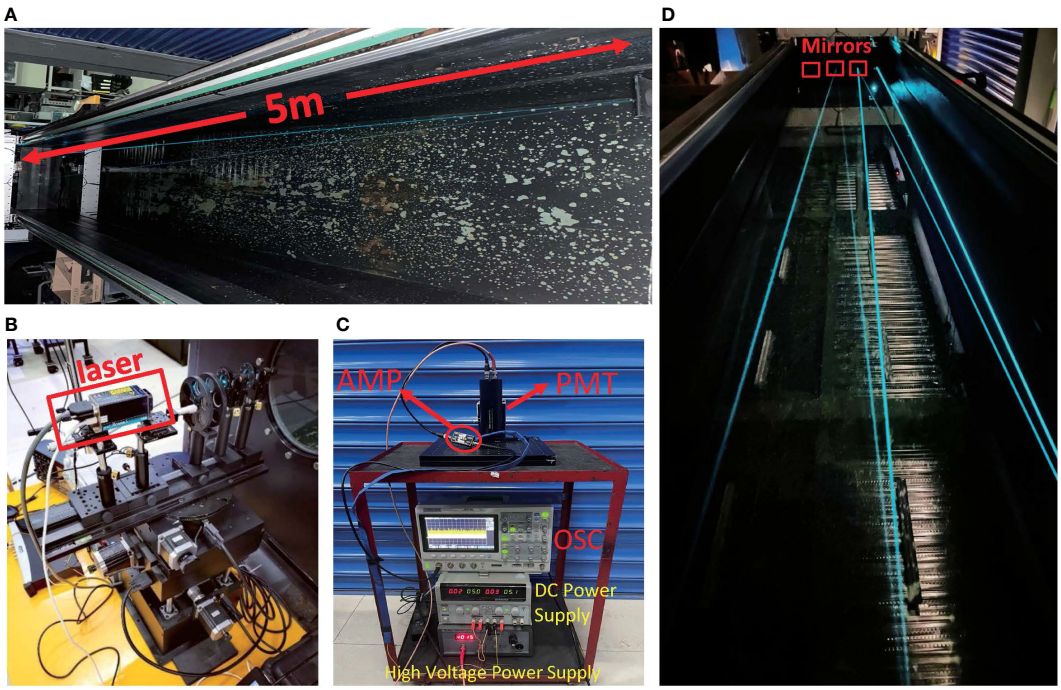


FIGURE 3
Components of the UWOC system. **(A)** Water tank. **(B)** Transmitter: 488nm laser. **(C)** Receiver: photomultiplier tube (PMT), amplifier unit (AMP), oscilloscope (OSC), DC power supply and high voltage power supply. **(D)** 35 m blue-green light reflection paths.

TABLE 2 Experimental parameters.

Water Types	Optical Power	Attenuation Coefficient	Bitrate
water type I	0.1 mW	0.1169 m^{-1}	16 ~ 24Mbps
water type II	0.2 mW	0.2318 m^{-1}	16 ~ 24Mbps
water type III	0.3 mW	0.471 m^{-1}	16 ~ 24Mbps

the absolute amplitude spectrum mismatch, Pearson correlation coefficient, and the BER mismatch between each emulated received signal and real received signal are calculated and compared.

Suppose a real received signal and its emulated signal are denoted as X and $\tilde{X}$, respectively. The calculation of absolute amplitude spectrum mismatch is described as:

$$spectrum \quad mismatch(X,\tilde{X}) = abs(FFT(X) - FFT(\tilde{X})) \tag{9}$$

The BER mismatch can be computed as:

$$BER \quad mismatch(X,\tilde{X}) = abs(BER(X) - BER(\tilde{X})) \tag{10}$$

And the correlation coefficient is defined as:

$$Corr(X,\tilde{X}) = \frac{Cov(X,\tilde{X})}{\sqrt{Var(X)Var(\tilde{X})}} \tag{11}$$

Where, $Var(\cdot)$ is the variance, $Cov(X,\tilde{X})$ represents the covariance of X and $\tilde{X}$.

The BER directly reflects the noise intensity of the UWOC channel, so three typical channels with different orders of magnitude of BER, shown in Table 3, are selected to test the performance of DCC-GAN. To compare with conventional neural networks and demonstrate the superiority of the generative adversarial approach, a CNN model is designed with similar complexity. Its specific structure and parameters are shown in Table 4. Furthermore, an MLP model is also introduced from a previous study (Ye et al., 2017) related to wireless channel estimation, which contains five layers and neurons in each layer are 200,500,250,120 and 200, respectively. The activation function of each layer is ReLU, except for no activation function in the last layer. In both models, the Adam optimizer updates the weights and the loss function is Mean Square Error (MSE), the batch size for training is 20 and the learning rate is 0.001.

4.1 Performance comparison in the time domain

In engineering applications, the Pearson correlation coefficient is often used to measure the similarity between signal sequences (Ahmed, 2015). The correlation coefficient value lies from -1 to 1, where 1 represents perfect correlation, while -1 shows a negative correlation and 0 indicates no correlation. Figure 4 shows the comparison of correlation coefficients of the signals generated by the three neural network-based channel emulators in channel-1, channel-2 and channel-3, respectively. During the training process, the convergence rate of DCC-GAN is similar to CNN and faster than MLP. After 200 training epochs, the correlation coefficient of DCC-GAN is 0.99 in all cases, while the values of CNN and MLP are 0.95, 0.96, 0.95 and 0.83, 0.86, 0.84 in channel-1, channel-2 and channel-3, respectively. Figure 5 shows the BER mismatch performance comparison of the three network models in three channels. From the figure, the BER mismatch performance of DCC-GAN is much better than MLP and CNN, where the BER mismatches of DCC-GAN-based channel emulator are 6.7%, 10.4%, 9.3% of the CNN-based method and only 0.03%, 0.16%, 0.82% of the MLP-based method in channel-1, channel-2 and channel-3, respectively. It indicates that the signal waveform generated by the DCC-GAN-based channel emulator is the closest to the real signal.

Due to the bandwidth limitation of the electro-optical devices, the nonlinear distortion of the UWOC system increases with the transmission rate. To estimate the proposed method at various transmission rates, the BER versus transmission rate curves are demonstrated in Figure 6A. When the transmission rate increases, the BER curves of the emulated and real received signals exhibit a same trend towards a specific increase. The maximum BER mismatch of the DCC-GAN-based channel emulator is 0.4 dB at various transmission rates. Hence the proposed channel emulator is generally applicable to the variation of transmission rate.

In addition to bandwidth limitation, inappropriate working point settings of optoelectronic devices can also cause system errors. At the receiver, the supply voltage directly affects the PMT's dark current noise and gain performance, prompting the need for an appropriate working point to optimize the output signal-to-noise ratio for each application scenario. Figure 6B displays a comparison of BER between the real signal of the UWOC system and the emulated signal generated by DCC-GAN at various PMT working points. Results show that as the supply voltage gradually increases, the BER trend of the simulated signal aligns well with that of the real signal, with the optimal voltage

TABLE 3 Parameters of three typical channels.

Channel Status	Water Types	Bitrate (Mbps)	BER
channel-1	water type III	16	4.5×10^{-4}
channel-2	water type II	20	1.83×10^{-3}
channel-3	water type I	24	3.28×10^{-2}

TABLE 4 Model Parameters of CNN.

Type of layer	Activation function	Kernel size/Pool size	Output shape
Input	–	–	K × 100 × 20
Conv1D	ReLU	5	K × 100 × 64
MaxPooling1D	–	2	K × 50 × 64
Conv1D	ReLU	3	K × 50 × 32
MaxPooling1D	–	2	K × 25 × 32
Conv1D	–	3	K × 25 × 16
Flatten	–	–	K × 400
Dense	Tanh	–	K × 64
Dense	–	–	K × 2000

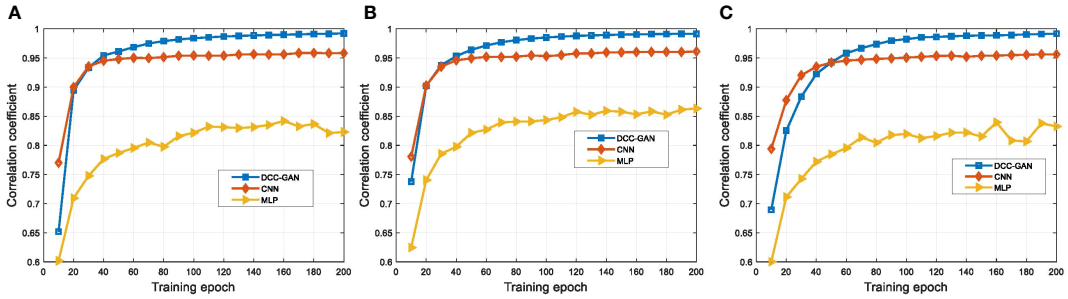


FIGURE 4
The performance of correlation coefficient for different channel emulators on the test set after 200 training epochs in **(A)** channel-1, **(B)** channel-2 and **(C)** channel-3.

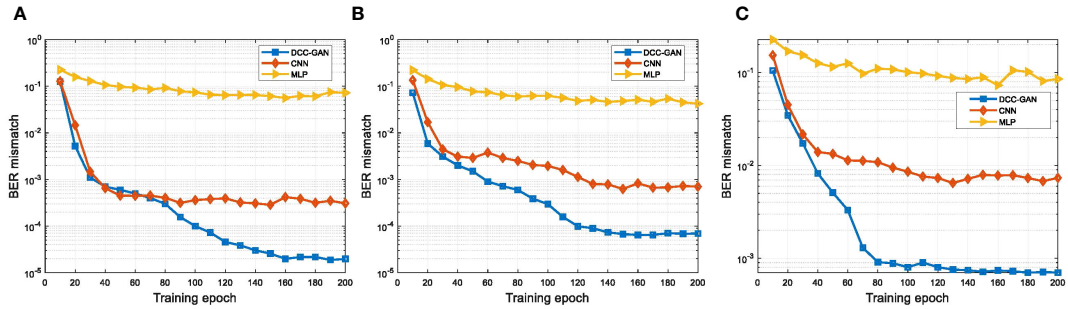


FIGURE 5
The performance of BER mismatch for different channel emulators on the test set after 200 training epochs in **(A)** channel-1, **(B)** channel-2 and **(C)** channel-3.

being around 1000 V. When the voltage exceeds 700 V, the BER mismatch does not surpass 0.72 dB, which signifies the capability of DCC-GAN to effectively capture changes in the channel state due to the nonlinear response of optoelectronic devices.

4.2 Comparison in the frequency domain

Features in the frequency domain can show some phenomena that cannot be found in the time domain, so some experiments are carried out to compare the spectrum of the signals. The results generated by three neural network-based channel emulators and the real received signal in channel-1 are shown in Figure 7A. The absolute mismatches of magnitude between each emulated spectrum and the real one are shown in Figure 7B for a more transparent demonstration, where the average mismatches of MLP, CNN and DCC-GAN are 2.56, 1.14 and 0.25 dB, respectively. Figures 8, 9 show the comparison of the spectrum performance for three channel emulators in channel-2 and channel-3, respectively. The average spectrum mismatches are 2.38, 0.89 and 0.24 dB in channel-2, 2.53, 0.97 and 0.34 dB in channel-3, respectively.

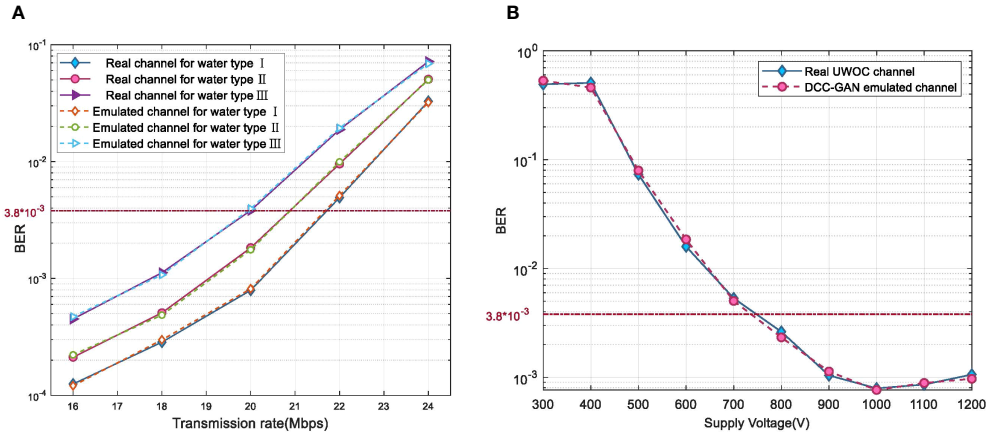


FIGURE 6
BER comparison of real received signal and emulated signal at various **(A)** transmission rates and **(B)** PMT working points.

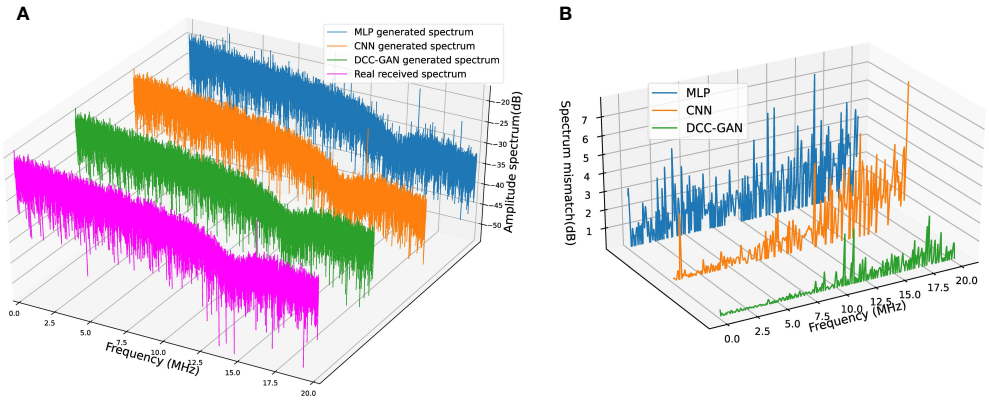


FIGURE 7
(A) Spectrum comparison of real received signal and signals generated by three channel emulators in channel-1. **(B)** Corresponding spectrum mismatch between the real spectrum and the generated spectrum.

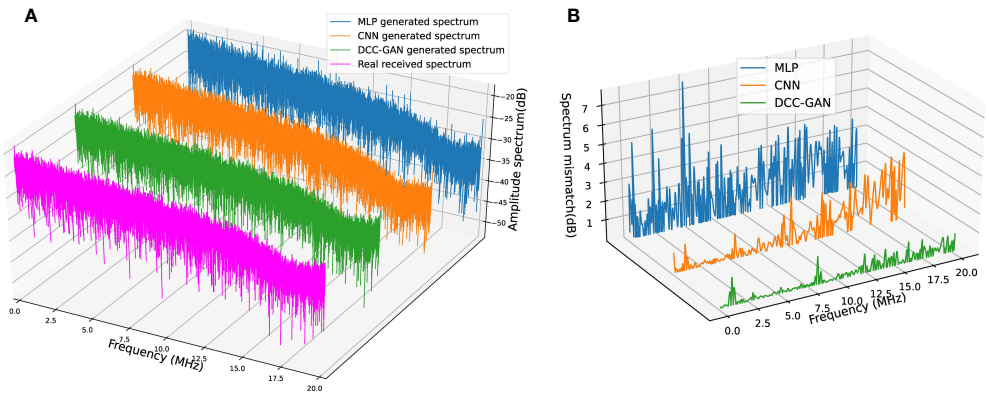


FIGURE 8
(A) Spectrum comparison of real received signal and signals generated by three channel emulators in channel-2. **(B)** Corresponding spectrum mismatch between the real spectrum and the generated spectrum.

10.3389/fmars.2023.1149895

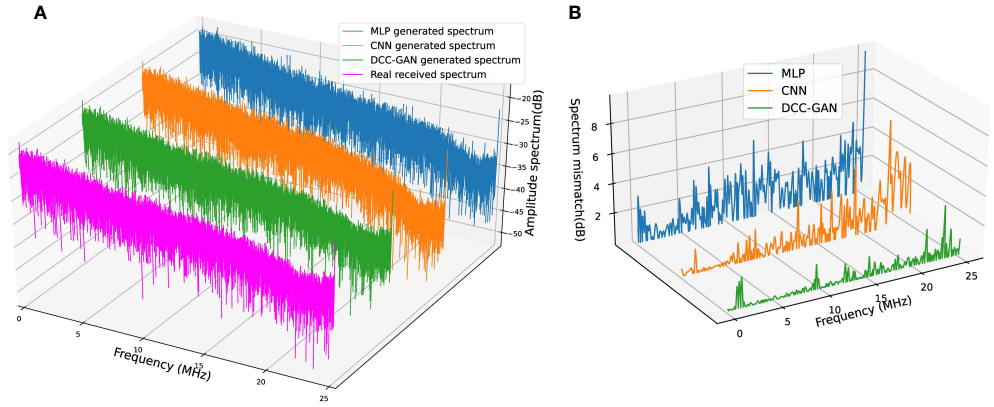


FIGURE 9
(A) Spectrum comparison of real received signal and signals generated by three channel emulators in channel-3. **(B)** Corresponding spectrum mismatch between the real spectrum and the generated spectrum.

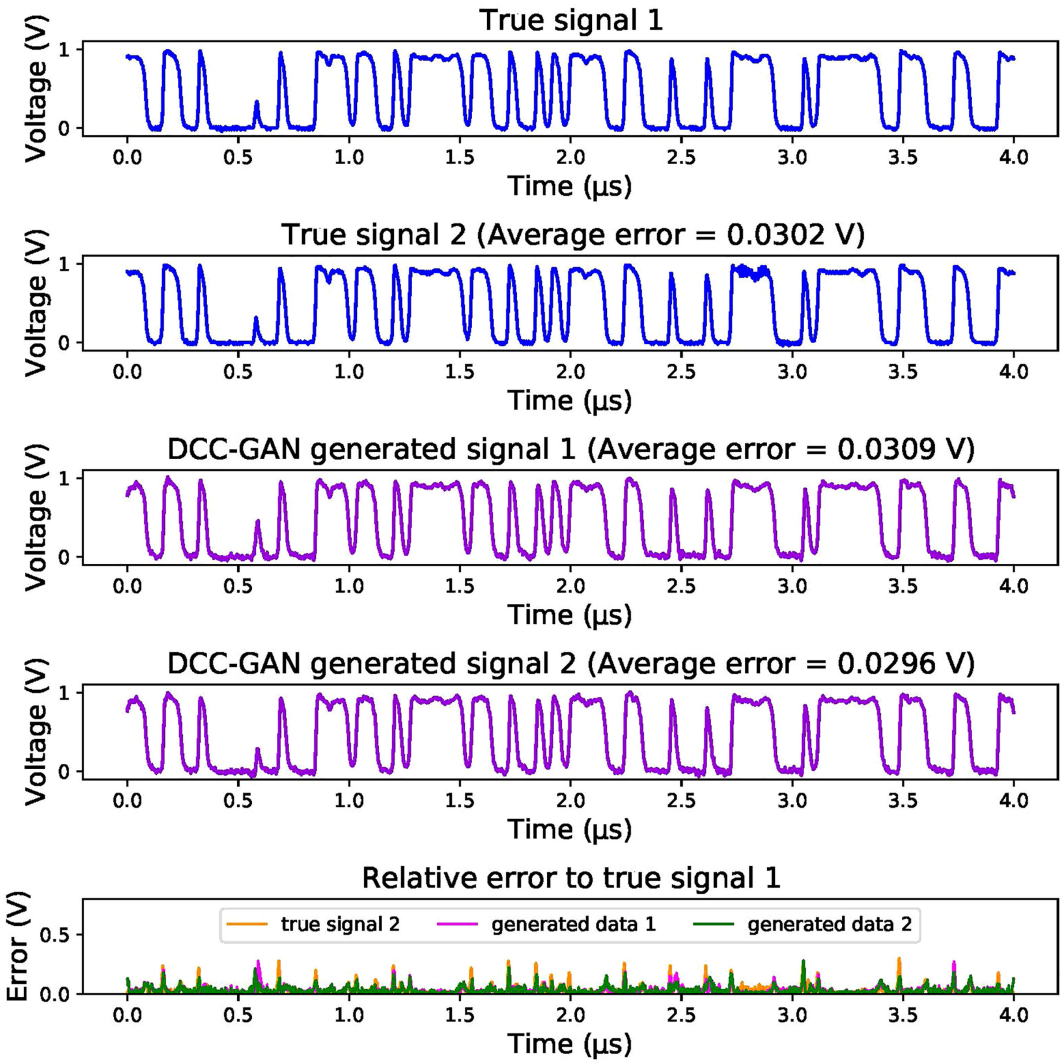


FIGURE 10
Comparison of DCC-GAN generated samples and real samples.

From the mismatch curves in Figures 7B, 8B, and 9B, it can be noticed that the mismatch of MLP is high at all frequency bands, the mismatch of CNN is mainly clustered in the higher frequency bands, while the mismatch of DCC-GAN is lower than the others in all bands, and inevitable the lowest average spectrum mismatch. Obviously, the spectrum generated by the proposed emulator is the closest one to the real signal in all experimental channels. Therefore, the DCC-GAN-based channel emulator can more accurately capture the characteristics of different channels in the frequency domain.

4.3 Discussion on other advantages of DCC-GAN

Benefiting from the diversity of samples generated by GAN, the DCC-GAN-based channel emulator has the additional capability to simulate the randomness of the received signal. This means that the emulated signal generated by the proposed model will not be identical each time for the same input signal, just like an actual receive procedure. For example, if a certain digital sequence is transmitted twice, two signals with slight random differences will be received. In Figure 10, these two real signals are named true signal 1 and true signal 2. Then, the same signal is fed into the channel emulator multiple times to check the differences. Two signals of simulations are selected to compare with the two real received signals. For better visualization, the relative errors of each signal to the true signal 1 are also displayed, and the average error are 0.0302, 0.0309 and 0.0296 V, respectively. The three error curves are not identical to each other, and the average error between the generated signal and the real signal is quite close to the average error between the real signals, indicating that the generated signals have similar random characteristics to the real signal.

Kernel density estimation (KDE) is a non-parametric estimation method which is commonly used in statistics to estimate the probability density function of a random variable(O'Brien et al., 2016). Based on the KDE approach, the comparisons of the data distribution generated by MLP, CNN and DCC-GAN with the real signal are shown in Figures 11A–C, respectively. The real data satisfies a bimodal distribution with a peak-to-peak distance of 0.956 and two half-peak widths of 0.117 and 0.125. For the above three properties, the errors of the data distribution generated by DCC-GAN are only 0.001, 0.001 and 0.002, respectively, while the values of CNN and MLP are 0.039, 0.128, 0.096 and 0.381, 0.483, 0.542, respectively. So it can be clearly observed that the distribution generated by DCC-GAN converges most approximately to the real distribution.

The above experimental results reveal that the emulated signal and the real signal share highly similar characteristics, the DCC-GAN-based channel emulator can not only learn the channel distribution accurately, but also output the emulated signal with randomness to restore the UWOC channel more realistically.

5 Conclusion

This paper proposes a novel DCC-GAN-based model to emulate the UWOC channel more realistically, which combines the advantages

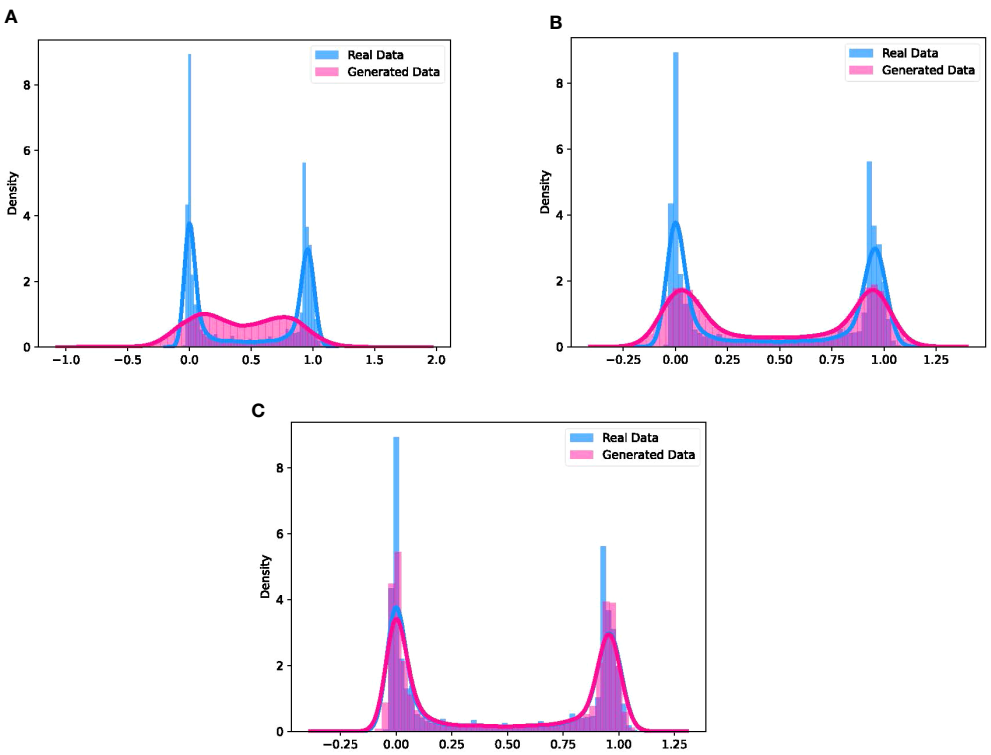


FIGURE 11
Comparison between the real distribution of UWOC channel and the distribution of data generated by **(A)** MLP, **(B)** CNN and **(C)** DCC-GAN.

of CGAN, DCGAN and WGAN-GP algorithms to achieve high-quality generated results and stable training. A series of evaluation experiments regarding the spectrum, correlation coefficient and BER have verified the universality of the proposed channel emulator on different water channels and various transmission rates. The results indicate the effectiveness of DCC-GAN by demonstrating superior performance in both time and frequency domains compared with MLP and CNN-based approaches. Besides, the proposed model can learn the distribution of channel output more realistically to restore the underwater communication signal. The trained model can be used offline to generate diverse signal samples for subsequent experimental analysis, which will offer significant savings on experimental costs and effectively expedite the research advance of the UWOC systems. Therefore, this study opens a promising way to apply deep learning techniques in the UWOC channel modeling field.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

HH and MF contributed to conception and design of the study. HH performed the experimental analysis and wrote the first draft of the manuscript. MF contributed to the manuscript revision. MF, XL, and BZ provided guidance and funding for this research. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the Key Research and Development Project of Hainan Province (No.ZDYF2022GXJS001), Shandong Provincial Natural Science Foundation Grant (ZR2020MF011) and a grant from the National Natural Science Foundation of China (No.61971253).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ahmed, S. N. (2015). "9 - essential statistics for data analysis," in *Physics and engineering of radiation detection, 2nd ed*. Ed. S. N. Ahmed (Amsterdam, The Netherlands: Elsevier), 541–593. doi: 10.1016/B978-0-12-801363-2.00009-7

Arjovsky, M., Chintala, S., and Bottou, L. (2017). "Wasserstein generative adversarial networks," in *INTERNATIONAL CONFERENCE ON MACHINE LEARNING, VOL 70, eds. D. Precup and Y. Teh (1269 LAW ST, SAN DIEGO, CA, UNITED STATES: JMLR-JOURNAL MACHINE LEARNING RESEARCH), vol. 70 of Proceedings of Machine Learning Research. 34th International Conference on Machine Learning*, Sydney, AUSTRALIA, AUG 06-11, 2017.

Chi, N., Haas, H., Kavehrad, M., Little, T. D., and Huang, X.-L. (2015). Visible light communications: demand factors, benefits and opportunities [guest editorial]. *IEEE Wireless Commun.* 22, 5–7. doi: 10.1109/MWC.2015.7096278

Cochenour, B. M., Mullen, L. J., and Laux, A. E. (2008). Characterization of the beam-spread function for underwater wireless optical communications links. *IEEE J. Oceanic Eng.* 33, 513–521. doi: 10.1109/JOE.2008.2005341

Dong, Y., Wang, H., and Yao, Y.-D. (2021). Channel estimation for one-bit multiuser massive mimo using conditional gan. *IEEE Commun. Lett.* 25, 854–858. doi: 10.1109/LCOMM.2020.3035326

Duntley, S. Q. (1963). Light in the sea*. *J. Opt. Soc Am.* 53, 214–233. doi: 10.1364/JOSA.53.000214

Gabriel, C., Khalighi, M.-A., Bourennane, S., Léon, P., and Rigaud, V. (2013). Monte-Carlo-based channel characterization for underwater optical communication systems. *J. Opt. Commun. Netw.* 5, 1–12. doi: 10.1364/JOCN.5.000001

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). "Generative adversarial nets," in *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 27 (NIPS 2014)*, vol. 27 . Eds. Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Weinberger (10010 NORTH TORREY PINES RD, LA JOLLA, CALIFORNIA 92037 USA: NEURAL INFORMATION PROCESSING SYSTEMS (NIPS), 2672–2680.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. (2017). "Improved training of wasserstein gans," in *Proceedings of the 31st international conference on neural information processing systems* (Red Hook, NY, USA: Curran Associates Inc), 5769–5779. NIPS'17.

Hoang, Q. M., Nguyen, T. D., Le, T., and Phung, D. Q. (2018). "Mgan: training generative adversarial nets with multiple generators," in *International Conference on Learning Representations*.

Jaruwatanadilok, S. (2008). Underwater wireless optical communication channel modeling and performance evaluation using vector radiative transfer theory. *IEEE J. Selected Areas Commun.* 26, 1620–1627. doi: 10.1109/JSAC.2008.081202

Kaushal, H., and Kaddoum, G. (2016). Underwater optical wireless communication. *IEEE Access* 4, 1518–1547. doi: 10.1109/ACCESS.2016.2552538

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., et al. (2017). "Photo-realistic single image super-resolution using a generative adversarial network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 105–114. doi: 10.1109/CVPR.2017.19

Liang, X., Hu, Z., Zhang, H., Gan, C., and Xing, E. P. (2017). "Recurrent topic-transition gan for visual paragraph generation," in *2017 IEEE International Conference on Computer Vision (ICCV)*. 3382–3391. doi: 10.1109/ICCV.2017.364

Miramirkhani, F., and Uysal, M. (2018). Visible light communication channel modeling for underwater environments with blocking and shadowing. *IEEE Access* 6, 1082–1090. doi: 10.1109/ACCESS.2017.2777883

Mirza, M., and Osindero, S. (2014). Conditional generative adversarial nets. *ArXiv*. doi: 10.48550/arXiv.1411.1784

Mobley, C. (1994). *Light and water: radiative transfer in natural waters*.

O'Brien, T. A., Kashinath, K., Cavanaugh, N. R., Collins, W. D., and O'Brien, J. P. (2016). A fast and objective multidimensional kernel density estimation method: fastkde. *Comput. Stat Data Anal.* 101, 148–160. doi: 10.1016/j.csda.2016.02.014

Pan, Z., Yu, W., Yi, X., Khan, A., Yuan, F., and Zheng, Y. (2019). Recent progress on generative adversarial networks (gans): a survey. *IEEE Access* 7, 36322–36333. doi: 10.1109/ACCESS.2019.2905015

Radford, A., Metz, L., and Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*. doi: 10.48550/arXiv.1511.06434

Reed, S., Akata, Z., Mohan, S., Tenka, S., Schiele, B., and Lee, H. (2016). "Learning what and where to draw," in *Proceedings of the 30th international conference on neural information processing systems* (Red Hook, NY, USA: Curran Associates Inc), 217–225. NIPS'16.

Righini, D., Letizia, N. A., and Tonello, A. M. (2019). "Synthetic power line communications channel generation with autoencoders and gans," in *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 1–6. doi: 10.1109/SmartGridComm.2019.8909700

Sahu, S. K., and Shanmugam, P. (2018). A theoretical study on the impact of particle scattering on the channel characteristics of underwater optical communication system. *Optics Commun.* 408, 3–14. doi: 10.1016/j.optcom.2017.06.030

Souly, N., Spampinato, C., and Shah, M. (2017). "Semi supervised semantic segmentation using generative adversarial network," in *2017 IEEE International Conference on Computer Vision (ICCV)*. 5689–5697. doi: 10.1109/ICCV.2017.606

Tang, S., Dong, Y., and Zhang, X. (2014). Impulse response modeling for underwater wireless optical communication links. *IEEE Trans. Commun.* 62, 226–234. doi: 10.1109/TCOMM.2013.120713.130199

Ye, H., Li, G. Y., and Juang, B. (2017). Power of deep learning for channel estimation and signal detection in ofdm systems. *IEEE Wireless Communication Lett.* PP, 114–117. doi: 10.1109/LWC.2017.2757490

Ye, H., Liang, L., Li, G. Y., and Juang, B.-H. (2020). Deep learning-based end-to-end wireless communication systems with conditional gans as unknown channels. *IEEE Trans. Wireless Commun.* 19, 3133–3143. doi: 10.1109/TWC.2020.2970707

Zeng, Z., Fu, S., Zhang, H., Dong, Y., and Cheng, J. (2017). A survey of underwater optical wireless communications. *IEEE Commun. Surveys Tutorials* 19, 204–238. doi: 10.1109/COMST.2016.2618841

Zhao, Y., Zou, P., Yu, W., and Chi, N. (2019). Two tributaries heterogeneous neural network based channel emulator for underwater visible light communication systems. *Opt. Express* 27, 22532–22541. doi: 10.1364/OE.27.02253211

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 01 August 2023
DOI 10.3389/fmars.2023.1182653

Check for updates

OPEN ACCESS

EDITED BY
Haiyong Zheng,
Ocean University of China, China

REVIEWED BY
Farook Sattar,
University of Victoria, Canada
Wei Huang,
Ocean University of China, China

*CORRESPONDENCE
Jian Xu
jian.xu@tju.edu.cn

RECEIVED 09 March 2023
ACCEPTED 17 July 2023
PUBLISHED 01 August 2023

CITATION
Jin K, Xu J, Zhang X, Lu C, Xu L and Liu Y (2023) An acoustic tracking model based on deep learning using two hydrophones and its reverberation transfer hypothesis, applied to whale tracking.
Front. Mar. Sci. 10:1182653.
doi: 10.3389/fmars.2023.1182653

COPYRIGHT
© 2023 Jin, Xu, Zhang, Lu, Xu and Liu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

An acoustic tracking model based on deep learning using two hydrophones and its reverberation transfer hypothesis, applied to whale tracking

Kangkang Jin, Jian Xu*, Xuefeng Zhang, Can Lu, Luochuan Xu and Yi Liu

School of Marine Science and Technology, Tianjin University, Tianjin, China

Acoustic tracking of whales' underwater cruises is essential for protecting marine ecosystems. For cetacean conservationists, fewer hydrophones will provide more convenience in capturing high-mobility whale positions. Currently, it has been possible to use two hydrophones individually to accomplish direction finding or ranging. However, traditional methods only aim at estimating one of the spatial parameters and are susceptible to the detrimental effects of reverberation superimposition. To achieve complete whale tracking under reverberant interference, in this study, an intelligent acoustic tracking model (CIAT) is proposed, which allows both horizontal direction discrimination and distance/depth perception by mining unpredictable features of position information directly from the received signals of two hydrophones. Specifically, the horizontal direction is discriminated by an enhanced cross-spectral analysis to make full use of the exact frequency of received signals and eliminate the interference of non-source signals, and the distance/depth direction combines convolutional neural network (CNN) with transfer learning to address the adverse effects caused by unavoidable acoustic reflections and reverberation superposition. Experiments with real recordings show that 0.13 km/MAE is achieved within 8 km. Our work not only provides satisfactory prediction performance, but also effectively avoids the reverberation effect of long-distance signal propagation, opening up a new avenue for underwater target tracking.

KEYWORDS

underwater acoustic target tracking, two hydrophones, cross-spectral analysis, convolutional neural network, transfer learning

1 Introduction

Whales play an extremely important role in the structure and dynamics of natural ecosystems (Roman et al., 2014). They can not only improve primary productivity (Henley et al., 2020), but also regulate carbon dioxide in the atmosphere and marine environment (Roman et al., 2016). Since the moratorium on commercial whaling in 1986, the global whale population has continued grown, with a concomitant increase in the frequency of the whale stranding (Parsons and Rose, 2022), which has attracted widespread attention. In 2020, Klaus pointed out the whale stranding typically occur during their migrations (Vanselow, 2020). Despite several attempts by some scholars to use satellite tags for individual movement behaviors, they still are unable to understand whale movements below the surface, which leaves the potential patterns or causes of whale stranding incompletely expressed (Perez et al., 2022). Therefore, mastering the continuous and high-precision movement trajectories of whales is of great value for the protection of whale diversity and stranding management.

Passive acoustic monitoring (PAM) offers a novel, long-term, large-scale monitoring advantage that can provide species distribution and activity information for vocal species, making it an ideal bioacoustic tool for whale tracking (Davis et al., 2017; Aulich et al., 2019). PAM utilizes a distributed single-receiver hydrophone system, which enables the estimation of cetacean population densities without the need for tracking and directly protecting whales during migration. Currently, there is a growing expectation for tracking systems designed for high-mobility whales to have a smaller design, low power consumption, and fewer hydrophones (Ferreira et al., 2021; Frasier et al., 2021; Cheeseman et al., 2022; Jones et al., 2022). Previous studies have explored the use of two hydrophones to determine the orientation or distance of underwater targets using acoustic-based technology. However, due to the coupling between the azimuth and distance parameters (Ding et al., 2020), the distance estimates expressed according to the analytic equations are poor when the azimuth varies with the interference of reverberation and acoustic reflections, which significantly reduces the tracking accuracy of the whales.

With the increasing development of artificial intelligence, new statistical prediction methods based on deep learning have shown better performance in existing underwater target location prediction. In recent years, more and more deep neural networks have been proposed one after another, such as CNNs (Song, 2018; White et al., 2022), deep neural networks (DNNs) (Yangzhou et al., 2019), recurrent neural networks (RNNs) (Shankar et al., 2020) and transformers (Kujawski and Sarradj, 2022). These models have been successfully applied in many fields of geophysics. Jiang et al. (2020) proposed a new algorithm fusing deep neural network and CNN for sound source orientation using the voltage difference and cross-correlation function extracted from binaural signals. The CNN architecture developed by (White et al., 2022) uses a custom image input to exploit the temporal and frequency domain feature differences between each sound source to achieve multi-category ocean sound source detection. All these works demonstrate the potential of deep learning for sound source localization and detection. Notably, ITAI Orr et al. (2021) successfully published a paper in the journal of Science Robotics, using the deep neural network to improve the angle resolution by four times. However, these methods have significant limitations: 1) Relying on manually selected features to define a signal of interest requires highly sophisticated knowledge (Jiang et al., 2019) of signal processing and may not adequately describe the complex and variable time-frequency properties of sound. 2) The large number of parameters is a time-consuming step that requires exploring various neural network hyperparameters to obtain an optimal model.

While CNNs offer significant advantages such as automatically extracting relevant features from whale signals. However, their application necessitates access to large public PAM datasets. To address these problems, the concept of transfer learning was suggested (Bursać et al., 2022). Transfer learning is employed as a modeling strategy wherein a model trained on one data set (source model) is utilized to make predictions on another data set (target model). This approach enables the model to undergo update learning with small samples, thereby enhancing the adaptability of learning methods (Obara et al., 2022). This can be done in two ways: (a) fine-tuning the source model on the target dataset; (b) using the source model as a feature extractor to extract robust features for the target dataset to build the target model. (Saeed Khaki 2021) utilized transfer learning between corn and soybean yields by sharing the weights of the backbone feature extractors (biological information transfer), which demonstrated the ability of the model to predict accurately (Khaki et al., 2021).

In this study, given the favorable properties of transfer learning, we apply this approach to address localization errors due to different effects of reverberation on different signals. Thus, we propose CIAT, a composite intelligent acoustic tracking model, which mines and preserves the signal-spatial unpredictability features from two hydrophones, to achieve accurate and efficient whale tracking. This study dramatically opens a new path to tracking whale cruises without large physical “real” arrays. Specifically, our key innovations include:

(1) Remove the effects of non-source signals: an unsupervised algorithm based on enhanced cross-spectral analysis is used for horizontal azimuth estimation, which ensures the uniqueness of the solutions of CIAT and eliminate the interference of non-source signals.

(2) CNN-based distance/depth estimation pre-trained model: Automatically mine and efficiently establish signal-space feature transfer mechanism.

(3) Combining transfer learning to improve computational efficiency: For Munk or SWellEX-96 (SW-96) application environments, CIAT shares weights of the convolutional layers of the pre-trained model to reduce model parameters and subsequently helps the training process despite the small-field discretized measured data.

(4) Strengthen robustness and scalability: Comparing the experimental data of the random walk characteristics of

two hydrophones proves that CIAT has strong robustness and scalability.

2 Materials and methods

2.1 Dataset

Acquiring labeled underwater acoustic target data is challenging in practical applications. To overcome this problem, the network is trained on the synthetic data based on the prior hydrological environment information and the sound field model, to establish the pre-training model. Then, the knowledge learned by the model on the synthetic data is transferred to the small-domain discretized actual data to enhance the model's performance across different domains. Especially in the ocean waveguide environment, there are factors such as noise, reverberation, and interference, which will cause differences between the synthetic training data and the measured data. Transfer learning offers significant advantages when applied to new tasks, as it does not necessitate an identical data structure. This flexibility is particularly beneficial in dealing with deviations between synthetic and actual data. In this study, we use the measured dataset as the validation set of CIAT. As shown in Figure 1 and Table 1, the actual experimental dataset is briefly described, together with its deployment and environmental parameters (Fu et al., 2020; Kwon et al., 2020; Gupta et al., 2021; Ajala et al., 2022; Zhang et al., 2022).

From Figure 2, it is evident that there are many similarities between the acoustic signals of the sound source ship and bowhead whales. Specifically, there is a clear comb-like structure at the vocalization of the bowhead whale, which corresponds to the sound source ship. What's more, both the radiated signal from the sound source ship and the calls of whales share common characteristics such as uniform background noise and being considered quasi-steady-state processes in the short term. To fulfill the validation requirements of this study, the SW-96 experimental data is well-suited. Hence, this study employs acoustic data resembling whale signals to assess the feasibility of CIAT. As the availability of measured data is limited, synthetic data will be used to complement the CIAT data preparation. Detailed data information can be found in Table 2.

Synthetic data are generated through broadband modeling based on normal wave theory. Normal wave model is a classic sound field model, which mainly studies the amplitude and phase changes of sound signal in the sound field. It is suitable for far fields such as low frequency, shallow sea, constant level and other far fields. The solution is expressed as an integral solution in the wave equation. KRAKEN (Byun et al., 2019) uses the finite difference method to discretize the continuous problem in the wave equation, and the resulting solution is as follows:

$$p(r,z) = \frac{i}{\rho(z_s)\sqrt{8\pi r}} \cdot \exp\left(-\frac{i\pi}{4}\right) \cdot \sum_{l=1}^{\infty} \frac{\psi(z_s, r_l)}{\sqrt{r_l}} \exp(ir_l r) \quad (1)$$

where, r is the horizontal distance, is the depth, represents the density of seawater, z_s represents the depth of the sound source, and $\psi(z_s,r_l)$ is a constant and is the lth order normal wave.

The waveguide environment is simulated by the KRAKEN simulation program, and the parameters refer to the SW-96 or Munk experiment. And set the placement depth of the simulated sound source to 9m and the distance between the two hydrophones to be 150m. After calculating the sound pressure values of the broadband receiving space points, the solution of the time-varying wave equation is obtained by the Fourier synthesis method of the frequency domain solution. By doing so, uninterrupted time domain reception signals for both hydrophones are generated.

$$p(r,z,t_j) = \frac{1}{N}\sum_{k=1}^{N} S(\omega_k)\, p(r,z,\omega_k) e^{-j\omega_k t_j} \quad (2)$$

where, $S(\omega_k)$ is the sound source spectrum; is the number of FFT points, and the transmission frequency (ω_k) is {109, 127, 145, 163, 198, 232, 280, 335, 385}.

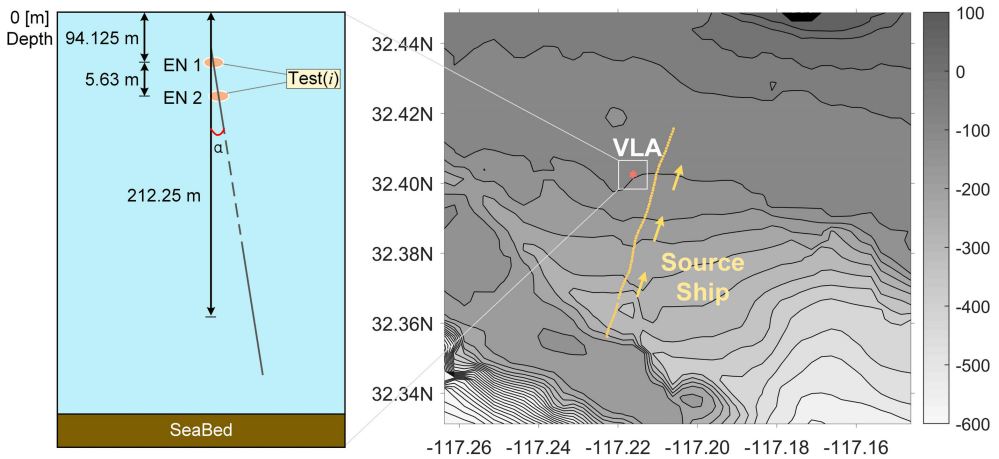


FIGURE 1
The study area near San Diego, California. The red dot marks the recording position VLA (32°40.254′ N, 117°21.620′ W) with a slight skew α, the yellow line is the track of the source ship from south to north, and the filled rectangle is defined as hydrophone signals selected as the data source for this CIAT.

TABLE 1 Overview of analytical acoustic data recorded by two acoustic recorders.

Name	Position	Deployed years	Start time	End time	Duration time (min)	Sampling Rate (Hz)	Depth (m)	Bandwidth (Hz)
1	32°40.254' N 117°21.620' W	10/5/96	23:15	0:30	75	1500	94.125	100~400
2	32°40.254' N 117°21.620' W	11/5/96	23:15	0:30	75	1500	99.755	100~400

The sensor calibration of all acoustic recorders is 185.3dB, and the water depth is 216.5m.

2.2 Model architecture

According to Risoud et al. (2018), azimuth, distance and depth are the three key parameters for sound source localization. However, it is important to note that azimuth estimation and distance/depth estimation are different types of tasks that may require different model architectures and feature representations. Traditional algorithms, such as cross-spectral analysis, are commonly used for azimuth estimation by analyzing the phase information of the sound signals (Li et al., 2019). In contrast, deep learning models have powerful feature learning and expressive capabilities, which can effectively capture distance- and depth-related patterns and features in sound signals. To simplify the training and inference process of the model and improve the accuracy of parameter estimation, we will estimate these parameters separately using their respective features and information. Doing so

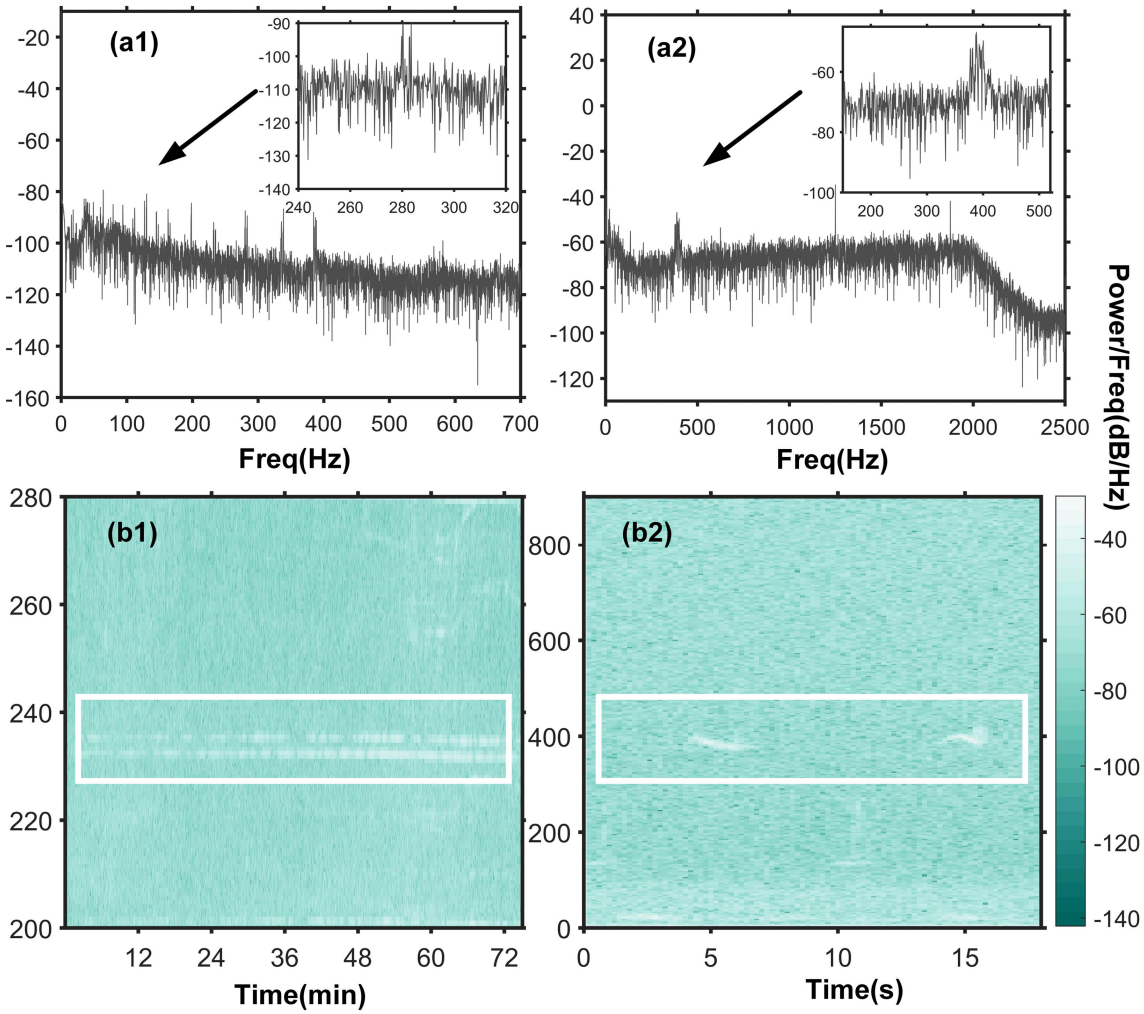


FIGURE 2
Power spectrum and time-frequency plots of the received signal from No. 1st hydrophone and the bowhead whale signal. **(A1)** The power spectrum of the signal received by No. 1st hydrophone, and **(B1)** the time-frequency diagram. **(A2)** The power spectrum of the bowhead whale, and **(B2)** the time-frequency plot. Data source NOAA Ocean Passive Acoustics Data Recorded by instrument NRS01 (72.443N, 156.602W) under the mooring platform.

TABLE 2 Data description.

Data Name	Data Composition	Data Description	Data size
Synthetic data	Source data	Based on SW-96 environmental parameters using broadband modeling	6999
	Munk data	Based on Munk environmental parameters using broadband modeling	_
Actual data	SW-96 data	SWellEX-96 experiment	_

avoids introducing data association problems and redundant information. Our proposed model combines three key technologies: unsupervised learning algorithm based on enhanced cross spectral analysis, CNN and transfer learning (Ramírez-Macías et al., 2017; Fortune et al., 2020; Kovacs et al., 2020), and Figure 3 shows the CIAT flowchart.

It can be seen from Figure 3 that CIAT begins by using the improved cross-spectrum analysis method to determine the direction of the sound source and can effectively focus on the position of the sound source, which helps to improve the accuracy and robustness of the sound source localization. Subsequently, employ a combination of CNN and transfer learning to estimate the distance/depth of the sound source. By using the CNN model, we can extract features about the depth and distance of sound sources from the input signal. Transfer learning allows us to leverage models pre-trained on other related tasks, thereby accelerating the convergence of the network and improving performance. Finally, the azimuth estimation and the distance/depth estimation results are integrated to realize the trajectory prediction. Figure 4 shows a detailed overview of the steps involved in the process.

Step 1: Enhanced cross-spectral analysis is used to get the horizontal azimuth. We calculate the cross-spectral values of the time-domain data within the frames, and then filter the spectral peaks of the frequency points to get the target angle information. Compared with traditional algorithms, this unsupervised learning algorithm eliminates the interference of non-source signals and the multiple solutions of CIAT.

Step 2: A pre-trained model is built based on the CNN algorithm to mine signal-spatial features. The source data of ambient-field spatial features are reconstructed using broadband modeling, and more unpredictable features between the received signals and the target positions are mined by establishing a signal-spatial transfer mechanism. Compared with the traditional beamforming technology, the pre-trained model could directly perceive the signal-spatial features instead of indirectly extracted phase and frequency features.

Step 3: Use transfer learning to increase the generalization ability of the CIAT model. The convolutional layers of the pre-trained model are frozen by transfer learning to preserve the effect of signal-spatial feature perception in a specific application environment (Xu and Vaziri-Pashkam, 2021; Bedriñana-Romano et al., 2022; Dumortier et al., 2022). Small-domain discrete actual data is added to the target environment to strengthen the non-mapping connection between the fully connected layer features and the actual target locations. The CIAT model could adapt to dynamic perturbations in the marine environment, significantly improving tracking accuracy.

Based on the received signals from the two hydrophones, the azimuth of the sound source is first calculated using an enhanced cross-spectrum analysis. Then a pre-trained model is built using CNN algorithm to infect signal-spatial features. Finally, transfer learning is combined to enhance the generalization ability of the CIAT model.

2.2.1 Enhanced cross-spectral algorithm

The cross-spectrum method utilizes the principle of signal correlation (Virovlyansky, 2020; Lo, 2021) and can effectively suppress noise. Let $s_1(t)$ and $s_2(t)$ be the broadband signals

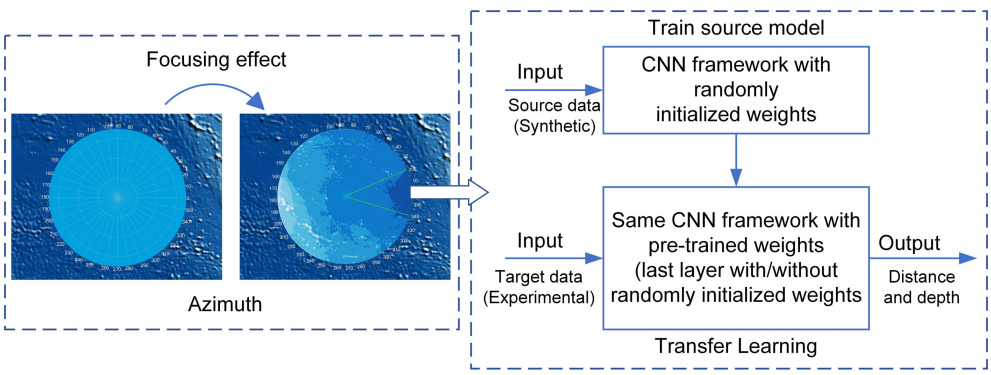


FIGURE 3
Schematic diagram of the overall architecture of CIAT.

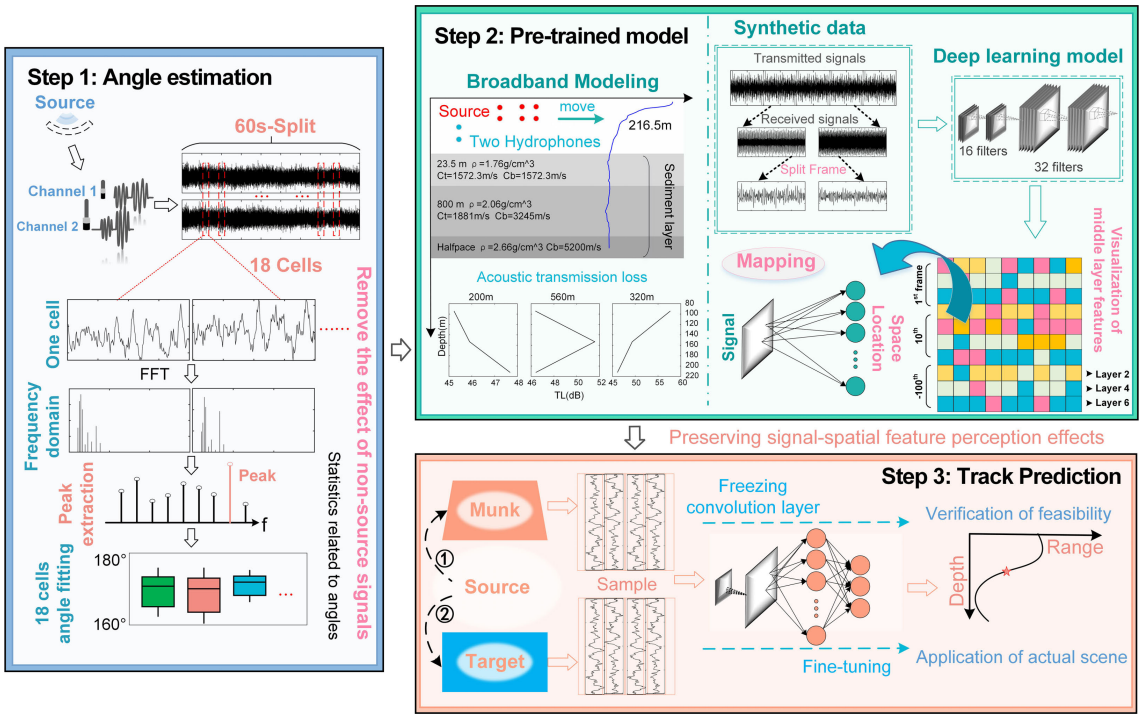


FIGURE 4
A detailed overview of the three steps performed by CIAT.

received by the two hydrophones, then the cross energy spectral density is expressed as:

$$E_{12}(f) = F_1(f)F_2^{\star}(f) = |F_1(f)|^2 e^{i2\pi f\Delta t} \tag{3}$$

where, $F_1(f)$ and $F_2(f)$ are the spectral density functions of $s_1(t)$ and $s_2(t)$, respectively. According to the time delay characteristics of the Fourier transform, the time delay information is included in the phase information of the cross spectrum, then the phase of the cross-spectrum density at the frequency f is:

$$\varphi(f) = arctan[I(f)/Q(f)] \tag{4}$$

For a wideband signal with a bandwidth of B, in order to improve the accuracy of the phase difference measurement, we divide the time-domain received signals of the two hydrophones into frames, and calculate the cross-spectrum value of each frame separately. Then calculate the phase difference of each frequency sampling point in the signal bandwidth according to the above formula, and take the maximum value as the accurate phase difference of the center frequency sampling point to calculate the azimuth angle of the incident signal. Without considering the phase ambiguity, the maximum phase difference is:

$$\Delta\varphi(f) = \max(arctan[I(f_m)/Q(f_m)]) \tag{5}$$

where, $\left(f_0 - \frac{B}{2}\right) \le f_m \le \left(f_0 + \frac{B}{2}\right)$. The improved cross-spectral analysis method estimates the azimuth of the target by taking the frequency point corresponding to the maximum spectral value. Compared with the traditional cross-spectrum method, the method effectively eliminates the interference of non-source signals, thereby significantly improving the direction-finding accuracy.

2.2.2 Training process

CNN is one of the most powerful deep learning architectures that can automatically extract necessary features from raw data without any hand-crafted features. It has gained popularity in various fields such as image recognition, speech recognition, and natural language processing. In addition, the main reasons for using dual-channel end-to-end training are as follows. (1) the input is provided by raw audio data recorded by two hydrophones, which allows it to perform joint feature learning with passive whales, avoiding manual feature selection. Meanwhile, (2) an end-to-end data-driven approach brings us the possibility to capture more complex spatiotemporally correlated latent features of the two hydrophones through the main convolution operation (Chen and Schmidt, 2021; Dayal et al., 2022).

Table 3 shows the size and number of convolutional filters in the proposed topological network. Adding a batch normalization layer after the input layer enhances the training process by reducing the drift of the input data distribution. This normalization technique accelerates network training by ensuring more stable gradients and mitigating the impact of varying input distributions. By normalizing the activations within each mini-batch, batch normalization promotes faster convergence and improves the overall efficiency of the network, and then concatenates two identical convolutional blocks. From an audio signal processing perspective, a convolutional unit can be viewed as a set of finite impulse response (FIR) filters with learnable coefficients, allowing more complex and comprehensive sample latent features to be extracted from large-scale data. The max pooling operation preserves more important features. The same is true for the

TABLE 3 CIAT parameters.

Type/stride	CIAT parameters
BN	
conv	(1×5)(16)
max pool	(1×3)
conv	(1×5)(16)
max pool	(1×3)
conv	(1×5)(32)
max pool	(1×3)
conv	(1×5)(32)
max pool	(1×3)
FC- Dropout(-) Output (range and depth)	

remaining two convolution blocks. The "distributed features" are flattened and fed into a fully connected hidden layer of 100 units, designed to integrate and arrange the content in the filtered acoustic signal to obtain the final function as a solution.

$$\theta = (R, D) = \mathrm{F}_{out}\left(H^{L}(H^{L-1}(\cdots H^{l}(\cdots H^{1}(s))))\right) \quad (6)$$

where $H()$ is the calculation process of a complete hidden layer. s is the time domain acoustic data of two hydrophones. $\mathrm{F}_{out}(x)= Act(\omega x + b)$ represents the fully connected layer, where w and b are the parameters of the fully connected layer. ReLU activation function is used in all layers except the output layer to ensure that all outputs are positive and reduce the risk of gradient explosion and gradient disappearance during network training. In each training round, the model is optimized for accuracy using the Adam algorithm.

2.2.3 Model fine-tuning

In CIAT, we build the target models using exactly the same architecture as the pre-trained (Zhong et al., 2021) models and use the parameters of these pre-trained models (except for the parameters of the output layer) as initial parameters. These transferred models are then retrained using small samples of actual data, a process called fine-tuning. Different transfer learning experiments are also performed to test the robustness of the transfer learning scheme by passing only some parameters of the hidden layers or fine-tuning the parameters of the selected layers, and the model performance was evaluated using the same approach. Here, we demonstrate that even using a small experimental training set, it is possible to extract significant signal-spatial features by expanding the dataset with computer-generated raw acoustic data.

2.3 Prediction performance evaluation

Model performance metrics for Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Correct Positioning Ratio (CPR) are defined as below:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left(\left|r_i - r_{\hat{i}}\right| + \left|d_i - d_{\hat{i}}\right|\right) \quad (7)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left((r_i - r_{\hat{i}})^2 + (d_i - d_{\hat{i}})^2\right)} \quad (8)$$

$$CPR = \frac{\sum_{i=1}^{N}(\eta(i))}{N} * 100\,\% \quad (9)$$

$$\eta(i) = \begin{cases} 1, \frac{|r-\hat{r}|}{r} < 0.1 \quad and \quad \frac{|d-\hat{d}|}{d} < 0.1 \\ 0, \quad else \end{cases} \quad (10)$$

where N is the number of test sets, r is the real distance, and $\hat{r}$ is the predicted distance; d is the real depth, and $\hat{d}$ is the predicted depth. The smaller the *MAE* and *RMSE*, the better the performance, and the larger the *CPR* value, the better the model performance. These three indicators can intuitively reflect the closeness of the predicted result to the true value (Masmitja et al., 2020; Fonseca et al., 2022; Guzman et al., 2022; Skarsoulis et al., 2022).

3 Results

3.1 Horizontal azimuth estimation

The azimuth estimation process refers to Step 1 of the Model Architecture. We use enhanced cross-spectral analysis to obtain the target horizontal azimuth information. The local northeast coordinate system is established with the 1st hydrophone of the HLA as the origin, and the relative coordinates of other positions are recalculated by Universal Transverse Mercator Grid System (UTM) transformation to obtain the actual azimuth (blue line in Figure 5A). To determine the mutual spectral values of the two signals, two hydrophones of VLA (Chambault et al., 2022; Yang et al., 2022) are chosen to record time-domain data in frames. Assuming the normal direction of the line connecting the 1st hydrophone and the sound source ship at the 60th minute is 0°, the azimuth angle less than 60min is θ, and the azimuth angle more than 60min is 180°-θ.

Due to the similarity in average spectral values of the signals captured by the two hydrophones, the traditional cross-spectrum analysis method faces challenges in distinguishing them. As a result, the calculated angle tends to be either 0 or NaN (not a number), indicating that it cannot be reliably determined due to the similarity in average spectral values. Compared to conventional spectral analysis algorithms, our enhanced cross-spectral analysis ensures the accuracy of azimuth estimation by finding the spectral peaks corresponding to the main frequency points. This unsupervised learning algorithm maintains the intrinsic connection between the two received signals, eliminates the influence of non-source signals, and ensures the unique solution and objectivity of CIAT.

In Figure 5A, the boxplot visually represents the distribution and dispersion of the azimuth data. It effectively summarizes key statistics such as medians, quartiles, etc., providing insight into the central

tendency and variability of azimuth values. Additionally, the scatterplot in the same figure shows azimuth data obtained from a fifth-order polynomial fit, which reveals patterns and trends exhibited throughout the specified time period. As seen in detail (Table 4), particularly, the Absolute Error (AE) in the angle exceeds 10° at about 59 minutes. This phenomenon that the azimuth error is the largest when the target is closest to the hydrophone is consistent with the results of Watkins and Schevill et al., which confirms the effectiveness of our horizontal azimuth estimation algorithm and further boosts the credibility of our intelligent acoustic tracking model.

3.2 Distance/depth estimation

Distance/depth estimation includes CNN pre-trained model and transfer learning. First, the pre-trained model of CNN is built for processing received signals. The input of the model is $N * 2 * S$ dimension, where N represents the signal sample length, 2 denotes the number of channels, and S represents the signal frame length. To ensure compatibility and optimize performance, we implement the entire framework using the Python programming language and the TensorFlow library on a Windows 10 x64 system. Compared to large networks like U-Net, CNN has a shallow network structure that does not require many parameters to train its performance. This characteristic has led our model to outperform most previously used models in this research area.

The frame lengths 1001, 2001, and 3001 all demonstrate conformity to the normal distribution as predicted by the theory, thus verifying the validity of the model and its prediction accuracy. Notably, the frame length of 1001 exhibits the highest accuracy in predictions (Figure 5B). Since the underwater depth of the whales is almost constant during migration, this paper does not place a high value on depth changes. For the frame length of 1001, the distance estimation errors within 6 km are 0.0322 km/MAE, 0.0805 km/RMSE, and 94.57%/CPR. The above fully illustrates that our CNN pre-trained model could directly perceive the signal-spatial features.

We visualize the trend changes of weights acting on 16 convolutional kernel units in the first layer of the dual-channel system. Figure 6 illustrates this, where (a) represents the weight values of 16-1; (b) 16-2; (c) 16-3; and (d) 16-4. The shaded regions indicate perfect recordings when both sound waves arrive simultaneously, otherwise, they indicate a delay. From Figure 6, we can infer the following:

1) The trend changes between different weights reflect the time difference or phase difference of the sound waves reaching the two hydrophones. The weights show significant changes or overlaps at specific positions. For example, at the upward-pointing Perfect shaded arrow, we can infer that the time or phase difference of the sound waves' arrival is small.
2) The differences between different weights can reflect the variations in the signals received by the two hydrophones. If the weights exhibit noticeable differences at certain positions, such as the right-pointing Delayed shaded arrow, it suggests significant discrepancies in the signals received by the hydrophones at that position.

By considering the combined trend changes and differences in weights, we can deduce that the signals received by the two hydrophones have different arrival times and phase differences, and there are significant discrepancies at certain positions. This aligns with the actual scenario of sound propagation reaching the two hydrophones, thereby enhancing the model's interpretability and reliability.

Further, Figure 7 provides insights into the intermediate layer feature representations of CIAT. When examining the signal features of different time frames (signals 1, 2, 3), the features extracted from the last 100 frames are slightly better than those extracted from the first and tenth frames. The reason behind this observation is that the initial time period predominantly captures the direct path sound signal, which does not exhibit a distinct multipath reflection signal pattern. As the network layers deepen, the extracted features become more specific and sparser, indicating the presence of spatial selective gradients within CIAT. Comparing (a) and (b) in Figure 7, without transfer learning (marked by

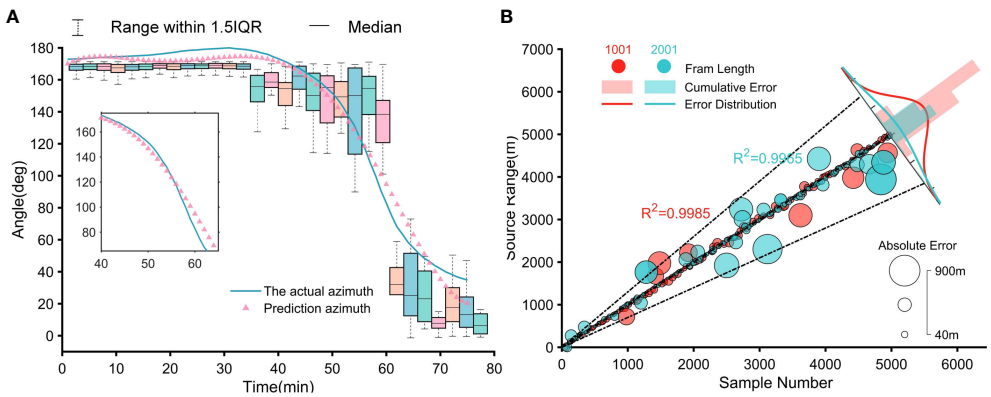


FIGURE 5
Estimation results. **(A)** SW-96 experimental data azimuth estimation. **(B)** Comparison of distance estimation results for pre-trained model frame lengths of 1001 and 2001.

TABLE 4 Azimuth estimation results.

Time/min	Actual azimuth	Conventional Spectral Analysis	AE	Enhanced cross Spectral Analysis	AE
10	174.424°	0	174.424°	173.935°	0.489°
20	176.873°	NaN	_	171.728°	5.145°
30	179.805°	NaN	_	174.083°	5.722°
40	173.227°	NaN	_	171.121°	2.106°
50	151.641°	NaN	_	146.687°	4.954°
59	90.659°	NaN	_	100.807°	10.148°

ellipses), the obtained features are blurry, and even with increasing network layers, the features extracted from two similar time frames remain indistinguishable. However, through transfer learning (marked by rectangles), the learned features are not only representative but also avoid the issue of feature blurriness.

The observations strongly suggest that CIAT is capable of extracting signal features from various time frames through a nonlinear feature extractor. Additionally, the model exhibits good generalization capabilities when applied to real-world data. These findings lay a solid foundation for the potential success of using CIAT in tracking whales during migration.

Next, the signal-spatial feature parameters of our pre-trained model obtained in the ideal environment are applied to the target environment by transfer learning to evaluate the effect of the target model on the perception of the actual received signal features (Gemba et al., 2017; Worthmann et al., 2017; Agrelo et al., 2021; Coli et al., 2022). The target model's input is Munk-based synthetic data to determine the effective transfer of signal-spatial feature mechanism, thus ensuring the feasibility of the proposed model. After that, the CNN pre-trained model's convolutional layer is frozen. However, this frozen CNN pre-trained model does not serve as the final model for the effect of dynamic ocean perturbations,

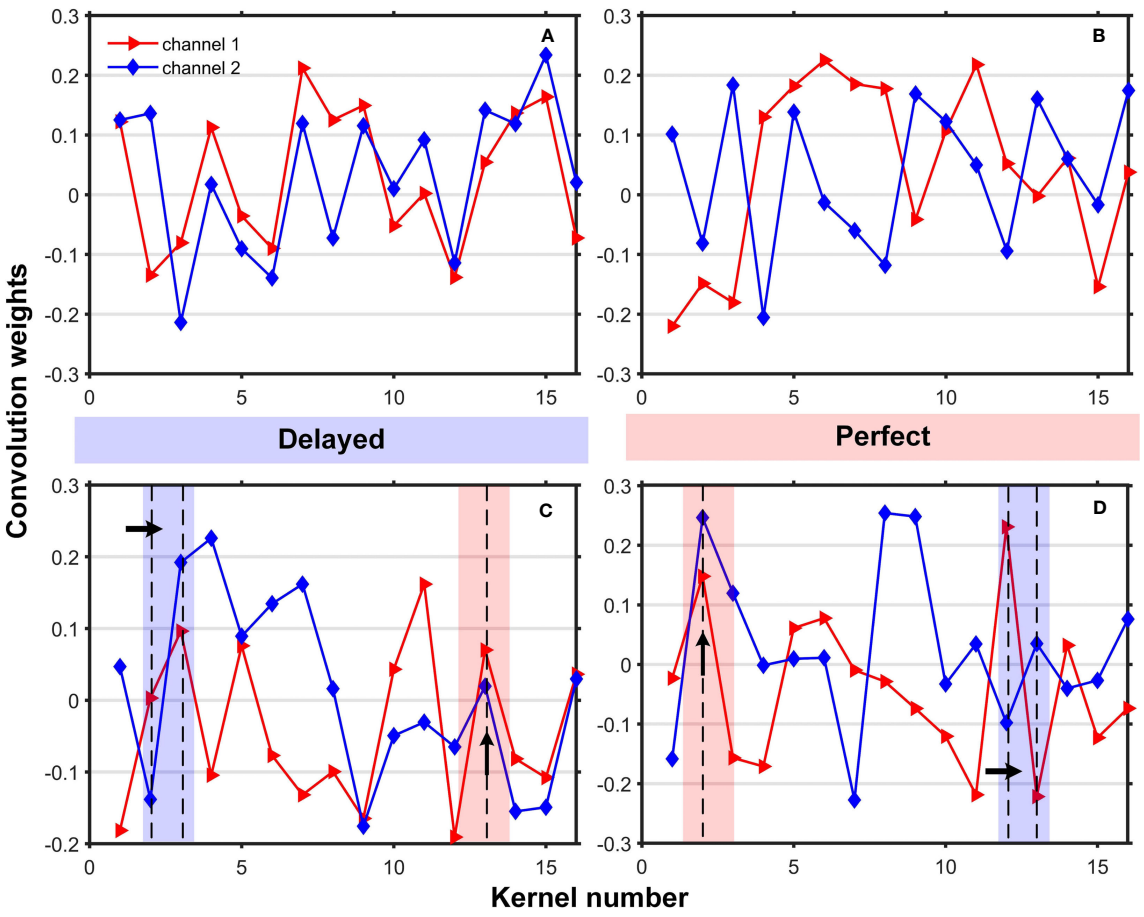


FIGURE 6
Respectively act on the weights of the dual-channel convolution kernels. **(A)** represents the weight value of 16-1; **(B)** 16-2; **(C)** 16-3; **(D)** 16-4. The shaded areas represent: two sound waves arriving at the same time are recorded as Perfect, otherwise, Delayed.

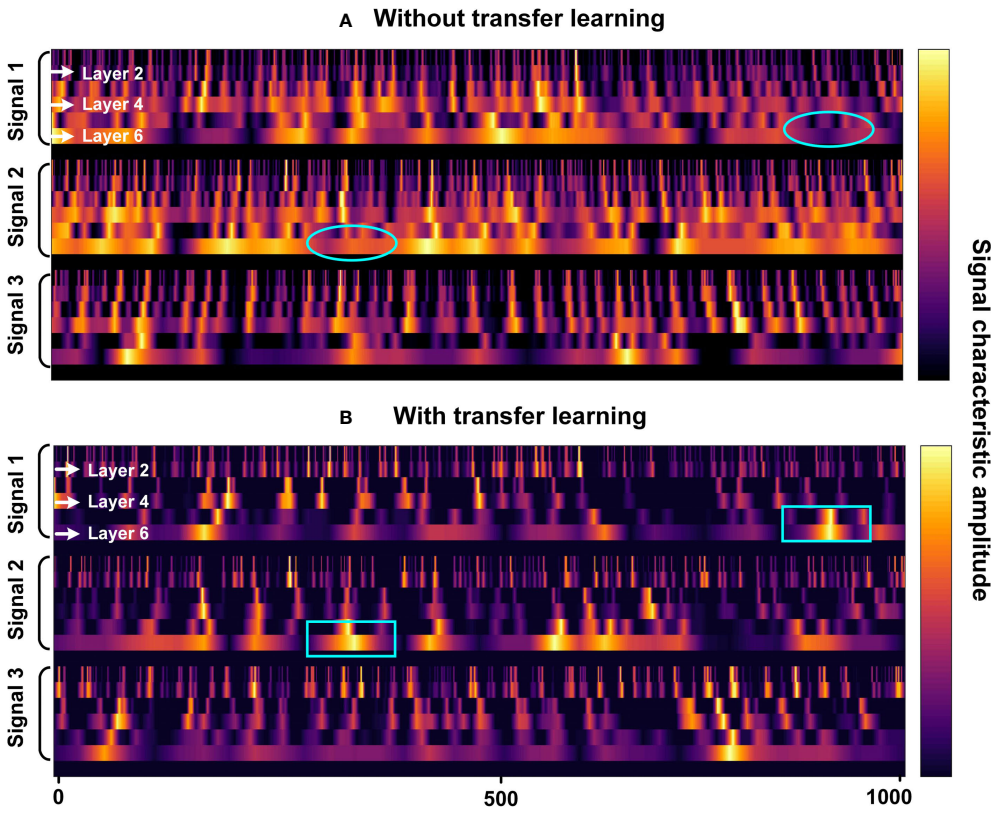


FIGURE 7
The middle layer feature representation for CIAT. **(A)** Without transfer learning; **(B)** With transfer learning.

which would result in an environmental mismatch between the source and target model datasets. Therefore, we use transfer learning to share the weight parameters of the CNN pre-trained model and put small sample data to the target model for achieving accurate prediction positions by fine-tuning the fully connected layer and setting Dropout 0.5 to build the Munk target model.

The estimation errors of the Munk target model are 0.015km/MAE and 97%/CPR (Figure 8A). It can be seen that the predicted distance of the target in the Munk environment is consistent with the actual distance, indicating that our transfer learning algorithm could make the model's generalization performance enhanced and adapt to different environments with guaranteed accuracy. In addition, we test the reproducibility of the transfer algorithm by changing the signal pattern of the source from comb to FM emission and also set Dropout 0.3. Figure 8B shows that the distance estimation errors are 0.031km/MAE and 93%/CPR, which also has high accuracy and proves the robustness of the CIAT.

Next, we apply this transfer algorithm to the actual experimental data with ambient noise and reverberation. Based on our frozen CNN pre-trained model, the first 9 minutes of raw acoustic data from two hydrophones are used as the input to the SW-96 target model, and two Dropout layers (0.5 and 0.1) are added to complete the sound source ship distance/depth prediction. As shown in the distance results, the estimation error of distance obtained within 8 km without transfer learning is 0.15 km/MAE (Figure 8C), while with transfer learning the distance estimation errors are 0.13 km/MAE, 0.164 km/RMSE, and 100% CPR, respectively (Figure 8D), demonstrating that the distance prediction accuracy using transfer learning at sparse data is higher than that without transfer learning. And Figure 9A shows that the predicted depth of the target in the SW-96 environment is consistent with the actual depth. Besides, in the same experimental environment, we also compare CIAT and traditional matching field processing (MFP) techniques (Wang et al., 2020). The results are shown in Table 5, which shows that the traditional method is severely limited by multipath propagation and spatial correlation in the marine environment, and it cannot complete the tracking task solely by relying on two hydrophones. These further verify that our proposed model only based on two hydrophones can adapt to the effects of dynamic marine environmental perturbations brought about by scene switching and can be extended to applications in actual marine environments.

3.3 Transferability and sensitivity

Our model enables to perform high-precision tracking in both Munk and SW-96 actual environments, and it is a key advantage of our CIAT to achieve high-precision tracking at 8 km 0.13 km/MAE in actual marine environments using two hydrophones. At the same time, CIAT can also adapt to switching between different marine

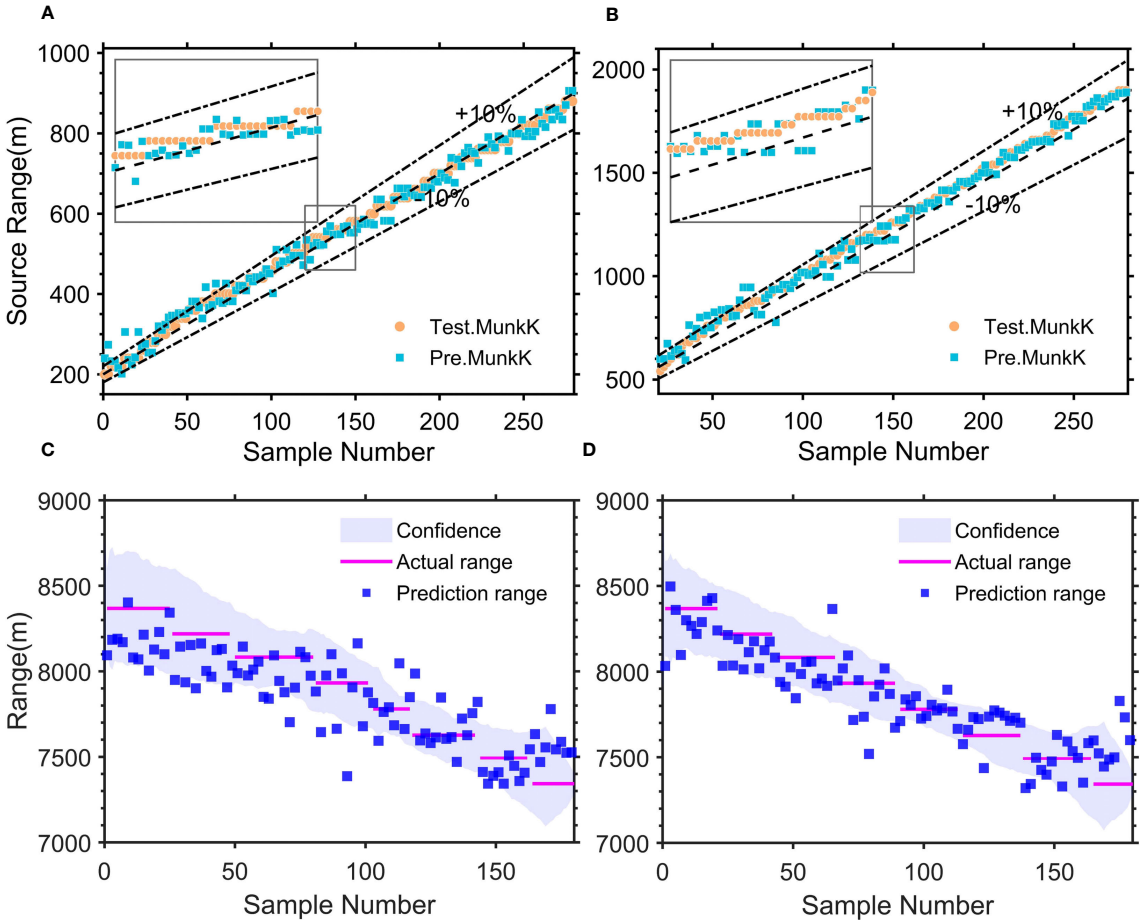


FIGURE 8
Positioning and tracking results. **(A)** Distance estimation results for synthetic data of the Munk environment, and **(B)** results for the change of signal form to FM signal. **(C)** The prediction result of SW-96 experimental sound source distance without transfer learning, and **(D)** with transfer learning.

environments like Munk and SW-96, but since both CNN and transfer learning in CIAT are black-box models, there is currently no effective physical mechanism to explain this phenomenon. Therefore, another important direction of our work focuses on explaining the physical mechanism of CIAT to support switching between different marine environments.

Theoretically, our CIAT is mainly affected by the ambient noise and ocean reverberation that exist in different marine environments when applied. However, since the source dataset is synthetic data used for broadband modeling, it is determined that the features shared will not be ambient noise.

From the SW-96 experimental results, the CIAT is directly applied to sound source distance/depth estimation after the first 9 minutes of transfer training. The statistical errors of the predicted 10-16 min distance are 0.367 km/MAE, 0.429 km/RMSE, and 91.87%/CPR, while the statistical errors of 10-19 min are 0.545 km/MAE, 0.628 km/RMSE, and 61.13%/CPR, and the effective time of the model tracking time is longer than 7 min (Figure 9B). From the measured data MAE, RMSE and CPR (Figure 9C), these three performance indicators can show that the error of CIAT increases with increasing tracking time, demonstrating that the spatial characteristics of the transmitted signals belong to the time domain. Additionally, as tracking time increases, various interface scattered acoustic waves are continuously superimposed in the hydrophone signals, also exhibiting time-domain characteristics. Therefore, we believe that the signal-spatial features conveyed by the transfer learning of CIAT are oceanic reverberations, which are the physical mechanism of their ability to support switching between different marine environments.

Transfer learning in CIAT conveys the signal-spatial features that are ocean reverberations, which support the interpretation of switching between different marine environments. We then conducted two sets of experiments to further measure the ability of CIAT to adapt to such environmental differences.

Group 1: The spacing between the two hydrophones is fixed for different permutations.

As shown in Figure 10A, the prediction error distribution tends to be consistent, although the combination categories are not identical. Setting the distance to 5.63m, the prediction errors for different combinations are shown in Table 6, which proves that the signal-spatial features perceived by the pre-trained model are effectively transferred under a certain spacing. Therefore, the model can obtain accurate prediction results using 2 hydrophones under a certain spacing. This experiment illustrates that under a

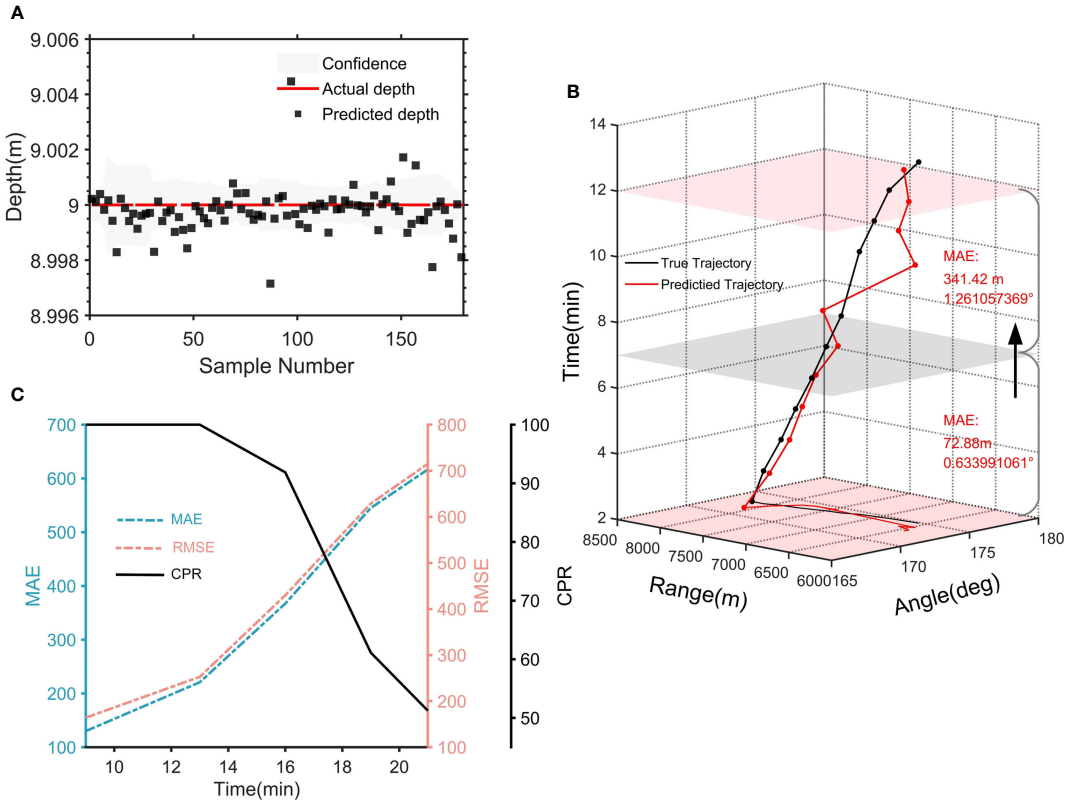


FIGURE 9
Tracking results. **(A)** Depth estimation results. **(B)** Tracking trajectory of the sound source ship. X-axis the distance of the source ship from the 1st hydrophone, Y-axis the azimuth of the source ship, and Z-axis the time dimension. Green line the real trajectory and the orange line is the predicted. **(C)** The three indicators of MAE, RMSE, and CPR verify the accuracy of the CIAT model, and show the trend of the indicators as time increase and evaluate the generalization ability of the model.

certain spacing, the change in spatial location has little effect on the adaptive ability of CIAT.

Group 2. One hydrophone is settled, and the spacing between the two hydrophones is adjusted.

Figure 10B shows that the prediction error distribution still tends to be similar when the spacing between two hydrophones is changed. As shown in Table 7, the prediction errors fluctuate slightly without significant differences. This experiment illustrates that CIAT is still highly adaptable to the environment when the spacing and spatial location of two hydrophones are both changed.

4 Discussion

In this study, we propose a composite intelligent tracking model (CIAT) to achieve both azimuth and distance/depth estimation with

TABLE 5 Comparison results of CIAT and MFP.

Distance (m)	CIAT	AE	MFP	AE
	MAE: 116.229		MAE: 2451.375	
8368	8161.774	206.225	8600	232
8219	8033.095	185.904	6600	1619
8083	7982.247	100.752	100	7983
7932	7846.973	85.026	7600	332
7780	7799.913	19.913	7600	180
7627	7684.014	57.014	5100	2527
7495	7454.631	40.368	5100	2395
7343	7577.628	234.628	3000	4343

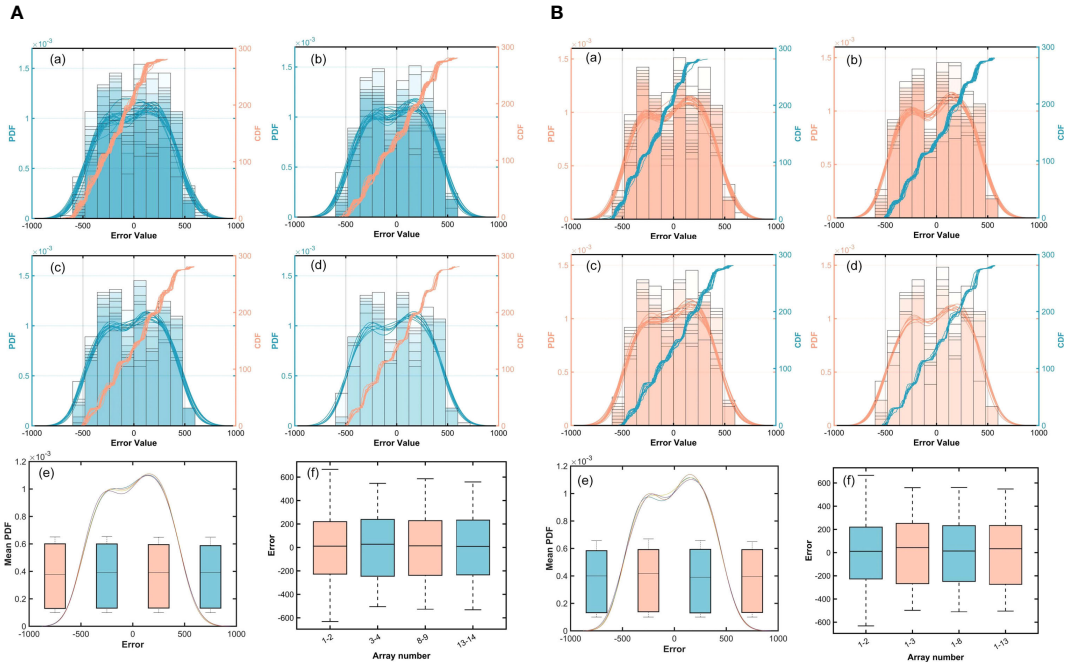


FIGURE 10
Prediction error distribution for repeated experiments. **(A)**: (a) Error histogram, probability density function and cumulative distribution function of error points at 5.63m. (b) 19-11.26m. (c) 14-38.41m. (d) 7-67.56m. (e) and (f) Error distribution plots and the mean PDF. **(B)**: (a) Error histogram, probability density function and error point cumulative distribution function for different hydrophone spacing, respectively. (b) 3-8 (c) 8-12 (d) 13-7 (e) and (f) Error distribution plots and the mean PDF.

solely two hydrophones, thereby allowing complete and accurate tracking of whales, especially 0.13 km/MAE within the range of 8km. It addresses that the current spatial-temporal correlation techniques are limited by the hydrophone quantity accumulation, arrival time sensitivity and low tracking accuracy. Additionally, another important direction of our study focuses on explaining the physical mechanism of CIAT to support switching applications in different marine environments.

For the horizontal azimuth estimation, we use the enhanced cross-spectral analysis based on unsupervised algorithm to overcome the problem that traditional methods are seriously affected by non-source signals and multiple solutions of CIAT. We calculate the cross-spectrum values of the time domain sub-frames of the two hydrophone received signals, and then estimate the azimuth of the target based on the obtained spectral peaks of the corresponding frequency points. The results demonstrate that the minimum error reaches 0.489° and the average error is 4.762° within 75 min, which solves the failure of the traditional cross-spectral orientation methods and obtains the azimuth information with high precision.

For the distance/depth estimation, the spatial feature source data is reconstructed by broadband modeling to overcome the sparsity of the measured data. Then, a CNN pre-trained model is constructed to mine more obvious and robust features between the received signals and the target positions by establishing the signal-spatial transfer mechanism to avoid the dependence on indirectly extracted features such as phase and frequency.

Transfer learning is used to improve the generalization ability of CIAT model. For the Munk and SW-96 marine environments, the perceptual effects of signal-spatial features are preserved by freezing the convolutional layers of the CNN pre-trained model. Then small domain discretization of actual data is introduced to the target model to enhance the non-mapping relationship between fully connected layer features and actual target locations. The results

TABLE 6 Numerical statistical properties of errors.

	Max/m	Min/m	Mean/m	Variance/m	Median/m	Skewness/m	Kurtosis/m
1-2	665.88	-631.85	10.63	237.12	10.94	0.021	2.15
2-3	658.59	-606.03	0.39	223.89	-8.62	0.11	2.24
3-4	546.37	-505.06	-2.88	245.52	27.43	-0.029	1.84
18-19	565.90	-484.46	-5.60	248.80	-6.21	-0.041	1.85
19-20	547.84	-500.84	-6.29	248.10	-2.05	-0.035	1.84
20-21	626.94	-641.60	6.28	239.46	14.93	-0.064	2.09

TABLE 7 Numerical statistical properties of errors.

	Max/m	Min/m	Mean/m	Variance/m	Median/m	Skewness/m	Kurtosis/m
1-2	665.88	-631.85	10.63	237.12	10.94	0.021	2.15
1-3	551.20	-477.41	-3.27	244.41	46.10	-0.04	1.82
1-4	559.69	-497.78	5.03	243.63	42.88	-0.026	1.83
1-19	570.02	-503.21	6.33	245.12	18.54	-0.025	1.83
1-20	556.42	-493.69	-4.38	244.00	2.47	-0.015	1.84
1-21	557.50	-475.58	6.66	246.05	9.65	-0.022	1.84

demonstrate that our model exhibits generalization capabilities that enable it to adapt to changes in scene switching, hydrophone spacing, and signal reception form, and accurately predict target location information even with less data and unknown environmental conditions. Furthermore, from the perspective of theoretical analysis and repeatable experiments, it is demonstrated that the signal-spatial features transmitted by transfer learning are ocean reverberation. This is crucial to explain the physical mechanism by which CIAT enables to support switching between different marine environments.

Our proposed whale tracking model breaks the paradigm of improving tracking accuracy by accumulating physically "real" arrays, but fully senses and mines the unpredictable signal-spatial features of the two hydrophones for precise tracking. Especially, the transmitted signal-spatial features are found to be oceanic reverberations during the prediction process. This provides an explanation for the physical mechanism by which CIAT would be able to support switching applications in different marine environments. However, one of the most important limitations of this study is the small size of the training/validation set used. It is foreseeable that in the future, more acoustic received signal could be collected as an extension to provide more precise information for whale diversity conservation and stranding management.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: http://swellex96.ucsd.edu/s5.htm.

Author contributions

JX and KJ conceived the study and coordinated the project effort. KJ and CL conducted the acoustic data analysis, tracking model validation, writing, and visualization. JX, KJ, XZ, CL, LX, and YL conducted the formal analysis, review, editing, and supervision. All authors contributed to the article and approved the submitted version.

Funding

This research was jointly supported by the National Natural Science Foundation of China (41706106).

Acknowledgments

We thank Ocean Acoustics laboratory members for critical reading of the manuscript and constructive suggestions during our research and Jim Murray formerly of the Marine Physical Lab for valuable SWellEx-96 experimental data.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Agrelo, M., Daura-Jorge, F. G., Rowntree, V. J., Sironi, M., Hammond, P. S., Ingram, S. N., et al. (2021). Ocean warming threatens southern right whale population recovery. *Sci. Adv.* 7, eabh2823. doi: 10.1126/sciadv.abh2823

Ajala, S., Muraleedharan Jalajamony, H., Nair, M., Marimuthu, P., and Fernandez, R. E. (2022). Comparing machine learning and deep learning regression frameworks for accurate prediction of dielectrophoretic force. *Sci. Rep.* 12, 1–17. doi: 10.1038/s41598-022-16114-5

Aulich, M. G., McCauley, R. D., Saunders, B. J., and Parsons, M. J. (2019). Fin whale (Balaenoptera physalus) migration in Australian waters using passive acoustic monitoring. *Sci. Rep.* 9, 1–12. doi: 10.1038/s41598-019-45321-w

Bedriñana-Romano, L., Zerbini, A. N., Andriolo, A., Danilewicz, D., and Sucunza, F. (2022). Individual and joint estimation of humpback whale migratory patterns and their environmental drivers in the Southwest Atlantic Ocean. *Sci. Rep.* 12, 1–16. doi: 10.1038/s41598-022-11536-7

Bursać, P., Kovačević, M., and Bajat, B. (2022). Instance-based transfer learning for soil organic carbon estimation. *Front. Env. Sci.* 10. doi: 10.3389/fenvs.2022.1003918

Byun, G., Akins, H., Song, H. C., and Kuperman, W. A. (2019). Robust matched field processing for array tilt and environmental mismatch. *J. Acoust. Soc Am.* 146, 2962–2962. doi: 10.1121/1.5137294

Chambault, P., Kovacs, K. M., Lydersen, C., Shpak, O., Teilmann, J., Albertsen, C. M., et al. (2022). Future seasonal changes in habitat for Arctic whales during predicted ocean warming. *Sci. Adv.* 8, eabn2422. doi: 10.1126/sciadv.abn2422

Cheeseman, T., Southerland, K., and Park, J. (2022). Advanced image recognition: a fully automated, high-accuracy photo-identification matching system for humpback whales. *Mamm. Biol.* 102(3), 915–929. doi: 10.1007/s42991-021-00180-9

Chen, R., and Schmidt, H. (2021). Model-based convolutional neural network approach to underwater source-range estimation. *J. Acoust. Soc Am.* 149, 405–420. doi: 10.1121/10.0003329

Coli, G. M., Boattini, E., Filion, L., and Dijkstra, M. (2022). Inverse design of soft materials via a deep learning–based evolutionary strategy. *Sci. Adv.* 8 (3), eabj6731. doi: 10.1126/sciadv.abj6731

Davis, G. E., Baumgartner, M. F., Bonnell, J. M., and Bell, J. (2017). Long-term passive acoustic recordings track the changing distribution of North Atlantic right whales (Eubalaena glacialis) from 2004 to 2014. *Sci. Rep.* 7, 13460. doi: 10.1038/s41598-017-13359-3

Dayal, A., Yeduri, S. R., Koduru, B. H., Jaiswal, R. K., Soumya, J., Srinivas, M. B., et al. (2022). Lightweight deep convolutional neural network for background sound classification in speech signals. *J. Acoust. Soc Am.* 151, 2773–2786. doi: 10.1121/10.0010257

Ding, J., Ke, Y., Cheng, L., Zheng, C., and Li, X. (2020). Joint estimation of binaural distance and azimuth by exploiting deep neural networks. *J. Acoust. Soc Am.* 147, 2625–2635. doi: 10.1121/10.0001155

Dumortier, L., Guépin, F., Delignette-Muller, M. L., Boulocher, C., and Grenier, T. (2022). Deep learning in veterinary medicine, an approach based on CNN to detect pulmonary abnormalities from lateral thoracic radiographs in cats. *Sci. Rep.* 12, 1–12. doi: 10.1038/s41598-022-14993-2

Ferreira, R., Dinis, A., Badenas, A., Sambolino, A., Marrero-Pérez, J., Crespo, A., et al. (2021). Bryde's whales in the North-East Atlantic: New insights on site fidelity and connectivity between oceanic archipelagos. *Aquat. Conserv.* 31, 2938–2950. doi: 10.1002/aqc.3665

Fonseca, C. T., Pérez-Jorge, S., Prieto, R., Oliveira, C., Tobeña, M., Scheffer, A., et al. (2022). Dive behavior and activity patterns of fin whales in a migratory habitat. *Front. Mar. Sci.* 1134 (2022). doi: 10.3389/fmars.2022.875731

Fortune, S. M., Ferguson, S. H., Trites, A. W., Hudson, J. M., and Baumgartner, M. F. (2020). Bowhead whales use two foraging strategies in response to fine-scale differences in zooplankton vertical distribution. *Sci. Rep.* 10, 1–18. doi: 10.1038/s41598-020-76071-9

Frasier, K. E., Garrison, L. P., Soldevilla, M. S., Wiggins, S. M., and Hildebrand, J. A. (2021). Cetacean distribution models based on visual and passive acoustic data. *Sci. Rep.* 11, 1–16. doi: 10.1038/s41598-021-87577-1

Fu, L., Zhang, L., Dollinger, E., Peng, Q., Nie, Q., and Xie, X. (2020). Predicting transcription factor binding in single cells through deep learning. *Sci. Adv.* 6, eaba9031. doi: 10.1126/sciadv.aba9031

Gemba, K. L., Nannuru, S., Gerstoft, P., and Hodgkiss, W. S. (2017). Multi-frequency sparse Bayesian learning for robust matched field processing. *J. Acoust. Soc Am.* 141, 3411–3420. doi: 10.1121/1.4983467

Gupta, V., Choudhary, K., Tavazza, F., Campbell, C., Liao, W. K., Choudhary, A., et al. (2021). Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. *Nat. Commun.* 12, 1–10. doi: 10.1038/s41467-021-26921-5

Guzman, H. M., Collatos, C. M., and Gomez, C. G. (2022). Movement, behavior, and habitat use of whale sharks (Rhincodon typus) in the tropical Eastern Pacific Ocean. *Front. Mar. Sci.* 1068. doi: 10.3389/fmars.2022.793248

Henley, S. F., Cavan, E. L., Fawcett, S. E., Kerr, R., and Smith, S. (2020). Changing biogeochemistry of the Southern Ocean and its ecosystem implications. *Front. Mar. Sci.* 7. doi: 10.3389/fmars.2020.00581

Jiang, J. J., Bu, L. R., Duan, F. J., Wang, X. Q., Liu, W., Sun, Z. B., et al. (2019). Whistle detection and classification for whales based on convolutional neural networks. *Appl. Acoust.* 150, 169–178. doi: 10.1016/j.apacoust.2019.02.007

Jiang, S., Wu, L., Yuan, P., Sun, Y., and Liu, H. (2020). Deep and CNN fusion method for binaural sound source localization. *J. Engineering.* 2020 (13), 511–516. doi: 10.1049/joe.2019.1207

Jones, J. M., Hildebrand, J. A., Thayre, B. J., Jameson, E., Small, R. J., and Wiggins, S. M. (2022). The influence of sea ice on the detection of bowhead whale calls. *Sci. Rep.* 12, 1–15. doi: 10.1038/s41598-022-12186-5

Khaki, S., Pham, H., and Wang, L. (2019). Simultaneous corn and soybean yield prediction from remote sensing data using deep transfer learning. *Sci. Rep.* 21(1), 11132. doi: 10.1038/s41598-021-89779-z

Kovacs, K. M., Lydersen, C., Vacquiè-Garcia, J., Shpak, O., Glazov, D., and Heide-Jørgensen, M. P. (2020). The endangered Spitsbergen bowhead whales' secrets revealed after hundreds of years in hiding. *Biol. Letters.* 16, 20200148. doi: 10.1098/rsbl.2020.0148

Kujawski, A., and Sarradj, E. (2022). Fast grid-free strength mapping of multiple sound sources from microphone array data using a Transformer architecture. *J. Acoust. Soc Am.* 152 (5), 2543–2556. doi: 10.1121/10.0015005

Kwon, H. Y., Yoon, H. G., Lee, C., Chen, G., Liu, K., Schmid, A. K., et al. (2020). Magnetic Hamiltonian parameter estimation using deep learning techniques. *Sci. Adv.* 6, eabb0872. doi: 10.1126/sciadv.abb0872

Li, P., Zhang, X., and Zhang, W. (2019). Direction of arrival estimation using two hydrophones: Frequency diversity technique for passive sonar. *Sensors.* 19 (9), 2001. doi: 10.3390/s19092001

Lo, K. W. (2021). A matched-field processing approach to ranging surface vessels using a single hydrophone and measured replica fields. *J. Acoust. Soc Am.* 149, 1466–1474. doi: 10.1121/10.0003631

Masmitja, I., Navarro, J., Gomariz, S., Aguzzi, J., Kieft, B., O'Reilly, T., et al. (2020). Mobile robotic platforms for the acoustic tracking of deep-sea demersal fishery resources. *Sci. Robot.* 5, eabc3701. doi: 10.1126/scirobotics.abc3701

Obara, Y., and Nakamura, R. (2022). Transfer learning of long short-term memory analysis in significant wave height prediction off the coast of western Tohoku, Japan. *Ocean. Eng.* 266, 113048. doi: 10.1016/j.oceaneng.2022.113048

Orr, I., Cohen, M., Damari, H., Halachmi, M., Raifel, M., and Zalevsky, Z. (2021). Coherent, super-resolved radar beamforming using self-supervised learning. *Sci. Robot.* 6, eabk0431. doi:nbsp;10.1126/scirobotics.abk0431

Parsons, E. C. M., and Rose, N. A. (2022). "The history of cetacean hunting and changing attitudes to whales and dolphins," in *Marine Mammals: the Evolving Human Factor* (Cham, Switzerland: Springer Nature), 219–254. doi: 10.1007/978-3-030-98100-6_7

Perez, M. A., Limpus, C. J., Hofmeister, K., Shimada, T., Strydom, A., Webster, E., et al. (2022). Satellite tagging and flipper tag recoveries reveal migration patterns and foraging distribution of loggerhead sea turtles (Caretta caretta) from Eastern Australia. *Mar. Biol.* 169, 1–15. doi: 10.1007/s00227-022-04061-8

Ramírez-Macías, D., Queiroz, N., Pierce, S. J., Humphries, N. E., Sims, D. W., and Brunnschweiler, J. M. (2017). Oceanic adults, coastal juveniles: tracking the habitat use of whale sharks off the Pacific coast of Mexico. *PeerJ.* 5, e3271. doi: 10.7717/peerj.3271

Risoud, M., Hanson, J. N., Gauvrit, F., Renard, C., Lemesre, P. E., Bonne, N. X., et al. (2018). Sound source localization. *Eur. Ann. otorhinolaryngology Head Neck diseases.* 135 (4), 259–264. doi: 10.1016/j.anorl.2018.04.009

Roman, J., Estes, J. A., Morissette, L., Smith, C., Costa, D., McCarthy, J., et al. (2014). Whales as marine ecosystem engineers. *Front. Ecol. Environ.* 12, 377–385. doi: 10.1890/130220

Roman, J., Nevins, J., Altabet, M., Koopman, H., and Mccarthy, J. (2016). Endangered right whales enhance primary productivity in the Bay of Fundy. *PloS One* 11, e0156553. doi: 10.1371/journal.pone.0156553

Shankar, N., Bhat, G. S., and Panahi, I. M. (2020). Efficient two-microphone speech enhancement using basic recurrent neural network cell for hearing and hearing aids. *J. Acoust. Soc Am.* 148 (1), 389–400. doi: 10.1121/10.0001600

Skarsoulis, E. K., Piperakis, G. S., Orfanakis, E., Papadakis, P., Pavlidi, D., Kalogerakis, M. A., et al. (2022). A real-time acoustic observatory for sperm-whale localization in the Eastern Mediterranean Sea. *Front. Mar. Sci.* 674. doi: 10.3389/fmars.2022.873888

Song, H. C. (2018). Classification of multiple source depths in a time-varying ocean environment using a convolutional neural network (CNN). *J. Acoust. Soc Am.* 144 (3), 1744–1744. doi: 10.1121/1.5067732

Vanselow, K. H. (2020). Where are Solar storm-induced whale strandings more likely to occur? *Int. J. Astrobiol.* 19, 413–417. doi: 10.1017/S1473550420000051

Virovlyansky, A. L. (2020). Beamforming and matched field processing in multipath environments using stable components of wave fields. *J. Acoust. Soc Am.* 148, 2351–2360. doi: 10.1121/10.0002352

Wang, X., Waqar, M., Yan, H. C., Louati, M., Ghidaoui, M. S., Lee, P. J., et al. (2020). Pipeline leak localization using matched-field processing incorporating prior information of modeling error. *Mech. Syst. Signal. Pr.* 143, 106849. doi: 10.1016/j.ymssp.2020.106849

White, E. L., White, P. R., Bull, J. M., Risch, D., Beck, S., and Edwards, E. W. (2022). More than a whistle: Automated detection of marine sound sources with a convolutional neural network. *Front. Mar. Sci.* 9:879145. doi: 10.3389/fmars.2022.879145

Worthmann, B. M., Song, H. C., and Dowling, D. R. (2017). Adaptive frequency-difference matched field processing for high frequency source localization in a noisy shallow ocean. *J. Acoust. Soc Am.* 141, 543–556. doi: 10.1121/1.4973955

Xu, Y., and Vaziri-Pashkam, M. (2021). Limits to visual representational correspondence between convolutional neural networks and the human brain. *Nat. Commun.* 12, 1–16. doi: 10.1038/s41467-021-22244-7

Yang, L., Liu, X., Zhu, W., Zhao, L., and Beroza, G. C. (2022). Toward improved urban earthquake monitoring through deep-learning-based noise suppression. *Sci. Adv.* 8, eabl3564. doi: 10.1126/sciadv.abl3564

Yangzhou, J., Ma, Z., and Huang, X. (2019). A deep neural network approach to acoustic source localization in a shallow water tank experiment. *J. Acoust. Soc Am.* 146 (6), 4802–4811. doi: 10.1121/1.5138596

Zhang, M., Cheng, Y., Bao, Y., Zhao, C., Wang, G., Zhang, Y., et al. (2022). Seasonal to decadal spatiotemporal variations of the global ocean carbon sink. *Global Change Biol.* 28, 1786–1797. doi: 10.1111/gcb.16031

Zhong, M., Torterotot, M., Branch, T. A., Stafford, K. M., Royer, J. Y., Dodhia, R., et al. (2021). Detecting, classifying, and counting blue whale calls with Siamese neural networks. *J. Acoust. Soc Am.* 149, 3086–3094. doi: 10.1121/10.0004828

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 04 August 2023
DOI 10.3389/fmars.2023.1174818

Check for updates

OPEN ACCESS

EDITED BY
Mark C. Benfield,
Louisiana State University, United States

REVIEWED BY
Peng Ren,
China University of Petroleum (East China), China
Jingsong Yang,
Ministry of Natural Resources, China
Jungang Yang,
Ministry of Natural Resources, China

*CORRESPONDENCE
Shengke Wang
neverme@ouc.edu.cn
Guoqiang Zhong
gqzhong@ouc.edu.cn

RECEIVED 27 February 2023
ACCEPTED 14 July 2023
PUBLISHED 04 August 2023

CITATION
Zhao Y, Fan Z, Li H, Zhang R, Xiang W, Wang S and Zhong G (2023) SymmetricNet: end-to-end mesoscale eddy detection with multi-modal data fusion.
Front. Mar. Sci. 10:1174818.
doi: 10.3389/fmars.2023.1174818

COPYRIGHT
© 2023 Zhao, Fan, Li, Zhang, Xiang, Wang and Zhong. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

SymmetricNet: end-to-end mesoscale eddy detection with multi-modal data fusion

Yuxiao Zhao[1], Zhenlin Fan[1], Haitao Li[1], Rui Zhang[2], Wei Xiang[3], Shengke Wang[1*] and Guoqiang Zhong[1*]

[1]College of Computer Science and Technology, Ocean University of China, Qingdao, China, [2]Department of Foundational Mathematics, Xi'an Jiaotong Liver-pool University, Suzhou, China, [3]School of Engineering and Mathematical Sciences, La Trobe University, Melbourne, VIC, Australia

Mesoscale eddies play a significant role in marine energy and matter transportation. Due to their huge impact on the ocean, mesoscale eddy detection has been studied for many years. However, existing methods mainly use single-modal data, such as the sea surface height (SSH), to detect mesoscale eddies, resulting in inaccurate detection results. In this paper, we propose an end-to-end mesoscale eddy detection method based upon multi-modal data fusion. Particularly, we don't only use SSH, but also add data of other two modals, i.e., the sea surface temperature (SST) and the velocity of flow, which are closely related to mesoscale eddy detection. Moreover, we design a novel network named SymmetricNet, which is able to achieve multi-modal data fusion in mesoscale eddy detection. The proposed SymmetricNet mainly contains a downsampling pathway and an upsampling pathway, where the low-level feature maps from the downsampling pathway and the high-level feature maps from the upsampling pathway are merged through lateral connections. In addition, we apply dilated convolutions to the network structure to increase the receptive field without sacrificing resolution. Experiments on multi-modal mesoscale eddy dataset demonstrate the advantages of the proposed method over previous approaches for mesoscale eddy detection.

KEYWORDS

deep learning, mesoscale eddy detection, multi-modal, data fusion, dilated convolutions

1 Introduction

With the development of deep learning (LeCun et al., 2015), many practical problems, such as those in the fields of pattern recognition and computer vision, have been tackled with breakthrough results (Krizhevsky et al., 2012; Sermanet et al., 2014). Among others, semantic segmentation as an important branch of computer vision (Mottaghi et al., 2014; Cordts et al., 2016; Caesar et al., 2018), has benefited from the powerful deep learning models (Everingham et al., 2015). Since fully convolutional networks achieved the state-of-

the-art performance on semantic segmentation (Long et al., 2015), a variety of deep learning approaches have been proposed for semantic segmentation (Chen et al., 2015; Ronneberger et al., 2015; He et al., 2017). Specifically, in this work, we model the mesoscale eddy detection problem from the perspective of semantic segmentation.

Mesoscale eddies (also known as weather-type ocean eddies) refer to ocean eddies with a diameter of 100-300 km and a life span of 2-10 months (Wyrtki et al., 1976; Chelton et al., 2007). They are generally divided into two categories, namely cyclonic eddies (counterclockwise rotation in the northern hemisphere) and anti-cyclonic eddies (counterclockwise rotation in the southern hemisphere). Mesoscale eddies not only play an important role in the transport of energy and particles in the ocean, but also have great effects on the oceanic biological environment. In consequence, there are many pieces of work on mesoscale eddy detection in the literature. Concretely, mesoscale eddy detection is to label the areas in an image where mesoscale eddies exist. However, it is very challenging to build a suitable detection method which can accurately detect the irregular shape of mesoscale eddies.

In the early days, traditional methods based on manual annotation, mathematical or physical knowledge and image processing techniques were used to detect mesoscale eddies. Nichol uses computers to search regions connected by the same gray level value in gray level images (Nichol, 1987), attempting to extract a similar eddy structure from the relationship diagram generated by these regions. Peckinpaugh and Holyer propose a method for eddy detection, which uses the Hough transformation method (Illingworth and Kittler, 1988) based on the edge detection in the remote sensing images (Peckinpaugh and Holyer, 1994). Due to the irregular shape of mesoscale eddies, Ji et al. use ellipse detection to detect mesoscale eddies (Ji et al., 2002). With the inspiration of ellipse detection, Fernandes proposes a new eddy detector which is capable of finding several eddies per satellite image (Fernandes, 2009). With the enrichment of satellite remote sensing data, a number of mesoscale eddy detection methods based on diverse data have been proposed. These mesoscale eddy detection methods can be divided into those using Eulerian data and those using Lagrangian data. For Eulerian data, the main methods are edge detection methods (Canny, 1986), Okubo-Weiss parameter value methods (Isern-Fontanet et al., 2003; Penven et al., 2005; Chelton et al., 2007), wavelet analysis methods based on the vorticity (Doglioli et al., 2007), wind angle methods based on geometric or kinematic characteristics of the flow field (Chaigneau et al., 2008), methods by using sea surface height variation (Chelton et al., 2011; Faghmous et al., 2012) and so on. For Lagrangian data, there are mainly Lagrangian stochastic methods (Lankhorst, 2006), rotation methods (Griffa et al., 2008), spiral trajectory search methods based on geometric features of trajectories (Dong et al., 2011a) and so on. However, these traditional methods have some defects in computational time and detection performance.

In recent years, the success of deep learning in various fields has provided a new paradigm for mesoscale eddy detection (Santana et al., 2022; Yu et al., 2022). Compared with traditional methods, deep learning based methods can extract rich feature information to improve the accuracy of mesoscale eddy detection. Unfortunately, there are not many deep learning based mesoscale eddy detection methods. Among them, Lguensat et al. propose EddyNet on SSH data for pixel-wise classification of eddies (Lguensat et al., 2018), which is a simple network architecture based on the U-Net (Ronneberger et al., 2015). Subsequently, Du et al. propose DeepEddy based on the principal component analysis network (PCANet) (Chan et al., 2015) and spatial pyramid pooling (SPP) (He et al., 2015), achieving a classification of SAR images (Du et al., 2019). Recently, Xu et al. adapt the PSPNet to mesoscale eddy detection (Xu et al., 2019), which is an architecture for semantic segmentation. Duo et al. use bounding boxes to achieve an object detection task for mesoscale eddy detection only based on sea level anomaly (SLA) data, not locating mesoscale eddies accurately by classifying each pixel (Duo et al., 2019). Similar to EddyNet, Santana et al. apply the U-Net model to mesoscale eddy detection based on SSH and SLA (Santana et al., 2020). Moschos et al. propose a deep learning method on SST, only completing a classification task on mesoscale eddy detection similar to DeepEddy (Moschos et al., 2020). Li et al. (2022) proposes a mesoscale detection network based on the extraction of eddy-related spatiotemporal information from multisource remote sensing data. However, there are some drawbacks in these approaches. There is no approach to detect mesoscale eddies using multi-modal data yet. In addition, some tasks such as classification and object detection are not suitable for mesoscale eddy detection, not segmenting mesoscale eddies with irregular shapes. Therefore, we model mesoscale eddy detection as a semantic segmentation problem in this paper. Specifically, we design an end-to-end deep network to detect mesoscale eddies by fusing multi-modal data, leading to improved accuracy over the previous methods.

Except for the methodology, a major challenge in mesoscale eddy detection lies in the fact that there are very few labeled datasets available. To address this problem, we build a multi-modal mesoscale eddy dataset. Specifically, we download the multi-modal data from the same sea area at the same time from the Copernicus Marine Environment Monitoring Service (CMEMS)[1]. The multi-modal data contain the sea surface height (SSH), the sea surface temperature (SST) and the velocity of flow, which can be used for mesoscale eddy detection, either independently or synthetically (Voorhis et al., 1976; Fu et al., 2010; Dong et al., 2011b; Mason et al., 2014). It should be noted that the flow velocity data contains two directions, namely zonal and meridional velocity, because the velocity vector at a certain point in the ocean is decomposed into the east/west direction (zonal) and the north/south direction (meridional). Hence, different from the SSH and SST data which include only one channel, the velocity of flow has two channels. Additionally, due to the extensive use of the SSH data for mesoscale eddies detection, we asked the experts to label the ground truth on the SSH images base on semantic segmentation tool so that it is easy to compare with previous mesoscale eddy detection approaches.

The multi-modal dataset we collected contains different variables affecting mesoscale eddies in the same sea area, so we concatenate four channels occupied by these three multi-modal

1 https://marine.copernicus.eu/

data and input them into the network together for feature learning. In order to fuse multi-modal data for mesoscale eddy detection and reduce the loss of information during feature extraction, we propose a novel deep architecture dubbed SymmetricNet. SymmetricNet mainly consists of a downsampling pathway and an upsampling pathway. Particularly, we combine the low-level feature maps of high resolution from the downsampling pathway and the high-level feature maps with rich semantic information from the upsampling pathway *via* lateral connections. We use element-by-element addition to achieve the fusion of the feature maps, replacing the concatenation of feature maps which is widely used to merge feature maps in previous semantic segmentation approaches. Furthermore, considering that convolutional operations reduce resolution and tend to lose fine-grained information, dilated convolutions (Yu and Koltun, 2016) are used in the upsampling pathway, which can increase the receptive field and aggregate multi-scale contextual information without losing resolution. As a result, the final feature map of our model has not only rich semantics, but also rich contextual information. In contrast to EddyNet and PSPNet, our method makes use of multi-modal data fusion for mesoscale eddy detection. In contrast to DeepEddy, SymmetricNet can locate multiple mesoscale eddies in a sea area, and classify them as cyclonic eddies or anti-cyclonic eddies.

In summary, the main contributions of our work are:

- We construct a mesoscale eddy multi-modal dataset containing the SSH, SST and the velocity of flow. It is annotated by experts based on the SSH images from dataset;
- We propose a novel end-to-end SymmetricNet, which can achieve multi-modal data fusion and mesoscale eddy detection. SymmetricNet is composed mainly of a downsampling pathway and an upsampling pathway, which fuses low-level feature maps from the downsampling pathway with high-level feature maps from the upsampling pathway *via* lateral connections. In addition, we employ dilated convolutions in an effort to increase the receptive field and to obtain more contextual information without losing resolution;
- Our approach outperforms previous methods, achieving excellent performance for mesoscale eddy detection on the multi-modal dataset collected by us.

The rest of this paper is organized as follows. In Section 2, due to the lack of related work, we describe directly the structure of our proposed SymmetricNet and the loss function. In Section 3, we present the constructed multi-modal dataset, the parameter settings used to train our network, and three comparative experiments. In Section 4, we discuss the results of comparative experiments. Section 5 concludes this paper.

2 Materials and methods

In this section, we first introduce the structure of our proposed network dubbed SymmetricNet, which is a symmetric network as shown in Figure 1. We then introduce lateral connections and dilated convolutions applied to SymmetricNet. We use lateral connections to fuse low-level feature maps with high-level feature maps, which replace the concatenation in previous methods with an element-by-element addition. In addition, dilated convolutions are used to increase the receptive field and obtain contextual information. Finally, we describe the loss function for the optimization of SymmetricNet.

2.1 Network architecture

2.1.1 SymmetricNet

Recent semantic segmentation methods usually use the encoder-decoder structure due to its great successes in many applications (Chen et al., 2018). The SymmetricNet is also a symmetric encoder-decoder architecture. As shown in Figure 1, SymmetricNet is composed of a downsampling pathway (left side), an upsampling pathway (right side) and a transition block (in the middle). As can be seen in Figure 1, there are four downsampling blocks in the downsampling pathway and four upsampling blocks in the upsampling pathway. Thus, the architecture of SymmetricNet is symmetric.

In the downsampling pathway, each downsampling block mainly consists of two layers of 3×3 convolution, each followed by a batch normalization (BN) layer and a rectified linear unit (ReLU). Next, a 2 × 2 max pooling operation with a stride of two is employed to each block for downsampling. Furthermore, in order to avoid over-fitting in our network, a dropout layer is applied to the fourth downsampling block. Particularly, the number of channels is doubled when performing a downsampling block. The downsampling pathway can be viewed as that the length and width of the feature maps are halved and the number of channels is doubled when passing a downsampling block. Similarly, in the upsampling pathway, each upsampling block consists of a deconvolutional operation, a lateral connection and a 3×3 dilated convolution with a rate of four. The deconvolutional operation in each upsampling block can double the length and width of the feature maps and halve the number of channels. The lateral connection fuses low-level feature maps from the downsampling pathway with high-level feature maps from the upsampling pathway. Thus, the effect of the upsampling pathway can be viewed as that the length and width of the feature maps are doubled, and the number of channels is halved when passing a upsampling block. Except for these four downsampling blocks and four upsampling blocks, there is a transition block following the fourth downsampling block, which consists of two layers of 3×3 convolution, each followed by a BN layer and an ReLU layer. Similar to the fourth downsampling block, there is a dropout layer at the end of the transition block to avoid over-fitting in SymmetricNet.

In the end, we take the output of the last upsampling block as input into the final softmax layer to achieve pixel-level classification, and finally attain the segmentation results for mesoscale eddy detection.

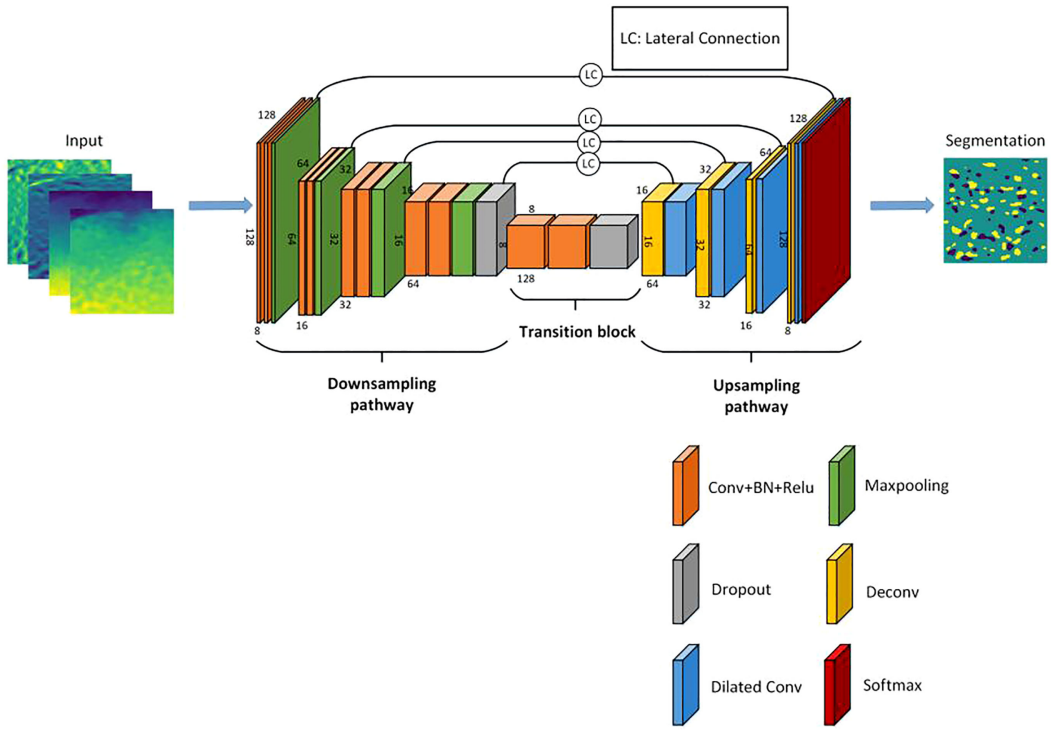


FIGURE 1
Schematics of the proposed SymmetricNet.

2.1.2 Lateral connections and dilated convolutions

In recent years, convolution has become an increasing popular method in deep learning thanks to its effectiveness in extracting rich semantic information from feature maps. However, fine-grained information can be lost by continuously convolutional operations. Although the resolution of feature maps increases when they are upsampled, some important details may be difficult to recover by the deconvolutional operation. Therefore, in SymmetricNet, the low-level feature maps of high resolution are fused with the high-level feature maps to capture fine-grained information lost in the downsampling pathway.

Additionally, dilated convolutions are adopted in our network to replace conventional convolutions so as to avoid the massive loss of contextual information as in conventional convolutional networks. Dilated convolution introduces the dilation rate in an attempt to increase the receptive field at a single pixel, and obtain more contextual information. Figure 2 illustrates the difference between a conventional convolutional kernel and two dilated convolutional kernels. Figure 2A shows the 3×3 convolutional kernel of conventional convolution, whereas Figure 2B and Figure 2C show the 3×3 convolutional kernels of dilated convolutions with a rate of two and four, respectively. The orange areas represent the non-zero parameters of the convolutional kernel, while the white areas represent the parameters filled with zero. There is a gap between the nonzero parameters of the dilated convolutional kernel, which is equal to the dilated rate minus one. It is obvious that the receptive field becomes larger due to the expansion of the convolutional kernel, and the increase of the receptive field results in enriched contextual information. However, the major drawback of dilated convolution lies in its excessive computational complexity and large memory requirement as the size of the dilated convolutional kernel increases. Therefore, we only apply dilated convolutions to the upsampling pathway.

Figure 3 illustrates a lateral connection of the low-level feature maps from the downsampling pathway and the high-level feature maps from the upsampling pathway in detail. Firstly, the high-level feature maps output from the transition block or upsampling blocks are upsampled by a deconvolutional operation. Next, we select the corresponding low-level feature maps in the downsampling pathway according to the size of the high-level feature maps, because the sizes of the feature maps that need to be added must be the same. Then, we apply a 3×3 dilated convolution with a rate of four to the low-level feature maps of high resolution, performing semantic extraction without reducing the resolution. In this case, we can mitigate the disadvantage that the low-level feature maps have weak semantic information. Subsequently, the low-level feature maps of high resolution and the high-level feature maps with rich semantic information are added in an element-by-element manner. Ultimately, a 3×3 dilated convolution with a rate of four is applied to the fused feature maps in an effort to gain multi-scale contextual information, while maintaining the resolution.

The lateral connections between the low-level feature maps of high resolution from the downsampling pathway and the high-level feature maps with rich semantic information from the upsampling

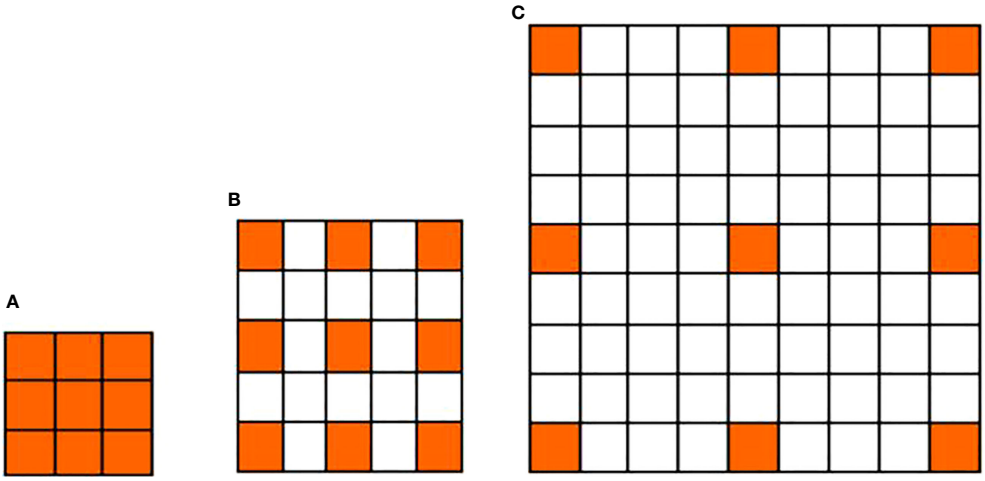


FIGURE 2
Comparison between a conventional convolutional kernel and two dilated convolutional kernels. **(A)** The 3×3 convolutional kernel of conventional convolution; **(B)** The 3×3 convolutional kernel of dilated convolution with a rate of 2; **(C)** The 3×3 convolutional kernel of dilated convolution with a rate of 4.

pathway achieve feature maps fusion, resulting in the feature maps with fine-grained and rich semantic information.

2.2 Loss function

In our work, we use a loss function which combines the dice loss function and the cross-entropy loss function for the optimization of SymmetricNet. It is defined as

$$L(P,G) = -log(1 - DL(P,G)) + L_{log}(P,G), \quad (1)$$

where $DL(P,G)$ the dice loss function, and $L_{log}(P,G)$ is the cross-entropy loss function.

We regard the mesoscale eddy detection problem as a semantic segmentation problem, which is essentially a pixel-level classification problem. The dice loss function is a popular loss function for training pixel-level classification networks, which is a similarity measure function used to calculate the similarity of two samples. Dice loss function is helpful to address the problem of class imbalance in semantic segmentation. Dice loss function combined with cross-entropy loss function can improve the stability of model training. Let us first introduce the dice coefficient which describes the similarity between the prediction and the ground truth. Denote by P the prediction and by G the ground truth. $|P|$ and $|G|$ represent the sums of elements in P and G respectively. Then, the dice coefficient function is defined as

$$DC(P,G) = \frac{2|P \cap G|}{|P| + |G|}. \quad (2)$$

According to the above formula, the prediction and the ground truth are exactly the same when the dice coefficient is one, and the segmentation result is optimal. By contrast, a dice coefficient of 0

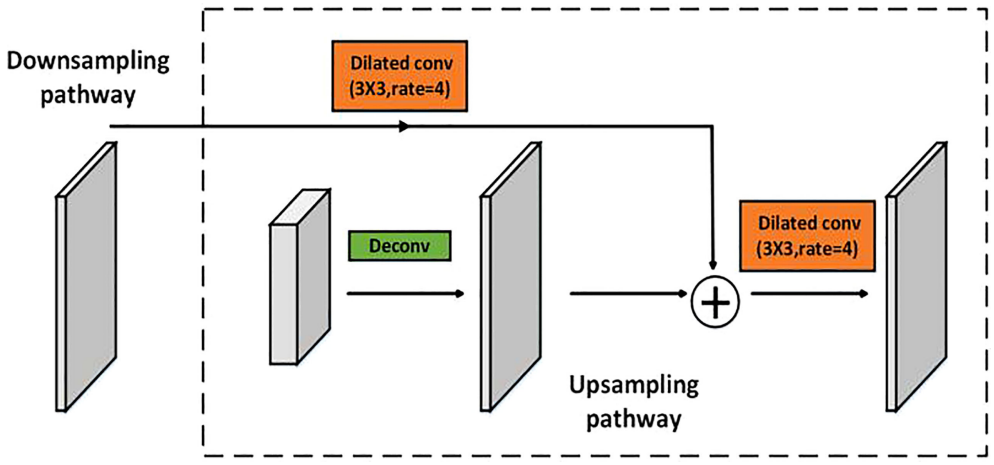


FIGURE 3
Illustration of a lateral connection.

refers to a completely erroneous segmentation result, implying that the prediction and the ground truth do not match at all. In other words, the larger the dice coefficient, the better the performance. As a result, we define the dice loss function as follows:

$$DL(P, G) = 1 - DC(P, G) = 1 - \frac{2|P \cap G|}{|P| + |G|} \quad (3)$$

However, there is a disadvantage in the dice loss function. The gradient of the dice loss function mainly depends on the sum of the elements in the prediction and the ground truth. The gradient will change sharply if it is too small, making the training difficult. Besides, mesoscale eddy detection is actually a 3-class classification problem, i.e., the cyclonic eddies, anti-cyclonic eddies and background classes. Therefore, the cross-entropy loss function can reduce the training difficulty of the network, which is the most commonly used loss function for multi-class classification problems.

In the end, we use the loss function in Eq.(1) to train our network, achieving excellent performance for mesoscale eddy detection.

3 Results

In this section, we first explain the details of the collected dataset. Then, we introduce parameter settings for training the proposed SymmetricNet. Finally, we demonstrate that our method is superior to other methods for mesoscale eddy detection in three aspects, i.e., the comparisons on different modals of data, different networks and different loss functions.

3.1 Dataset

So far, there are very few public datasets available for mesoscale eddy detection. Therefore, it is necessary to build a reliable dataset as the first step. In most papers on mesoscale eddy detection to date, the authors rely mainly on the SSH data for detection, lacking the data of other modals closely related to mesoscale eddy detection. Motivated by this observation, we construct a multi-modal dataset, which is composed of the SSH, SST and the velocity of flow.

Firstly, we download the SSH, SST and the velocity of flow for a total of ten years from January, 2000 to December, 2009 on the website of CMEMS. Specifically, the SSH, SST and the velocity of flow of our dataset are downloaded from the GLOBAL OCEAN ENSEMBLE PHYSICS REANALYSIS product, where the spatial resolution is 0.25 degree × 0.25 degree. The dimension of these three-modal data is 681 × 1440, where 681 is the dimension of the latitude, and 1440 is the dimension of the longitude. There is one datum for each month, such that there are 120 data coming from 120 consecutive months. h/south direction (meridional). Then, we choose the data of 40 months for a three-month interval of totally 120 months in order to make the data to be diverse. Lastly, we randomly select the data of these three modals from multiple regions with the size of 128 × 128, ensuring that the corresponding positions of the SSH, SST and the velocity of flow are the same.

In this case, the multi-modal data have four channels, where the first channel corresponds to the SSH, the second channel corresponds to the SST, the third and fourth channels correspond to the velocity of flow. Figure 4 shows examples of the SSH, SST and the velocity of flow corresponding to the channels. Considering that previous methods only use the SSH data, experts are invited to label the SSH images as the ground truth to make it easy to conduct comparison. In labeling, the cyclonic eddies are annotated as -1, the anti-cyclonic eddies are annotated as 1, and the background is annotated as 0. The SSH image and the ground truth in a certain sea area are shown in Figure 5. Figure 5A shows the SSH image, while Figure 5B shows the ground truth, where the yellow areas represent the anti-cyclonic eddies, the dark blue areas represent the cyclonic eddies, and the light blue areas represent the area without eddies. In the end, we randomly select 512 and 256 samples as the training and test sets, respectively.

3.2 Parameter settings

In our network, we adopted 8, 16, 32, 64 and 128 convolutional kernels for the 3×3 convolution applied to each downsampling block and intermediate transition block. Symmetrically, the numbers of all convolutional kernels of each upsampling block were taken as 64, 32, 16, and 8, respectively. The dropout in the last downsampling block and the transition block were set to 0.3 and 0.5, respectively. We trained our network using the Adam optimizer, which had an initial learning rate of 1.0×10^{-3} and a minimum learning rate of 1.0×10^{-30}. Additionally, the batch size was set to 8 and the number of epochs was set to 50.

3.3 Comparative experiments

In this section, in order to validate the effectiveness of our constructed multi-modal dataset, our proposed network, and our combined loss function, we conducted comparative experiments on different modals of data, different networks, and different loss functions, respectively.

3.3.1 Results on different modals of data

To study the significance of our constructed multi-modal dataset, we selected the SSH, SST and the velocity of flow separately from the corresponding channel of the multi-modal dataset. Then we trained and tested our network on these three modals of data and multi-modal data. There was no validation set because the amount of the collected data was not very large. Thus, we firstly compared the loss and accuracy on the training set to make an optimistic evaluation of the network. Figure 6 shows the learning curves on three modals of data and multi-modal data of training set, where the green, orange, blue and red curves represent the learning curves by using the SSH, SST, the velocity of flow and multi-modal data, respectively. As can be seen from Figure 6A, the

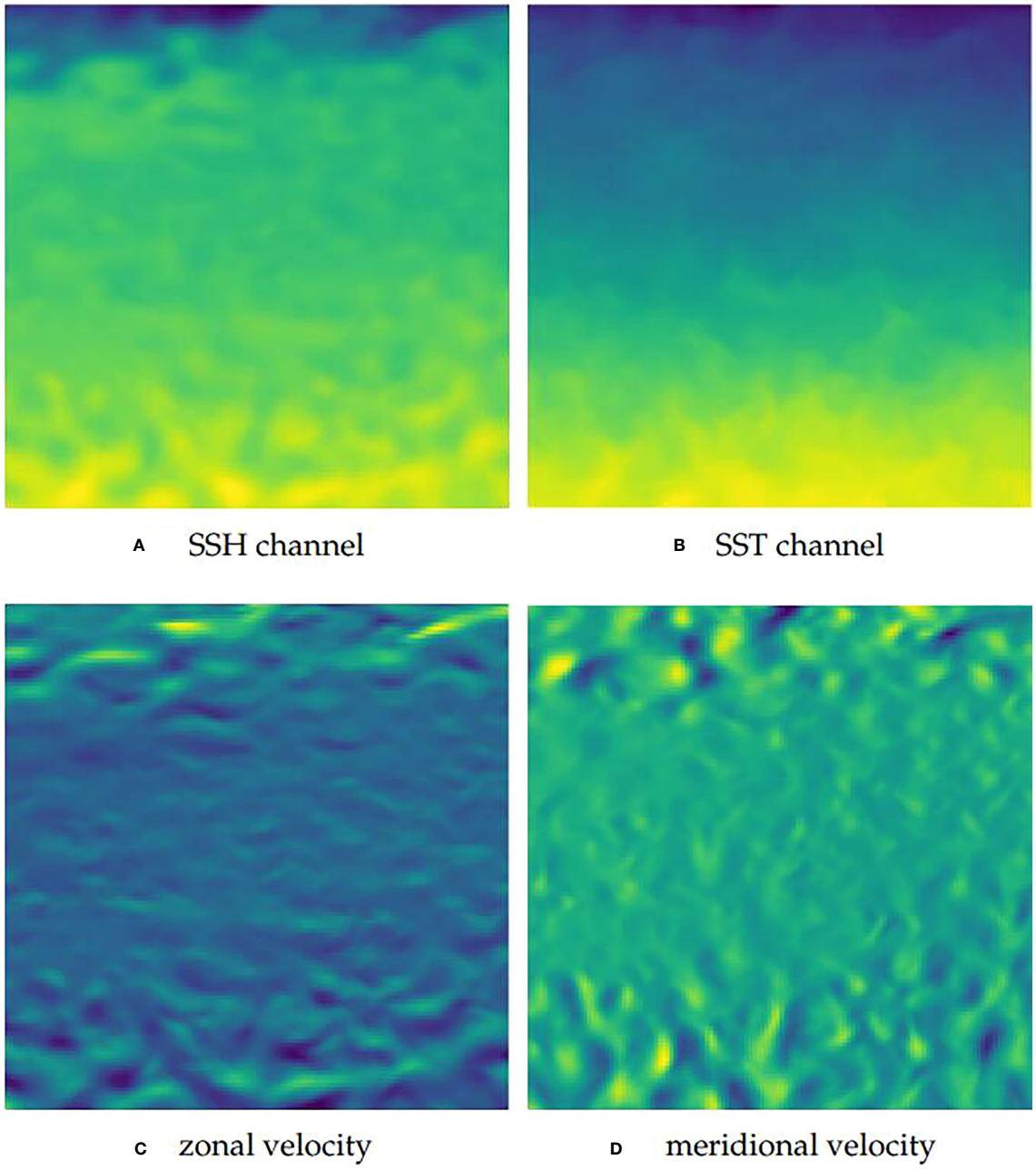


FIGURE 4
Examples of the SSH, SST and the velocity of flow corresponding to the channels of a multi-modal remote sensing image. **(A)** The SSH channel; **(B)** The SST channel; **(C)** The zonal velocity; **(D)** The meridional velocity.

losses gradually decrease and the loss using the multi-modal data is lower than that using other three modals of data as the training epoch increases. Similarly, it is observed in Figure 6B that the accuracy gradually increase and the accuracy using the multi-modal data is higher than that using other three modals of data as the training epoch increases.

Moreover, the loss and accuracy on three modals of data and multi-modal data after training 50 epochs are shown in Table 1, where the cross-entropy loss, dice loss and our loss combining the cross-entropy loss and the dice loss are shown, respectively. As can be seen from the table, the multi-modal data deliver the best results.

We tested our network using three modals of data and multi-modal data of the same test set. In addition to the measure of global accuracy shown in Table 1, we added four evaluation indexes to further prove the effectiveness of our collected multi-modal data, i.e., the pixel precision of cyclonic eddies, anti-cyclonic eddies, background classes and the mean precision of these three classes. Table 2 shows the detection results on three modals of data and multi-modal data, which demonstrate the significance of the multi-modal data. Through these experiments on the training and test sets, we can clearly see that the method based on the multi-modal data outperforms the methods based on the other three modals of data.

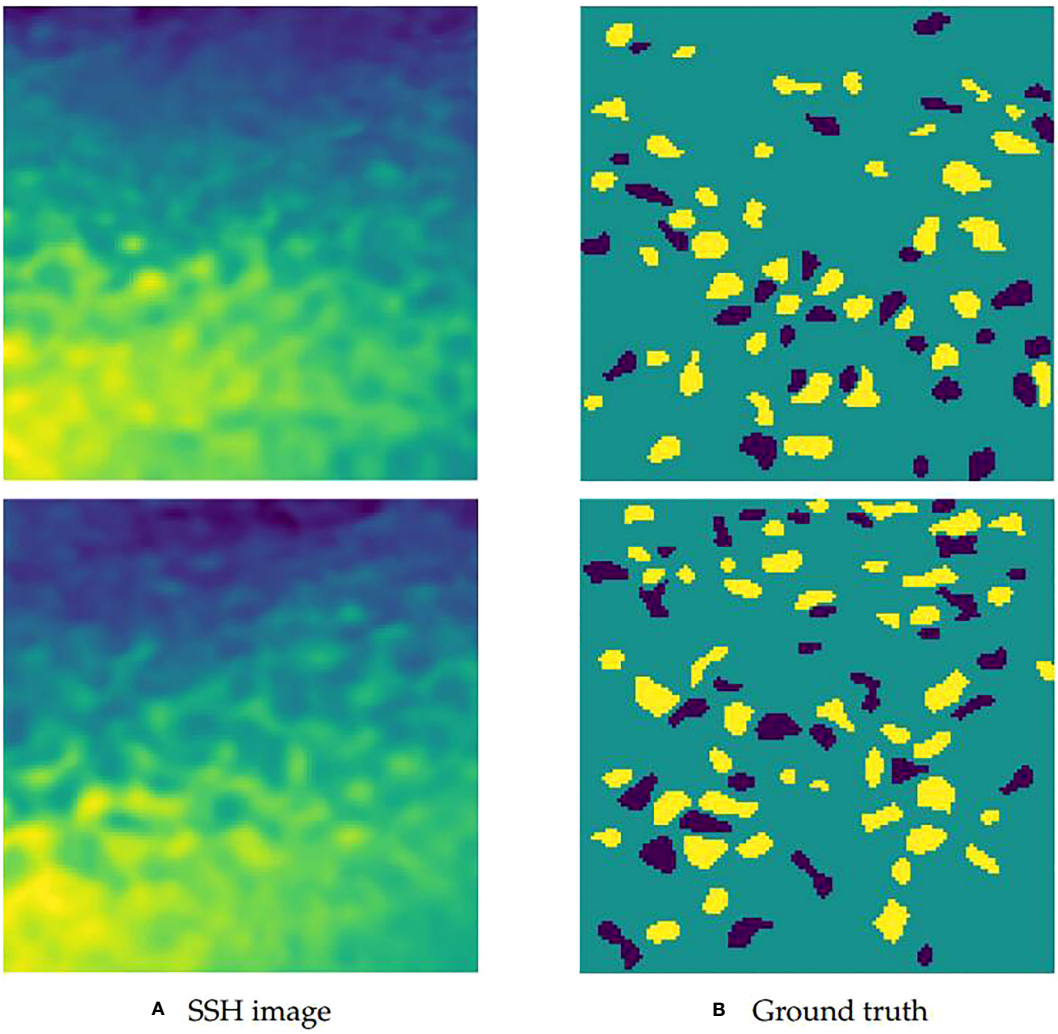


FIGURE 5
Example of the SSH image and the ground truth in a certain sea area. **(A)** Two examples of the SSH images; **(B)** The ground truth labeled by experts according to the SSH images.

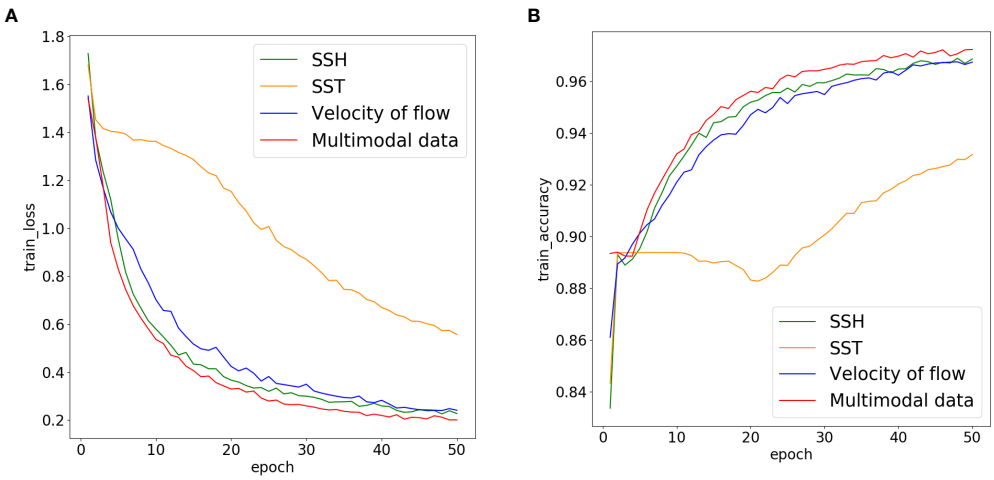


FIGURE 6
Loss and accuracy curves obtained by using our model on three modals of data and multi-modal data. **(A)** Loss curve **(B)** Accuracy curve.

TABLE 1 Loss and accuracy obtained by using our model on three modals of data and multi-modal data of training set.

Dataset	Cross-entropy loss	Dice loss	Our loss	Global accuracy
SSH	0.0935	0.1314	0.2351	96.77%
SST	0.2144	0.2889	0.5565	93.16%
Velocity of flow	0.0940	0.1341	0.2381	96.50%
Multi-modal data	**0.0763**	**0.1076**	**0.1902**	**97.32%**

The best results are highlighted in boldface.

In order to visually demonstrate the advantage of our multi-modal data over three single-modal of data, the examples of eddy detection results using three modals of data and multi-modal data of test set are shown in Figure 7. Through comparing with the ground truth, we can see that mesoscale eddy detection results based on multi-modal data are closest to ground truth.

3.3.2 Results on different networks

The result using the proposed network is compared with those of other networks with the objective of verifying the effectiveness of our framework. As mentioned in Section 1, there are few representative mesoscale eddy detection methods based on deep learning. Among them, DeepEddy, Duo et al. (2019) and Moschos et al. (2020) perform tasks of classification and object detection, and they cannot segment mesoscale eddies from remote sensing images. For methods using pixel-wise classification for mesoscale eddy detection, there are EddyNet, PSPNet and Santana et al. (2020). Considering that both the structure of EddyNet and Santana et al. (2020) rely on the U-Net, we select EddyNet as the representative of them. Hence, we choose to compare EddyNet and PSPNet with our proposed network. Besides, the SymmetricNet without dilated convolution is another compared network to prove the effectiveness of our network, which can also be viewed as an ablation study. Figure 8 shows the learning curves of these compared networks using the multi-modal training set, where the blue, orange, green and red curves represent the learning curves of EddyNet, PSPNet, SymmetricNet without dilated convolution and the proposed network, respectively. Figure 8A shows that the losses of all the models gradually decrease and the loss of our network is lower than that using the other networks as the training epoch increases. Similarly, it is observed from Figure 8B that the accuracy of all the models gradually increase and the accuracy of our network is higher than that using the other networks as the training epoch increases.

Similar to the preceding subsection, we also give detection performances by using the compared networks on the multi-modal training and test sets in Tables 3, 4, respectively. As can be observed from the tables, our proposed network yields best results among compared networks. Examples of eddy detection using the compared networks on the multi-modal test set are shown in Figure 9.

To demonstrate the advantage of the proposed network using our constructed multi-modal dataset more convincingly, the accuracy results obtained by using SymmetricNet and the compared networks on three modals of data and multi-modal data are shown in Table 5. The performance by using the proposed SymmetricNet on our constructed multi-modal data is the best.

3.3.3 Results on different loss functions

In this work, we trained our network with a loss function that combines the dice loss function and the cross-entropy loss function. The dice loss function is popular in semantic segmentation, and the cross-entropy loss function has been widely used on classification problems. In our experiments, we used different loss functions when training the proposed SymmetricNet on the multi-modal dataset to validate the effectiveness of our loss function.

The comparison among the cross-entropy loss function, dice loss function and our combined loss function is shown in Tables 6, 7. As can be seen from the tables, in terms of loss, precision and accuracy, our loss function achieves the best performance.

4 Discussion

In this section, we discuss the results of the comparative experiments in Section 3.3. We show the results of comparative experiments from three aspects, i.e., results on different modals of data, results on different networks and results on different loss

TABLE 2 Detection results obtained by using our model on three modals of data and multi-modal data of test set.

Dataset	Pixel precision			Mean	Global accuracy
	Anti-cyclonic	Cyclonic	Non eddy		
SSH	83.56%	90.16%	97.67%	90.46%	96.69%
SST	74.37%	69.51%	92.70%	78.86%	91.64%
Velocity of flow	82.41%	89.42%	97.63%	89.82%	96.50%
Multi-modal data	**87.85%**	**91.51%**	**98.14%**	**92.5%**	**97.06%**

The best results are highlighted in boldface.

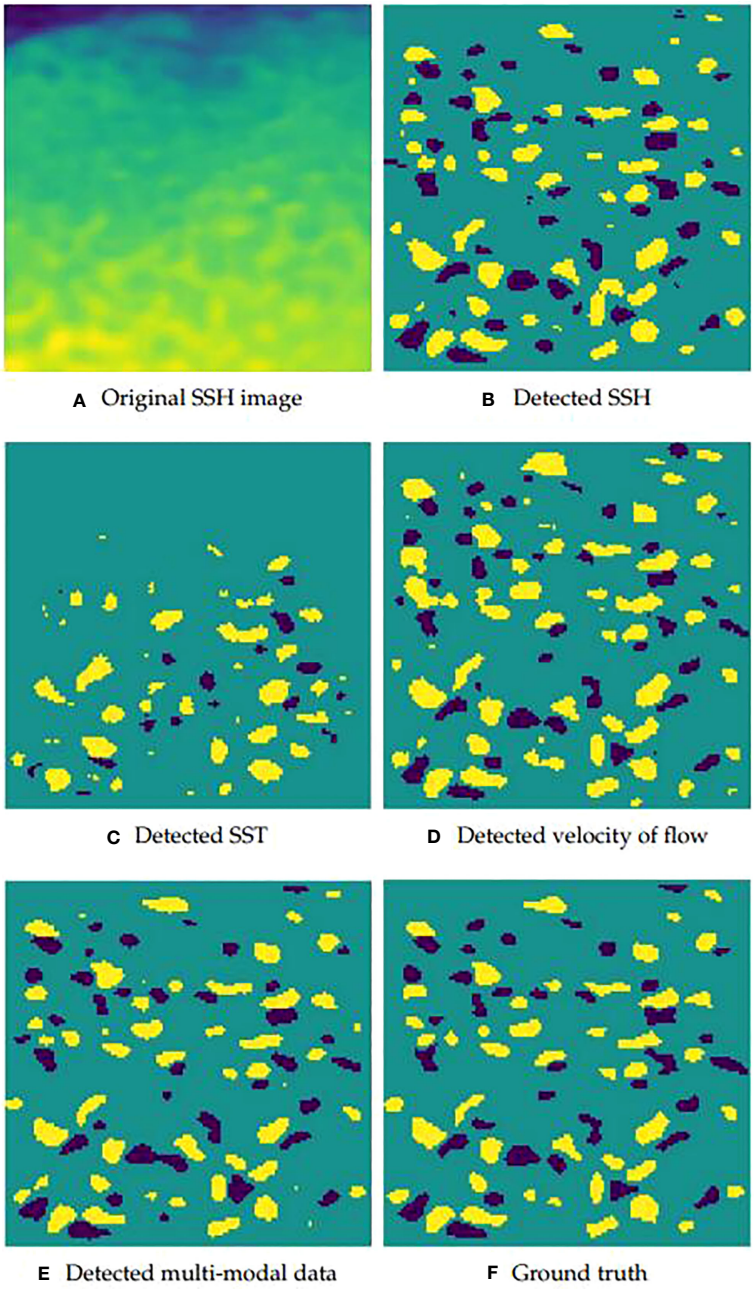


FIGURE 7
Eddy detection results obtained by using the proposed model on three modals of data and multi-modal data of test set. **(A)** Original SSH image in a region of sea; **(B–E)** Eddy detection results from the same region of sea using the SSH, SST, the velocity of flow and the multi-modal data, respectively; **(F)** Ground truth labeled by experts in the same region of sea according to the SSH image.

functions. Here, we firstly discuss the results from two aspects, i.e., the effect of different modals of data, the effect of different networks. The analysis of loss function is introduced in Section 2.2, thus there is no further discussion in this section. Lastly, we discuss future research.

4.1 The effect of different modals of data

In Section 3.3.1, we show results on different modals of data. Firstly, we show the learning curves on three modals of data respectively and multi-modal data in Figure 6, and show the loss and accuracy on three modals of data and multi-modal data after training 50 epochs in Table 1. From Figure 6 and Table 1, we can see that the results based on multi-modal data are significantly better than those based on the three single-modal data. Additionally, one can clearly see the influence of the three modals of data on mesoscale eddy detection. It is evident that the SSH is the most important among the three modals, which has been widely studied in the literature. The velocity of flow also plays a significant role in the research of mesoscale eddy detection. Not only is the

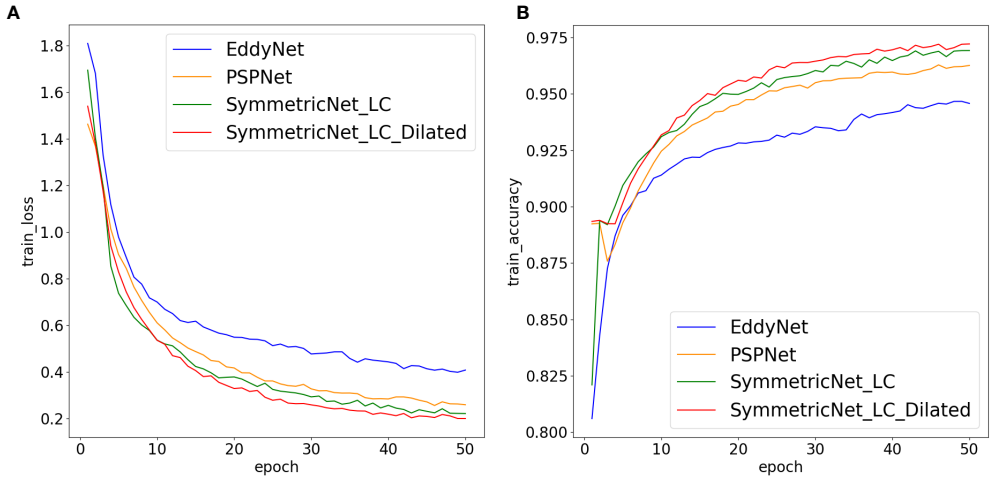


FIGURE 8
Loss and accuracy curves obtained by using different networks on the multi-modal training set. **(A)** Loss curve **(B)** Accuracy curve.

characteristics of the velocity of flow closely related to mesoscale eddies, but it occupies two of the four channels of multi-modal data. In addition, because a large fraction of the ocean satisfies the geotropic balance, the velocity of flow also directly links to the gradient of SSH. Therefore, both of these two modals of data have similar effects on mesoscale eddy detection. In comparison, the effect of SST on mesoscale eddy detection is not as evident as its counterparts. However, because SST can represent mesoscale eddies to a certain extent, it also has a certain positive influence on the mesoscale eddy detection. Therefore, multi-modal data can more comprehensively characterize mesoscale eddies during training, which is beneficial to the improvement of mesoscale eddy detection performance.

After the training, we show comparison results on the test set. In addition to the global accuracy shown in Table 1, we add the precision to further verify the experimental results. Precision refers to the proportion of the number of correctly classified pixels in a category to the number of all pixels predicted to be in this category, which is suitable for mesoscale eddy pixel-by-pixel classification task. Table 2 shows the precision of the three categories, i.e., cyclonic eddies, anti-cyclonic eddies and background. At the same time, we also calculate the mean precision of these three categories to verify the effectiveness of SymmetricNet. Regardless of the precision of a single category, the mean precision or the global accuracy, the results of SymmetricNet on the multi-modal data test set are higher than the results on other single-modal data. For cyclonic eddies and anti-cyclonic eddies which are difficult to detect, SymmetricNet achieved 91.51% and 87.85% precision on the multi-modal test set, which has great improvement compared with the results based on the other three single-modal data.

Lastly, we show Figure 7 to verify the validity of multi-modal data qualitatively. It is clear that the detection result using the SST data misses many eddies and the detection result using the velocity of flow data detects some 'fake' eddies erroneously. Although the detection results based on the SSH data and our multi-modal data are similar, one can assert that the detection result based on our multi-modal data is more accurate than that based on the SSH data in terms of detection details.

4.2 The effect of different networks

In order to prove that SymmetricNet proposed is superior to the current existing methods in mesoscale eddy detection, this paper applies multi-modal data to different networks models to conduct comparative experiments. In this paper, we carry out mesoscale eddy detection from the perspective of semantic segmentation. Thus, the compared methods chosen are EddyNet and PSPNet, which use pixel-by-pixel classification to achieve mesoscale eddy detection. EddyNet and the network proposed by Santana are implemented based on U-Net, the network structures of the two are roughly the same. Therefore, this paper selects EddyNet as the

TABLE 3 Loss and accuracy obtained by using different networks on the multi-modal training set.

Method	Cross-entropy loss	Dice loss	Our loss	Global accuracy
Eddynet	0.1636	0.2168	0.4083	94.58%
PSPNet	0.1015	0.1461	0.2595	96.27%
SymmetricNet (LC)	0.0867	0.1182	0.2126	97.04%
SymmetricNet (LC+Dilated)	**0.0763**	**0.1076**	**0.1902**	**97.32%**

The best results are highlighted in boldface.

TABLE 4 Detection results obtained by using different networks on the multi-modal test set.

Method	Pixel precision			Mean	Global accuracy
	Anti-cyclonic	Cyclonic	Non eddy		
Eddynet	75.48%	80.41%	94.44%	83.44%	93.77%
PSPNet	84.51%	84.44%	97.57%	88.84%	96.25%
SymmetricNet (LC)	87.07%	86.76%	98.01%	90.61%	96.72%
SymmetricNet (LC+Dilated)	**87.85%**	**91.51%**	**98.14%**	**92.5%**	**97.06%**

The best results is highlighted in boldface.

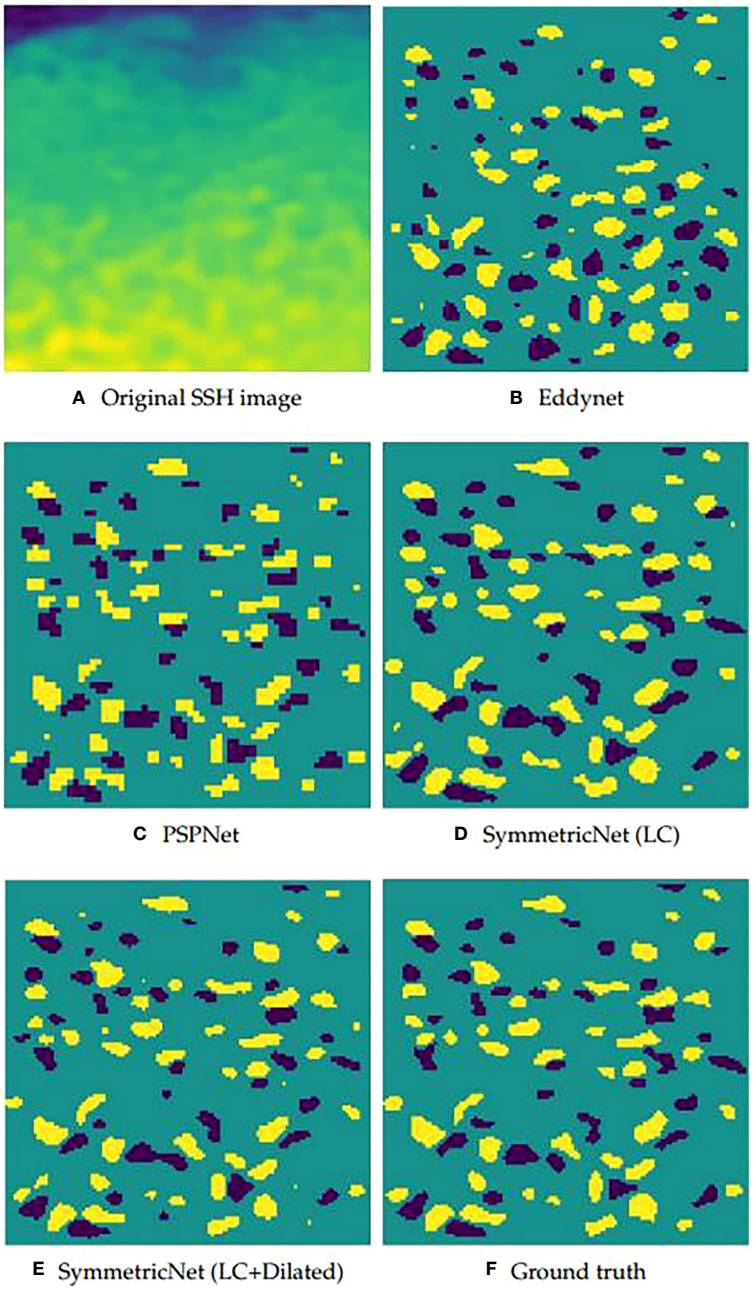


FIGURE 9
Eddy detection results obtained by using different networks on the multi-modal test set. **(A)** SSH image in a region of sea; **(B–E)** Eddy detection results from the same region of sea using the compared networks; **(F)** Ground truth labeled by experts in the same region of sea according to the SSH image.

TABLE 5 Accuracy obtained by using different networks on three modals of data and multi-modal data of test set.

Method	SSH	SST	Velocity of flow	Multi-modal data
Eddynet	93.66%	89.59%	93.55%	93.77%
PSPNet	96.15%	89.80%	95.77%	96.25%
SymmetricNet (LC)	96.37%	89.83%	95.90%	96.72%
SymmetricNet (LC+Dilated)	96.69%	91.64%	96.50%	**97.06%**

The best result is highlighted in boldface.

representative of the two. In addition, this paper replaces the dilated convolutions in SymmetricNet with the traditional convolutions, and uses it as a compared network to verify the effectiveness of dilated convolutions. Figure 8 shows loss and global accuracy curves of different networks on the multi-modal training set, Table 3 shows loss and global accuracy of different networks after training 50 epochs on multi-modal data. Global accuracy, precision of different categories and average precision of different networks on the multi-modal test set are shown in Table 4. We can see that the results of mesoscale eddy detection based on SymmetricNet are better than those obtained based on other comparative networks no matter in the process of training or testing. The number of convolutional layers of EddyNet is relatively shallow, which leads to insufficient feature extraction. Although the number of network layers of PSPNet are relatively deep, the downsampling scale of the pyramid pooling module in the network is large, resulting in serious information loss. Additionally, dilated convolutions can expand the receptive field to obtain more contextual information. Therefore, these comparative networks have poor performance on mesoscale eddy detection compared with SymmetricNet.

In addition to using quantitative indicators to verify the effectiveness of SymmetricNet proposed in this study, Figure 9 compares the results of mesoscale eddy detection based on different networks from a qualitative perspective. Apparently, the detection result of our method is the closest to the ground truth. However, EddyNet misses a lot of eddies, PSPNet locates eddies inaccurately, and SymmetricNet without dilated convolution detects some 'fake' eddies.

Table 5 shows the global accuracy of different networks based on different modals of data, further proving that the multi-modal data and SymmetricNet improve the mesoscale eddy detection performance. As can be seen from the table, our method is better than the others for all the data used, and the results obtained on our constructed multi-modal data are better than those tested on the individual modals of data for all the networks.

4.3 Future research

In this paper, we collect a multi-modal dataset and design SymmetricNet to detect mesoscale eddies, improving the accuracy of the mesoscale eddy detection. However, there are some shortcomings in this study, which need further improvement and perfection in future research. In this subsection, future research will be discussed in the following three aspects:

- To solve the problem that only single-modal data are mainly used for mesoscale eddy detection, a multi-modal dataset containing the SSH, SST and the velocity of flow is constructed. In the future study, we will continue to consider other modals of data affecting mesoscale eddies and expand the multi-modal data. In addition, in order to make the network have strong generalization ability, we will also increase the number of samples in the dataset in the future. Furthermore, in the process of data labeling, this study only uses SSH images for annotation. Although the annotation based on the SSH images is helpful for comparison with existing methods only using SSH, this study cannot output suitable fused feature maps for labeling. Therefore, the future research will find a suitable multi-modal data fused feature map, completing the annotation on the multi-modal data, and make the data match the ground truth.
- In response to the inaccuracy of the mesoscale eddy detection method, this study designs a deep network named SymmetricNet. Although SymmetricNet has achieved relatively good results, there is still room for improvement. The future work will continue to optimize the network. In this paper, we detect mesoscale eddies by pixel-by-pixel classification of ocean remote sensing images. Consequently, in the future research, we will learn ideas from current excellent work in semantic segmentation and improve existing networks to obtain better results.

TABLE 6 Loss and accuracy obtained by using different loss functions on the training set.

Method	Crossentropy loss	Dice loss	Our loss	Global accuracy
Only crossentropy loss	0.0922	–	–	96.31%
Only dice loss	–	0.1149	–	96.69%
Our loss	**0.0763**	**0.1076**	**0.1902**	**97.32%**

The best results are highlighted in boldface.

TABLE 7 Detection results obtained by using different loss functions on the test set.

Method	Pixel precision			Mean	Global accuracy
	Anti-cyclonic	Cyclonic	Non eddy		
Only crossentropy loss	76.82%	80.50%	94.91%	84.07%	
Only dice loss	77.68%	80.61%	96.76%	85.02%	95.60%
Our loss	**87.85%**	**91.51%**	**98.14%**	**92.50%**	**97.06%**

The best results are highlighted in boldface.

- Considering the lifetime of mesoscale eddies, we will detect the eddies trajectories following its path until its disappearance in the future. We think it would be significant to discuss the mesoscale eddy detection from this perspective.

5 Conclusions

In this paper, we construct a multi-modal dataset for mesoscale eddy detection, which contains the SSH, SST and velocity of flow data. Additionally, a new network termed SymmetricNet is proposed, which is capable of fusing multi-modal data to boost the mesoscale eddy detection accuracy. SymmetricNet is capable of fusing low-level feature maps from the downsampling pathway and high-level feature maps from the upsampling pathway *via* lateral connections. In addition, dilated convolution is employed in our proposed network to obtain rich contextual information without losing resolution. To evaluate the constructed multi-modal dataset, our proposed network and the combined loss function, we conduct extensive experiments on different modals of data, different networks and different loss functions. It was demonstrated that the proposed method using our constructed multi-modal dataset outperforms the state-of-the-art existing approaches on mesoscale eddy detection.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work, and approved it for publication.

Funding

This work was partially supported by the National Key Research and Development Program of China under Grant No. 2018AAA0100400, HY Project under Grant No. LZY2022033004, the Natural Science Foundation of Shandong Province under Grants No. ZR2020MF131 and No. ZR2021ZD19, Project of the Marine Science and Technology cooperative Innovation Center under Grant No. 22-05-CXZX-04-03-17, the Science and Technology Program of Qingdao under Grant No. 21-1-4-ny-19-nsh, and Project of Associative Training of Ocean University of China under Grant No. 202265007.

Acknowledgments

We want to thank “Qingdao AI Computing Center” and “Eco-Innovation Center” for providing inclusive computing power and technical support of MindSpore during the completion of this paper.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Caesar, H., Uijlings, J., and Ferrari, V. (2018). “Coco-stuff: thing and stuff classes in context,” in *Proc. IEEE computer vision and pattern recognition* (Salt Lake, UT: IEEE), 1209–1218.

Canny, J. F. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 8, 679–698. doi: 10.1109/TPAMI.1986.4767851

Chaigneau, A., Gizolme, A., and Grados, C. (2008). Mesoscale eddies off Peru in altimeter records: identification algorithms and eddy spatio-temporal patterns. *Prog. Oceanography* 79, 106–119. doi: 10.1016/j.pocean.2008.10.013

Chan, T.-H., Jia, K., Gao, S., Lu, J., Zeng, Z., and Ma, Y. (2015). Pcanet: a simple deep learning baseline for image classification? *IEEE Trans. Image Process.* 24, 5017–5032. doi: 10.1109/TIP.2015.2475625

Chelton, D. B., Schlax, M. G., and Samelson, R. M. (2011). Global observations of nonlinear mesoscale eddies. *Prog. Oceanography* 91, 167–216. doi: 10.1016/j.pocean.2011.01.002

Chelton, D. B., Schlax, M. G., Samelson, R. M., and de Szoeke, R. A. (2007). Global observations of large oceanic eddies. *Geophys. Res. Lett.* 34. doi: 10.1029/2007GL030812

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2015). "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *Proc. International Conference on Learning Representations*, (San Diego, CA: OpenReview).

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. European Conference on Computer Vision*, (Munich, Germany: Springer). 833–851.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Computer Vision and Pattern Recognition*, (Las Vegas, NV: IEEE). 3213–3223.

Doglioli, A., Blanke, B., Speich, S., and Lapeyre, G. (2007). Tracking coherent structures in a regional ocean model with wavelet analysis: application to cape basin eddies. *J. Geophys Res: Oceans* 112. doi: 10.1029/2006JC003952

Dong, C., Liu, Y., Lumpkin, R., Lankhorst, M., Chen, D., McWilliams, J. C., et al. (2011a). A scheme to identify loops from trajectories of oceanic surface drifters: an application in the kuroshio extension region. *J. Atmospheric Oceanic Technol.* 28, 1167–1176. doi: 10.1175/JTECH-D-10-05028.1

Dong, C., Nencioli, F., Liu, Y., and McWilliams, J. C. (2011b). An automated approach to detect oceanic eddies from satellite remotely sensed sea surface temperature data. *IEEE Geosci. Remote Sens. Lett.* 8, 1055–1059. doi: 10.1109/LGRS.2011.2155029

Du, Y., Song, W., He, Q., Huang, D., Liotta, A., and Su, C. (2019). Deep learning with multi-scale feature fusion in remote sensing for automatic oceanic eddy detection. *Inf. Fusion* 49, 89–99. doi: 10.1016/j.inffus.2018.09.006

Duo, Z., Wang, W., and Wang, H. (2019). Oceanic mesoscale eddy detection method based on deep learning. *Remote Sens.* 11, 1921. doi: 10.3390/rs11161921

Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vision* 111, 98–136. doi: 10.1007/s11263-014-0733-5

Faghmous, J. H., Styles, L., Mithal, V., Boriah, S., Liess, S., Kumar, V., et al. (2012). "Eddyscan: a physically consistent ocean eddy monitoring application," in *Proc. IEEE Conference on Intelligent Data Understanding*, (Boulder, CO: IEEE). 96–103.

Fernandes, A. M. (2009). Study on the automatic recognition of oceanic eddies in satellite images by ellipse center detection - the iberian coast case. *IEEE Trans. Geosci. Remote Sens.* 47, 2478–2491. doi: 10.1109/TGRS.2009.2014155

Fu, L.-L., Chelton, D. B., Le Traon, P.-Y., and Morrow, R. (2010). Eddy dynamics from satellite altimetry. *Oceanography* 23, 14–25. doi: 10.5670/oceanog.2010.02

Griffa, A., Lumpkin, R., and Veneziani, M. (2008). Cyclonic and anticyclonic motion in the upper ocean. *Geophys. Res. Lett.* 35. doi: 10.1029/2007GL032100

He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). "Mask r-cnn," in *Proc. IEEE International Conference on Computer Vision*, (Venice, Italy: IEEE). 2980–2988.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 1904–1916. doi: 10.1109/TPAMI.2015.2389824

Illingworth, J., and Kittler, J. (1988). A survey of the hough transform. *Comput. Vision Graphics Image Process.* 44, 87–116. doi: 10.1016/S0734-189X(88)80033-1

Isern-Fontanet, J., Garc´ıa-Ladona, E., and Font, J. (2003). Identification of marine eddies from altimetric maps. *J. Atmospheric Oceanic Technol.* 20, 772–778. doi: 10.1175/1520-0426(2003)20<772:IOMEFA>2.0.CO;2

Ji, G., Chen, X., Huo, Y., and Jia, T. (2002). A automatic detection method for mesoscale eddies in ocean remote sensing image. *Ocean Lake* 33, 139–144.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Proc. Adv. Neural Inf. Process. Syst.* 25, 1106–1114. doi: 10.5555/2999134.2999257

Lankhorst, M. (2006). A self-contained identification scheme for eddies in drifter and float trajectories. *J. Atmospheric Oceanic Technol.* 23, 1583–1592. doi: 10.1175/JTECH1931.1

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

Lguensat, R., Sun, M., Fablet, R., Tandeo, P., Mason, E., and Chen, G. (2018). "Eddynet: a deep neural network for pixel-wise classification of oceanic eddies," in *Proc. IEEE International Geoscience and Remote Sensing Symposium*, (Valencia, Spain: IEEE). 1764–1767.

Li, B., Tang, H., Ma, D., and Lin, J. (2022). A dual-attention mechanism deep learning network for mesoscale eddy detection by mining spatiotemporal characteristics. *J. Atmospheric Oceanic Technol.* 39, 1115–1128. doi: 10.1175/JTECH-D-21-0128.1

Long, J., Shelhamer, E., and Darrell, T. (2015). "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Computer Vision and Pattern Recognition*, (Boston, MA: IEEE). 3431–3440.

Mason, E., Pascual, A., and McWilliams, J. C. (2014). A new sea surface height–based code for oceanic mesoscale eddy tracking. *J. Atmospheric Oceanic Technol.* 31, 1181–1188. doi: 10.1175/JTECH-D-14-00019.1

Moschos, E., Stegner, A., Schwander, O., and Gallinari, P. (2020). Classification of eddy sea surface temperature signatures under cloud coverage. *IEEE J. Selected Topics Appl. Earth Observations Remote Sens.* 13, 3437–3447. doi: 10.1109/JSTARS.2020.3001830

Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., et al. (2014). The role of context for object detection and semantic segmentation in the wild. In Proc. *IEEE Comput. Vision Pattern Recognition Columbus OH.*, 891–898. doi: 10.1109/CVPR.2014.119

Nichol, D. G. (1987). Autonomous extraction of an eddy-like structure from infrared images of the ocean. *IEEE Trans. Geosci. Remote Sens.* GE-25, 28–34. doi: 10.1109/TGRS.1987.289778

Peckinpaugh, S. H., and Holyer, R. J. (1994). Circle detection for extracting eddy size and position from satellite imagery of the ocean. *IEEE Trans. Geosci. Remote Sens.* 32, 267–273. doi: 10.1109/36.295041

Penven, P., Echevin, V., Pasapera, J., Colas, F., and Tam, J. (2005). Average circulation, seasonal cycle, and mesoscale dynamics of the Peru current system: a modeling approach. *J. Geophys. Res. Oceans* 110.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: convolutional networks for biomedical image segmentation," in *Proc. Medical Image Computing and Computer-Assisted Intervention*, (Munich, Germany: Springer). 234–241.

Santana, O. J., Hernandez-Sosa, D., Martz, J., and Smith, R. N. (2020). Neural network training for the detection and classification of oceanic mesoscale eddies. *Remote Sens.* 12, 2625. doi: 10.3390/rs12162625

Santana, O. J., Hernandez-Sosa, D., and Smith, R. N. (2022). Oceanic mesoscale eddy detection and convolutional neural network complexity. *Int. J. Appl. Earth Observation Geoinformation* 113, 102973. doi: 10.1016/j.jag.2022.102973

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and Le Cun, Y. (2014). "Overfeat: integrated recognition, localization and detection using convolutional networks," in *Proc. International Conference on Learning Representations*, (Banff, Canada: OpenReview).

Voorhis, A. D., Schroeder, E. H., and Leetmaa, A. (1976). The influence of deep mesoscale eddies on sea surface temperature in the north atlantic subtropical convergence. *J. Phys. Oceanography* 6, 953–961. doi: 10.1175/1520-0485(1976)006<0953:TIODME>2.0.CO;2

Wyrtki, K., Magaard, L., and Hager, J. (1976). Eddy energy in the oceans. *J. Geophysical Res.* 81, 2641–2646. doi: 10.1029/JC081i015p02641

Xu, G., Cheng, C., Yang, W., Xie, W., Kong, L., Hang, R., et al. (2019). Oceanic eddy identification using an ai scheme. *Remote Sens.* 11, 1349. doi: 10.3390/rs11111349

Yu, F., and Koltun, V. (2016). "Multi-scale context aggregation by dilated convolutions," in *Proc. International Conference on Learning Representations*. (San Juan, Puerto Rico: OpenReview).

Yu, F., Qi, J., Jia, Y., and Chen, G. (2022). Evaluation of hy-2 series satellites mapping capability on mesoscale eddies. *Remote Sens.* 14, 4262. doi: 10.3390/rs14174262

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 15 August 2023
DOI 10.3389/fmars.2023.1146333

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, College Park,
United States

REVIEWED BY
Qiang Guo,
China University of Mining and
Technology, China
Hanqi Zhuang,
Florida Atlantic University, United States
Wenbin Xiao,
National University of Defence Technology,
China

*CORRESPONDENCE
Feng Yin
yinfeng@cuhk.edu.cn

RECEIVED 17 January 2023
ACCEPTED 10 July 2023
PUBLISHED 15 August 2023

CITATION
Huang W, Li D, Zhang H, Xu T and
Yin F (2023) A meta-deep-learning
framework for spatio-temporal
underwater SSP inversion.
Front. Mar. Sci. 10:1146333.
doi: 10.3389/fmars.2023.1146333

COPYRIGHT
© 2023 Huang, Li, Zhang, Xu and Yin. This is
an open-access article distributed under the
terms of the Creative Commons Attribution
License (CC BY). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that
the original publication in this journal is
cited, in accordance with accepted
academic practice. No use, distribution or
reproduction is permitted which does not
comply with these terms.

A meta-deep-learning framework for spatio-temporal underwater SSP inversion

Wei Huang[1], Deshi Li[2], Hao Zhang[1], Tianhe Xu[3] and Feng Yin[4]*

[1]School of Electronic Engineering, Faculty of Information Science and Engineering, Ocean University of China, Qingdao, China, [2]Electronic Information School, Wuhan University, Wuhan, China, [3]School of Space Science and Physics, Shandong University (Weihai), Weihai, China, [4]School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), Shenzhen, China

Sound speed distribution, represented by a sound speed profile (SSP), is of great significance because the nonuniform distribution of sound speed will cause signal propagation path bending with Snell effect, which brings difficulties in precise underwater localization such as emergency rescue. Compared with conventional SSP measurement methods via the conductivity-temperature-depth (CTD) or sound-velocity profiler (SVP), SSP inversion methods leveraging measured sound field information have better real-time performance, such as matched field process (MFP), compressed sensing (CS) and artificial neural networks (ANN). Due to the difficulty in measuring empirical SSP data, these methods face with over-fitting problem in few-shot learning that decreases the inversion accuracy. To rapidly obtain accurate SSP, we propose a task-driven meta-deep-learning (TDML) framework for spatio-temporal SSP inversion. The common features of SSPs are learned through multiple base learners to accelerate the convergence of the model on new tasks, and the model's sensitivity to the change of sound field data is enhanced via meta training, so as to weaken the over-fitting effect and improve the inversion accuracy. Experiment results show that fast and accurate SSP inversion can be achieved by the proposed TDML method.

KEYWORDS

sound speed profile (SSP) inversion, artificial neural networks (ANN), few-shot learning, task-driven meta-learning (TDML), over-fitting effect

1 Introduction

Underwater acoustic wave has become the most popular signal carrier in underwater wireless sensor networks (UWSNs) because of its smaller attenuation and better long-distance propagation performance compared with radio or optical signal by Erol-Kantarci et al. (2011). However, unlike terrestrial radio, underwater sound speed has significant spatio-temporal variability due to the influence of temperature, salinity, pressure by Jensen et al. (2011). This variability will lead to significant Snell effects, which is reflected in the bending of signal propagation path. The bending path brings difficulties for accurate sonar

ranging according to Dinn et al. (1995) and localization according to Isik and Akan (2009); Carroll et al. (2014); Liu et al. (2015); Wu and Xu (2017) in underwater applications such as target detection and rescue. Nevertheless, if the sound speed distribution is obtained, the signal propagation trajectory can be estimated for correcting ranging and positioning errors, which is of great significance for localization applications.

The sound speed distribution of a certain region is usually represented by a sound speed profile (SSP), which is intuitively expressed as a function of sound speed with depth. During the past decades, SSP inversion methods have been widely adopted in underwater wireless sensor networks for estimating sound speed distribution by leveraging sound field information such as time of arrival (TOA) and received signal strength indication (RSSI). The research of novel SSP inversion methods is very promising because they are more automatic and less labor-time-consuming than direct measurement of sound speed by sound velocity profiler (SVP) or conductivity-temperature-depth (CTD) systems refer to Zhang et al. (2015); Huang et al. (2018).

The SSP inversion is a difficult work because the classical ray tracing theory by Munk and Wunsch (1979) and normal mode theory by Munk and Wunsch (1983); Shang (1989) only establish the one-way mapping from ocean environmental information to sound field information, while to the best of our knowledge, there has been no empirical formula for the reverse mapping. Representative works of SSP inversion includes matching field processing (MFP) by Tolstoy et al. (1991), compressed sensing (CS) by Choo and Seong (2018); Li et al. (2019) and artificial neural networks (ANN) by Stephan et al. (1995); Huang et al. (2018). With the same degree of inversion accuracy when there are enough training data, the ANN outperforms MFP and CS in real-time performance due to the fact that after ANN converges, the SSP can be obtained through only once forward propagation by feeding measured sound field information, while iterative processes are ineluctable in MFP and CS based methods for searching the coefficients of principal components decomposed by the empirical orthogonal function (EOF).

For learning-based SSP inversion methods such as ANN, two conditions need to be satisfied: 1) training data and testing data should be taken from a same domain that is independent and identical distribution (*i.i.d.*) refer to Weiss et al. (2016); 2) there should be enough training data to avoid over-fitting problem. However, these two conditions are hard to be met at the same time because of two reasons. First, there are obvious spatio-temporal differences in the distribution and shape of SSPs as shown in Figures 1, 2, so SSPs sampled in different regions and time periods can not be used together as training data for a certain task. Second, due to the high labor and economic cost in measuring SSPs through SVP or CTD systems, SSPs are collected non-uniformly in different regions and time periods, leading to insufficient training SSPs in the spatio-temporal intervals that those tasks belong to. When training the learning model on a small dataset, which is called few-shot learning, there would be over-fitting problem (weak generalization performance), so that the inversion accuracy can not be guaranteed.

For accurately estimating the sound speed distribution in a random ocean area, there are still two important problems to be solved: how to maintain good generalization ability of the inversion model especially in few-shot learning situations, and how to select appropriate reference SSPs for an inversion task to satisfy the *i.i.d.* condition without knowing the actual sound-speed distribution of the task. Many approaches have been proposed to deal with the overfitting problem, such as regularization by Goodfellow et al. (2016), training dataset expanding with generative adversarial networks by Jin et al. (2020), and meta-learning approach by Finn et al. (2017). Regularization establishes a way to limit the model scale by narrowing down the values of weight parameters (L2 norm) or making the model parameters sparse (L1 norm). By this way, the ability of fitting complex relationships of the model is weaken so that overfitting problem could be reduced. Training dataset expanding aims to enrich the training dataset that could represent the whole situation of target domain, however, if the original training data concentrates on a small region, the expanded training dataset will not be uniformly distributed in the target domain, thus the model is still prone to be overfitting. Although training dataset expanding could be achieved artificially to balance the distribution of training data, it usually needs a heavy workload.

Meta-learning (ML) is a newly emerging machine learning method that is very suitable for few-shot learning by Vanschoren (2018); Hospedales et al. (2020). Though ML, a learning model gains experience over multiple learning episodes that covering a distribution of related tasks, and uses the experience to improve its future learning performance for a designated task. The 'learning to learn' feature of ML could lead to a variety of benefits such as data and computing efficiency. Currently, many ML frameworks and algorithms have been established in typical fields such as classification by Snell et al. (2017), object detection by Pérez-Rúa et al. (2020) in computer vision, exploration policies by Alet et al. (2020) in robot control, domain adaptation by Cobbe et al. (2019), hyper–parameter optimization by Finn et al. (2017), neural architecture search summarized by Elsken et al. (2019), etc. The model-agnostic meta-learning (MAML) for fast adaptation of deep networks proposed by Finn et al. (2017) establishes a fast training method for deep learning models on few-shot learning tasks, which becomes almost the most famous work of hyper–parameter optimization. Though MAML provides an idea of model optimization, it has inspired the solution of few-shot learning problems in many fields such as meta-reinforcement learning framework by Alet et al. (2020) for exploration issues. Due to the fact that historical SSPs are usually not accompanied by sound field data, the labeled data composed of sound field data and SSPs need to be constructed through ray theory, resulting in the inability to directly adopt meta learning frameworks from other fields into construction of underwater sound speed field. Therefore, it is necessary to establish a more applicable meta learning SSP inversion framework based on the practical problems in underwater SSP inversion.

In this paper, we propose a meta-deep-learning framework for few-shot spatio-temporal SSP inversion named as task-driven meta-learning (TDML), which provides a training strategy that is suitable

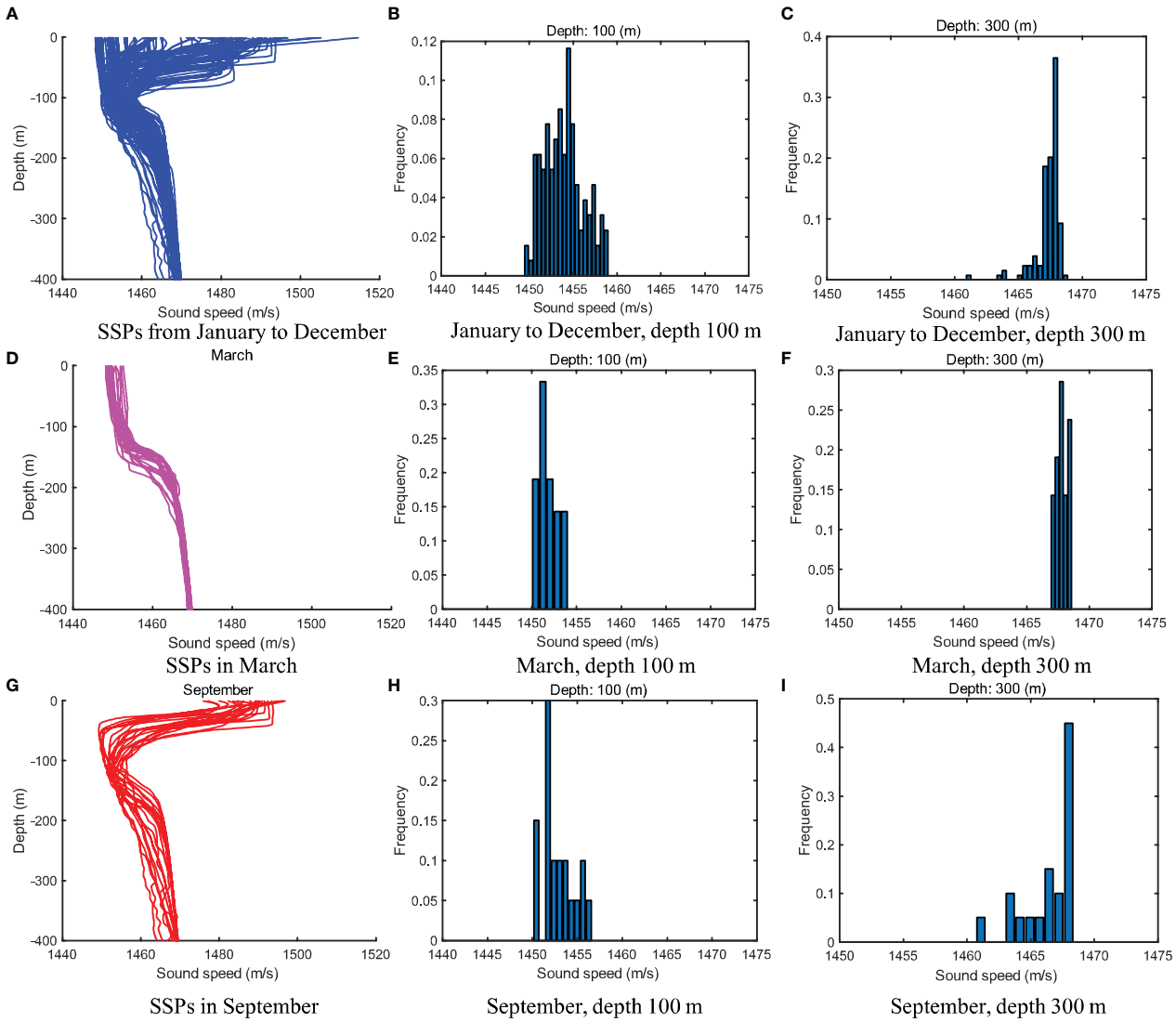


FIGURE 1
Historical SSPs sampled in different months from 40–50°C N and 150–160°C E of the North Pacific.

for multiple types of few-shot dataset learning. The core idea of TDML is to learn the common feature of different kinds of SSPs via a series of base learners, which forms a set of initialization parameters of the task learner. By this means, the convergence rate of the model could be accelerated and the sensibility to the input data could be retained, so that the model will not be over trained on few-shot task samples. The ability of fitting complex relationship or the training dataset is not changed by meta-learning itself, and it could be combined with regularization or training dataset expanding for solving overfitting issues in different applications. To guarantee that the distributions of reference SSPs and the inversion task meet the *i.i.d.* condition, all historical SSPs are first classified into different clusters by a proposed Pearson-correlation-based SSP local density clustering (PC-SLDC) algorithm, then the cluster which the task belongs to is decided by a proposed spatio-temporal-information-based K-nearest neighbor (STI-KNN) mapping algorithm. The contribution of this paper is summarized as follows:

- To accurately obtain sound-speed distribution in a random ocean area under few-shot learning situations, we propose a task-driven meta-deep-learning framework for spatio-temporal SSP inversion.
- To reduce negative transfer effect and deal with the over-fitting problem, we propose a task-driven meta-deep-learning SSP inversion algorithm, in which the updating rate of neuron connection weights could be dynamically adjusted and the convergence of inversion model could be accelerated.
- To satisfy the *i.i.d.* condition and select reference SSPs that possibly has the most similar distribution to the inversion task, we first propose a Pearson-correlation-based SSP local density clustering algorithm for historical SSPs clustering, then propose a spatio-temporal-information-based K-nearest neighbor algorithm for mapping the inversion task to a proper cluster leveraging the spatio-temporal information.

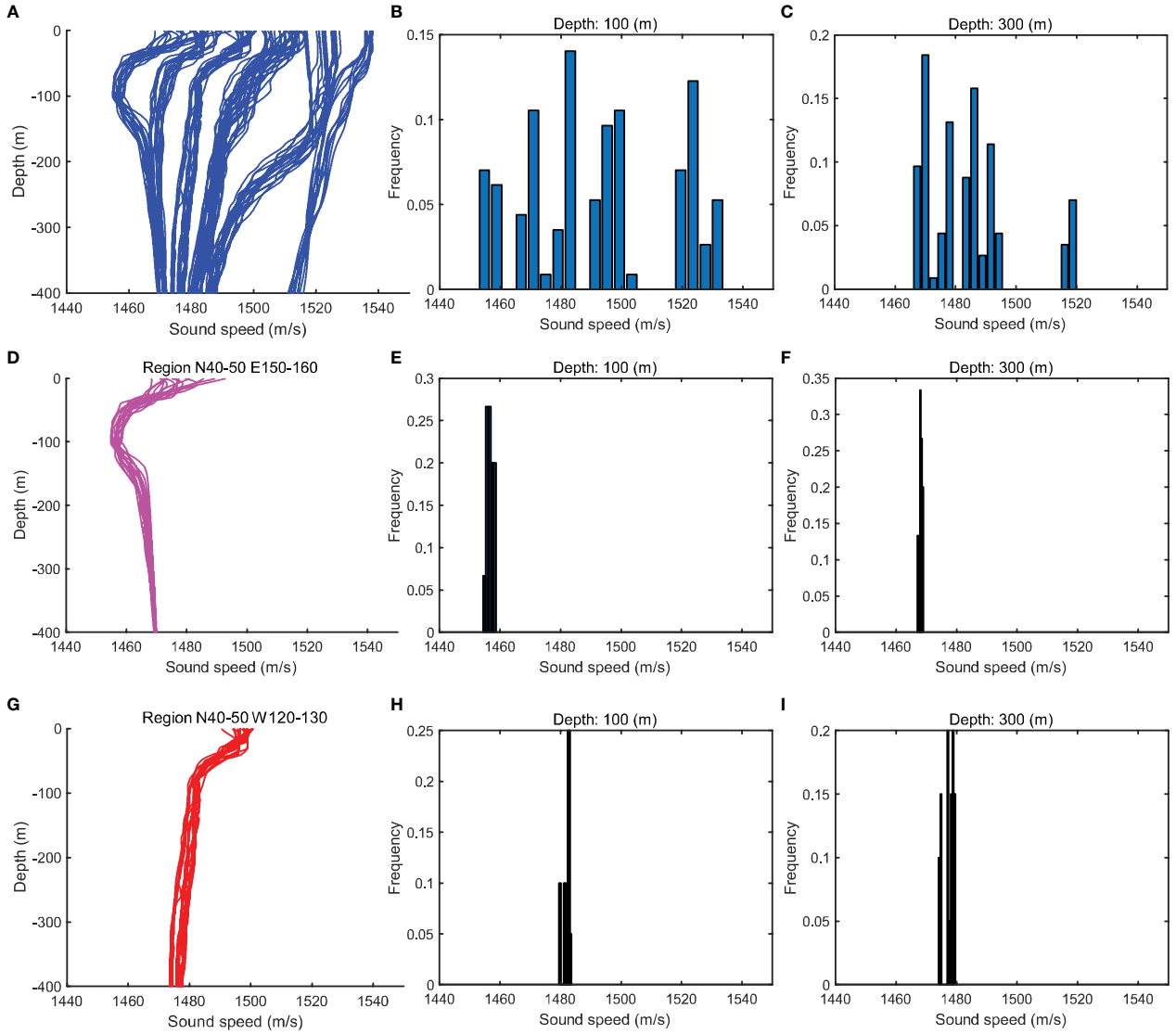


FIGURE 2
Historical SSPs sampled from different regions of the North Pacific Ocean in June (including historical periods).

The rest of this paper is organized as follows. In Sec. 2, we briefly review related works about SSP inversion and few-shot learning. In Sec. 3, the source of input data during training and inversion phase is provided. In Sec. 4, we first propose a TDML framework for spatio-temporal SSP inversion, then present an SSP clustering algorithm and task mapping algorithm to find proper reference SSPs for a specified inversion task. Simulation results are discussed in Sec. 5, and conclusions are given in Sec. 6.

2 Related works

2.1 Underwater SSP inversion

MFP, CS and ANN are three classical SSP inversion methods. In Tolstoy et al. (1991), the MFP technique is first introduced in SSP inversion with four steps: empirical orthogonal decomposition, candidate SSPs generation, simulated sound field calculation and sound field matching, the candidate SSP corresponding to the optimal matching sound field will be recorded as the final inversion result. Instead of reverse mapping from sound field information to sound speed distribution, the purpose of MFP is to find matching principal component coefficients. However, the high time complexity debase the real-time performance of MFP. In Li and Zhang (2010), the coefficient searching space was reduced first, then a traversal method was used to find the optimal solution, while in Li et al. (2015) a parallel grid searching algorithm was proposed to reduce the time consumption. However, the searching accuracy depends on the scanning step, so that the time overhead increases as the accuracy of SSP inversion improves. Heuristic optimization algorithms were introduced in Zhang (2005); Tang and Yang (2006); Zhang et al. (2012); Sun et al. (2016); Zheng and Huang (2017) to speed up the searching process of the optimal EOF coefficients, such as the simulated annealing algorithm in Zhang (2005),

geneticalgorithm in Tang and Yang (2006); Sun et al. (2016), and particle swarm optimization (PSO) algorithm in Zhang et al. (2012); Zheng and Huang (2017). However, to get the optimal result with a high probability, multiple iterations are necessary in these heuristic algorithms. Consequently, the time overhead of SSP inversion can not be reduced to a desired level.

In Li et al. (2019), a mapping relationship is established as a dictionary to describe the effect of small perturbation of principal component coefficients on the change of sound field data. Because the principal component coefficients can be solved directly by the dictionary and sound field data with a few iterations of the least-squares calculation, it can achieve better real-time performance than MFP. Nevertheless, the first-order Taylor expansion approximation for the nonlinear mapping relationship is adopted in the design of the dictionary, so the inversion accuracy is sacrificed to some extent.

Recently, Bianco et al. (2019) presented a detailed review of machine learning applications in the field of acoustic, showing that machine learning technologies have become very promising in ocean parameter estimation, such as seafloor characterization by Michalopoulou et al. (1993), range estimation by Komen et al. (2020), geoacoustic inversion by Piccolo et al. (2019), and SSP inversion by Bianco and Gerstoft (2017). A dictionary learning method is proposed in Bianco and Gerstoft (2017) for SSP inversion that can better explain sound speed variability with fewer coefficients compared with classical EOF decomposition, however, it still requires a lot of time for searching the related dictionary elements and coefficients.

Inspired by the ability of deep neural networks to fit nonlinear functions, we have proposed an ANN-based SSP inversion method in our previous works (Huang et al., 2018; Huang et al., 2021). Through off-line training, the ANN is able to learn the mapping relationship from signal propagation time to sound speed distribution; and during the inversion stage, the SSP can be estimated via once forward propagation by feeding the measured signal propagation time into the SSP inversion model, so the time overhead can be reduced. With enough training data, the ANN can hold a good inversion accuracy while significantly outperforms the MFP and CS in time overhead performance during the inversion stage, which indicates that the deep neural networks are very promising in the SSP inversion fields. However, due to the difficulty of SSP measurement and spatial-temporal distribution of SSP, the neural network model needs to be trained on small dataset in some cases, which is prone to be over-fitting. Therefore, how to deal with the over-fitting problem in few-shot learning is well worth studying.

2.2 Few-shot learning

Conventional deep neural networks are trained from scratch for a given task with lots of training samples. However, in some fields such as SSP inversion, historical data is scarce because of the difficulty in measuring SSPs by CTD or SVP systems, so the model should be able to learn the distribution features of data with only a small amount of samples, which is commonly known as few-shot learning. In this case, the conventional deep neural network will easily fall into over-fitting problem.

To solve the over-fitting problem in few-shot learning, some studies have been done recently as surveyed in Vanschoren (2018); Hospedales et al. (2020). Aiming at few-shot learning on specific tasks, multi-task learning jointly learns several related tasks, and benefits from the effect regularization due to parameter sharing refer to Rich (1997); Yang and Hospedales (2016). Transfer learning (TL) has been developed for few-shot learning in the past decade as surveyed in Weiss et al. (2016); Pan and Yang (2010). TL uses past experience of a source task to improve learning on a new task by transferring the model's parameter prior in Chang et al. (2018) or the feature extractor from the solution of a previous task in Yosinski et al. (2014). Because the TL model is first trained on a specific task, features of the task are memorized in the model, which would affect the learning rate and accuracy for a new task.

Recently, ML surveyed by Vanschoren (2018); Hospedales et al. (2020) has become a promising method for few-shot learning. Different from MTL and TL, a meta-objective is usually defined in ML to evaluate how well the base learner performs when helping to learn a new task. In Ravi and Larochelle (2017), a long short-term memory meta-learner is used to learn an update rule for training a neural network learner. During the training phase, the base learner provides the current gradient and loss to the meta learner, which then update the model parameters. In Finn et al. (2017), a model-agnostic meta-learning algorithm is proposed to learn a model parameter initialization which achieves better generalization performance to similar tasks. The Hessian matrix is illustrated in Finn et al. (2017) for gradient descent, which enhance the sensitivity of the model to the input data. The work of Nichol et al. (2018) further improves the learning rate of model on a new task by executing stochastic gradient decent for several iterations.

The concept of ML would be suitable for dealing with the over-fitting problem of underwater sound speed inversion with only a few reference samples. However, the negative transfer effect caused by training with different kinds of SSPs still needs to be solved so as to improve the inversion accuracy.

3 Preliminary

The SSP inversion is to establish the mapping from signal propagation time to the sound speed distribution. For clearly illustrating the inversion model, it is important to know more about the source of input data. In this section, we will present the signal propagation time measurement method for SSP inversion, and derive the simulated signal propagation time by ray tracing theory corresponding to each historical SSP for training inversion model.

3.1 Signal propagation time measurement

For SSP inversion, accelerating the measurement of signal propagation time is of great important to improve the real-time performance. Thus, the autonomous underwater vehicle (AUV)

assisted signal propagation measurement system proposed in our previous work Huang et al. (2021) is adopted in this paper, which has the advantages of stability and mobility compared with traditional ship-towed or seafloor fixed arrays in Zhang et al. (2015); Choo and Seong (2018); Li and Zhang (2010); Li et al. (2015); Zhang (2005); Tang and Yang (2006); Zhang et al. (2012); Zheng and Huang (2017); Zhang (2013).

The AUVs are able to suspend in the water. One AUV sailing at the bottom of the ocean act as the source node to start the measurement process, the other three AUVs are receivers that sail approximately in the same vertical plane with the bottom AUV and keep a fixed horizontal distance $\Delta\pi_h$ from each other. During once time measurement, the signal travels a round trip, then the clock asynchronization error can be reduced via the bidirectional TOA technology. The idea of virtual anchoring is introduced to increase the amount of measured time data. After one turn of communication, the three AUVs move forward with the distance $\frac{\Delta\pi_h}{\Pi}$ and start a new turn of measurement. After moving $\Pi-1$ times, a signal propagation time sequence containing $3*\Pi$ items can be obtained as the measurement result.

3.2 Signal propagation time simulation

The learning model of SSP inversion is usually trained offline, so the required input signal propagation time can not be measured at the model training stage. Therefore, the classical ray tracing theory is introduced to provide signal propagation time information as input data corresponding to a given SSP for inversion model training.

Assume the preset horizontal distance series of the AUV system is $\Pi=[\pi^1,\pi^2,\ldots,\pi^m], m=1,2,\ldots,3\Pi$ that forms totally $M=3\Pi$ transceiver pairs, then for a given SSP $S=[(s^1,1),\ldots,(s^d,d)]^T$, the relation between Π and S can be expressed according to our previous derivation Huang et al. (2021) as:

$$\begin{aligned}\pi^m &= \frac{s_1}{\cos\ \vartheta^{1,m}}\sum_{d=1}^{D-1}\left|\frac{\Delta z_d}{s^{d+1}-s^d}\left(\sqrt{\Upsilon_d^m}-\sqrt{\Upsilon_{d+1}^m}\right)\right|,\\ \Upsilon_d^m &= 1-\left(\frac{\cos\ \vartheta^{1,m}}{s^1}\right)^2(s^d)^2,\end{aligned} \tag{1}$$

where D is the total depth of the SSP, $\vartheta^{1,m}$ is the initial grazing angle at depth of the first speed point s^1 from source to the mth receiver, and Δz_d is the depth difference of the linear SSP at the dth layer with depth boundaries of d and $d+1$. Referring to (1), the π^m is actually a function of the initial grazing angle ϑ^m, which is not a prior parameter but can be obtained through searching algorithms. The ideal signal propagation time can be simulated according to our previous derivation Huang et al. (2021) by:

$$t^m=\sum_{d=1}^{D-1}\left|\frac{\Delta z_d}{s^{d+1}-s^d}\ln\left(\frac{s^d(1+\sqrt{\Upsilon_{d+1}^m})}{s^{d+1}(1+\sqrt{\Upsilon_d^m})}\right)\right| \tag{2}$$

where the t^m is also a function of the initial grazing angle $\vartheta^{1,m}$.

Actually, the peak detection error of arrival signal, and the position error of AUV will affect the measurement result of signal propagation time refer to Huang et al. (2021), so these errors should be considered to make the simulated signal propagation time more appropriate to the actual situation. Affected by clock asynchronization, environmental noise and multipath effect, the time detection of arrival signal will fluctuate around the real propagation time. It is shown that the measurement errors of signal propagation time are usually converted into range measurement errors that following normal Gaussian distribution according to Zhou et al. (2010); Thomson et al. (2018), with real distance as mean values and standard deviations to be one percent of real distances. The error level is reasonable that can be easily satisfied by existing underwater distance measurement technologies according to Kussat et al. (2005), and the location error could be further reduced by using ray tracing technique in Huang et al. (2019). However, the original time measurement error is adopted in this paper that following normal Gaussian distribution $\omega_c\sim\Omega(\mu_c=0,\sigma_c)$, where μ_c is the mean value and σ_c is the standard deviation. The noisy signal propagation time that fluctuates around the real time value is equivalent to the superpositon of time measurement error with normal distribution on the real signal propagation time, thus the simulated signal propagation time t_ω^m will be:

$$t_\omega^m=t^m+\omega_c\,. \tag{3}$$

For the distance scale about 400-500 meters of the AUV-assisted signal propagation time measurement system, the standard deviation σ_c will be a few miliseconds $(500(m)*0.01/1500(m/s))$.

According to Misra and Enge (2006), the position error of any surface AUV can be expressed as Gaussian distribution $\omega_{sp}^m\sim\Omega(\mu_{sp},\sigma_{sp})$, where μ_{sp} is the mean error and σ_{sp} is the standard deviation. When the geometry topology of satellites is symmetrically and uniformly distributed relative to target at the ocean surface and the system bias of satellites has been corrected, the mean error will follow $\mu_{sp}=0$. To reduce the impact of positioning error of the bottom AUV, there will be a position correction process of the bottom AUV before signal propagation time measurement, which is assisted by the surface AUVs forming a symmetrical topology such as equilateral triangle. In this case, the positioning error of the bottom AUV will also follow a normal distribution $\omega_{bp}\sim\Omega(\mu_{bp}=0,\sigma_{bp})$ according to Thomson et al. (2018), where μ_{bp} is the mean error and σ_{bp} is the standard deviation. However, if the trajectory of the bottom AUV deviates too much, the mean positioning error will not be statistical zero because the surface AUVs could not form a symmetrical distribution relative to the bottom AUV, and the distance measurementerrors caused by using empirical sound speed value will not be spatial averaged.

Considering the positioning errors of AUVs, the simulated horizontal distance series will be $\Pi_\omega=[\pi_\omega^1,\pi_\omega^2,\ldots,\pi_\omega^m], m=1,2,\ldots,3\Pi$, where $\pi_\omega^m=\pi^m+\omega_{bp}+\omega_{sp}^m$. By putting π_ω^m into (1), the initial grazing angle ϑ_ω^m that considering position errors of AUVs can be searched. Then the signal propagation time t_ω^m considering positioning errors can be calculated by (2).

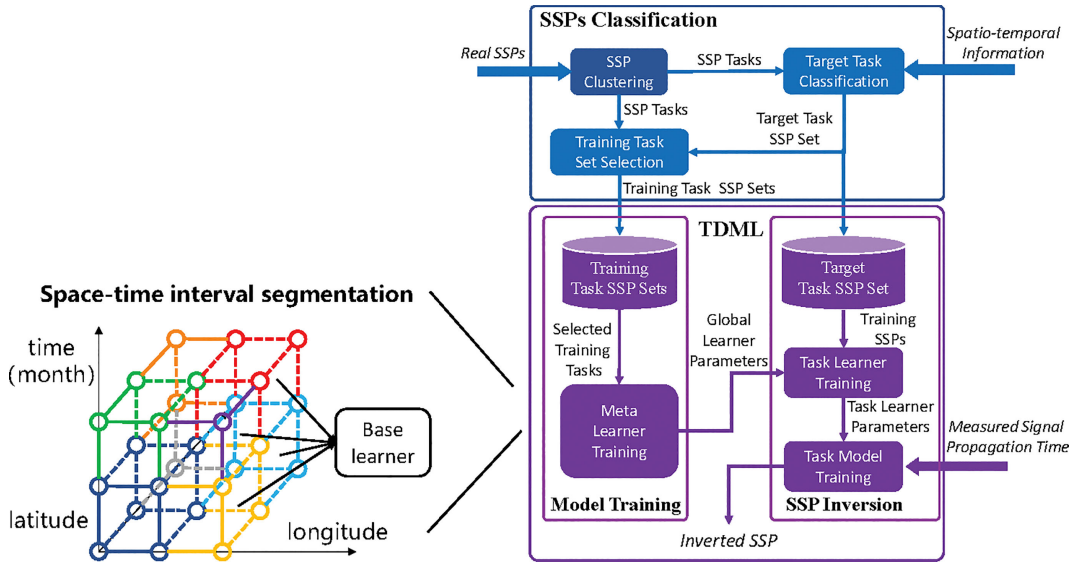


FIGURE 3
Task-driven Meta-learning Framework for SSP inversion.

4 Task-driven meta-learning framework for SSP inversion

Due to the high labor and time costs of SSP measurement with CTD or SVP system, there are usually a few reference SSPs that are similar to the potential distribution of the inversion task. In this case, the learning model is prone to be over-fitting when it is trained on a small dataset, resulting in weak generalization ability and low SSP inversion accuracy. To fast and accurately estimate the regional sound speed distribution with a few reference SSP samples, we propose a TDML framework for spatio-temporal SSP inversion as shown in Figure 3. We aim to learn the common features of different SSP groups through meta learning, that is, to train several base learners on multiple few-shot SSP datasets to collaboratively update the parameters of a global learner, so as to find a good set of initialization parameters for the target task learner. Thereafter, merely a few iterations of training is required to make the task learner converge on the few-shot dataset; meanwhile, the model retains the memory of common features.

Considering the spatio-temporal difference of SSP distribution, the ocean region is divided according to spatio-temporal information. A base learner is established for each region, and different types of SSPs obtained by clustering are also allocated to each spatio-temporal interval according to the spatio-temporal information of the cluster center, which could be used as training data. The spatio-temporal division scales are usually in varied forms, however, in this paper, the space is divided by 1 degree and time is divided by month.

In the proposed TDML framework, several kinds of learning models could be used as the base learner or task learner such as neural networks in Benson et al. (2000); (Huang et al., 2018; Huang et al., 2021) and Gaussian process in Yin et al. (2020). In order to guarantee a good robustness performance, the auto-encoding feature-mapping neural network (AEFMNN) proposed in our early work Huang et al. (2021) is utilized as the base and task learners. When the measured signal propagation time is fed into the trained task learner, the inversion SSP could be quickly obtained with once forward propagation.

4.1 SSPs clustering and task mapping

4.1.1 Pearson-correlation-based SSP local density clustering

The difference of SSP behaves in the variation trend of sound speed values with depth. To obtain SSP clusters with similar distribution, we propose a PC-SLDC algorithm, the structure of which is given in Figure 4. The SSPs distribution in the ocean is continuous, if the clusters of SSPs are divided without overlapping, the task SSP whose real distribution is at the margin of the cluster domain may not be accurately estimated because the reference data in this cluster is not uniformly or symmetrically distributed around the task SSP (as shown in Figure 4), which may lead to overfitting problem. Therefore, its better to cluster SSPs with partly overlapping. In this case, the SSP sample that lays at the margin of a cluster domain may belong to another cluster at the same time.

Euclidean distance has been widely adopted to describe the difference between two SSPs such as Choo and Seong (2018); Zhang et al. (2015)[1], but it can not reflect whether the variation trends with depth of two SSPs are consistent or not, especially for shallow-water SSPs that their gradients may be positive or negative.

1 For SSP with fixed number of sampling points, Euclidean distance is equivalent to mean square error Choo and Seong (2018) and root mean square error Zhang et al. (2015) in describing vector difference, and they are positively correlated.

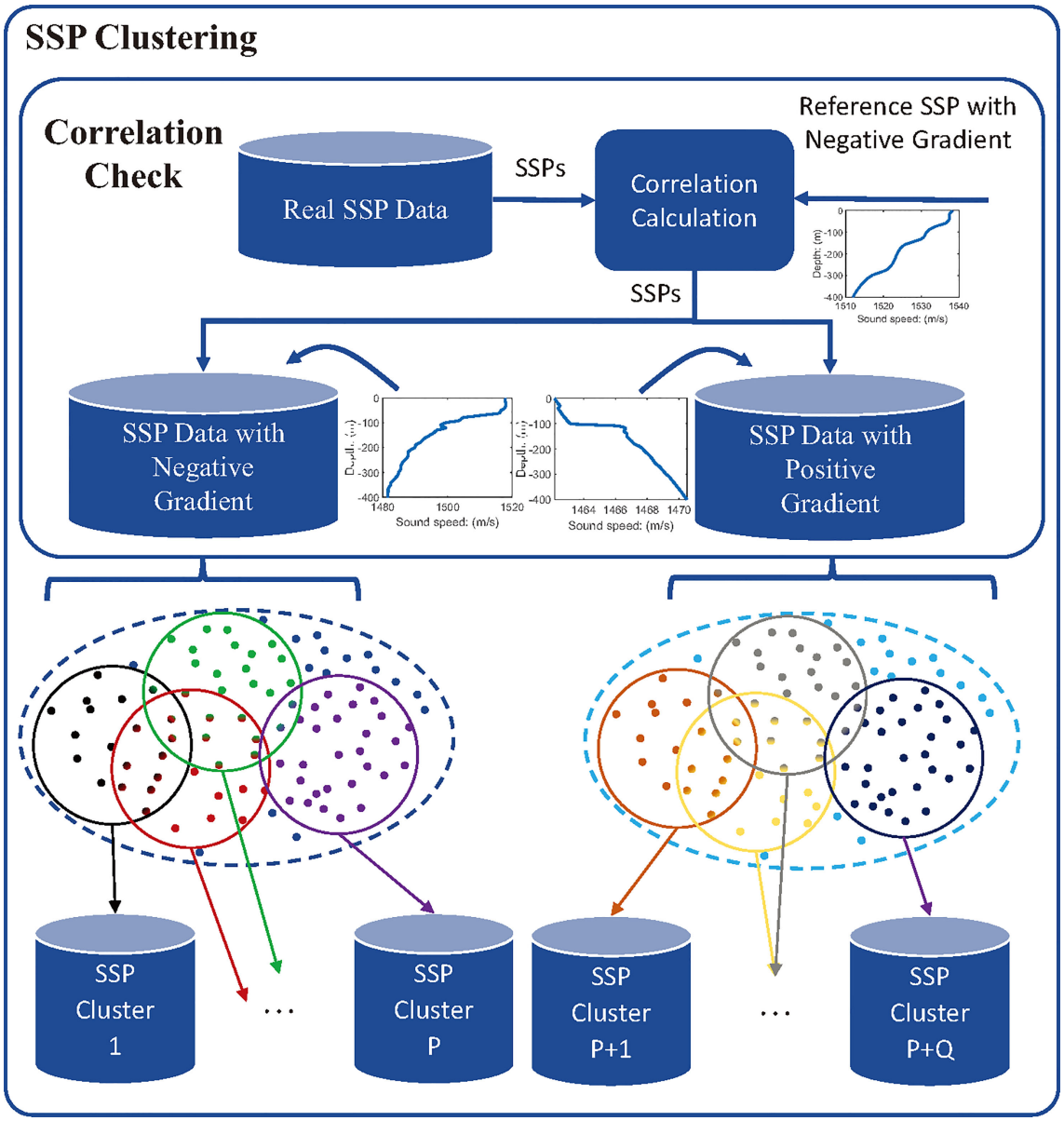


FIGURE 4
SSP Local density clustering method based on Pearson correlation test.

Therefore, a correlation check process is first established, in which a standard SSP with negative gradient is introduced as a reference to calculate the Pearson correlation coefficient between each historical SSP data and the reference one. Assume the reference SSP is $S_r = [(s_r^1, 1), (s_r^2, 2), \ldots, (s_r^d, d)]^T$, and the ith original SSP is $S_{o_i} = [(s_{o_i}^1, 1), (s_{o_i}^2, 2), \ldots, (s_{o_i}^d, d)]^T$, where d is the depth of corresponding sound speed in meters[2], the Pearson correlation coefficient ρ_{r,o_i} can be calculated by:

$$\rho_{r,o_i} = \frac{\sum_{d=1}^{D}\left(s_r^d - u_r\right)\left(s_{o_i}^d - u_{o_i}\right)}{\sqrt{\sum_{d=1}^{D}\left(s_r^d - u_r\right)^2}\sqrt{\sum_{d=1}^{D}\left(s_{o_i}^d - u_{o_i}\right)^2}}, \tag{4}$$

where $u_r = \frac{1}{D}\sum_{d=1}^{D} s_r^d$ is the average sound speed of SSP S_r, and $u_{o_i} = \frac{1}{D}\sum_{d=1}^{D} s_{o_i}^d$ represents the average sound speed of SSP S_{o_i}. With equation (4), all historical SSP data $\mathcal{S}_O = \{S_{o_1}, S_{o_2}, \ldots, S_{o_i}\}, i = 1, 2, \ldots, I$ will be divided into two group: the SSP group $\mathcal{S}_{O^-}$ with negative gradient or the SSP group $\mathcal{S}_{O^+}$ with positive gradient.

After the correlation check, the SSPs in each subset could be further clustered into different groups based on the Euclidean distance. If $S_{o_1^-} = [(s_{o_1^-}^1, 1), \ldots, (s_{o_1^-}^d, d)]^T$ and $S_{o_2^-} = [(s_{o_2^-}^1, 1), \ldots, (s_{o_2^-}^d, d)]^T$ are both SSPs in $\mathcal{S}_{O^-}$, the Euclidean distance $e_{o_1^-, o_2^-}$ is calculated as:

2 The sampling depth interval of original SSP data from world ocean database 2018 (WOD'18) is meter

$$e_{o_1^-,o_2^-} = \sqrt{\sum_{d=1}^{D}\left(s_{o_1^-}^d - s_{o_2^-}^d\right)^2}. \quad (5)$$

Similarly, for $S_{o_1^+} = [(s_{o_1^+}^1, 1), \ldots, (s_{o_1^+}^d, d)]^T$ and $S_{o_2^+} = [(s_{o_2^+}^1, 1), \ldots, (s_{o_2^+}^d, d)]^T$ in $\mathcal{S}_{O^+}$, the Euclidean distance $e_{o_1^+,o_2^+}$ can be calculated as:

$$e_{o_1^+,o_2^+} = \sqrt{\sum_{d=1}^{D}\left(s_{o_1^+}^d - s_{o_2^+}^d\right)^2}. \quad (6)$$

For classical K-means clustering algorithm or density-based spatial clustering (DBSCAN) algorithm, one sample is usually classified into one class. However, repetitive clustering of SSP is allowed in PC-SLDC algorithm, the details of which is given in Algorithm 1.

Input: historical SSPs:
$\mathcal{S}_O = \{S_{o_1}, S_{o_2}, \ldots, S_{o_i}\}$, including I SSPs;
reference SSP with Negative Gradient: S_r;
euclidean distance threshold: ψ_{dis};
neighbor number threshold: ψ_{num}.
Output: SSP cluster set:
$S_c = \{S_{C_1}, \ldots, S_{C_p}, \ldots, S_{C_{(p+1)}}, \ldots, S_{C_{(p+q)}}\}$,
including P clusters with negative gradient and
Q clusters with positive gradient
Step 1 Initialization:
SSP set with negative gradient $\mathcal{S}_{O^-} = \emptyset$;
SSP set with positive gradient $\mathcal{S}_{O^+} = \emptyset$;
euclidean distance matrix M_{ed};
candidate cluster center SSP set $\mathcal{S}_{Ct} = \emptyset$;
neighbor SSP set $\mathcal{S}_{Nbr} = \emptyset$;
SSP cluster set $\mathcal{S}_C = \emptyset$;
Step 2 Correlation check:
foreach *SSP sample* S_{oi} *in* S_O **do**
calculate the Pearson correlation coefficient $\rho_{r,oi}$ between S_r and S_{o_i} according to (6);
If $\rho_{r,oi} > 0$ **then**
Add S_{o_i} to $\mathcal{S}_O$;
else
add S_{o_i} to $\mathcal{S}_{O^-}$.
Step 3 Local density clustering of SSPs with negative gradient:
assign $\mathcal{S}_{Ct}$ / $\mathcal{S}_O$ and label the elements as
$\mathcal{S}_{Ct} = \{S_{ct_1}, S_{ct_2}, \ldots, S_{ct_a}\}$;
calculate the Euclidean distance among SSPs in S_{Ct} by (7) and store the results in M_{ed};
While $\mathcal{S}_{Ct} \neq \emptyset$ **do**
Randomly pick an SSP sample $S_{ct_a} \in \mathcal{S}_{Ct}$;
Reset $\mathcal{S}_{Nbr} = \emptyset$;
foreach *SSP* SSP $S_{o_{\bar{a}}^-} \in \mathcal{S}_{O^-}$ **do**
Check the Euclidean distance $e_{ct_a,o_{\bar{a}}^-}$
if $e_{ct_a,o_{\bar{a}}^-} < \Psi_{dis}$ **then**
add $S_{o_{\bar{a}}^-}$ to S_{Nbr}
if SSPs in $S_{Nbr} \geq \psi_{\text{num}}$ **then**
add a new cluster $\mathcal{S}_{C_p} = \mathcal{S}_{Nbr}$ to S_C;
remove SSPs from S_{ct} that are also contained in $\mathcal{S}_{C_p}$
else
remove S_{ct_a} from $\mathcal{S}_{Ct}$
Step 4 Local density clustering of SSPs with positive gradient:
repeat Step 3 by replacing $\mathcal{S}_O$ with $\mathcal{S}_{O^-}$, *a* with *b*, - with +, equation (7) with (8), and $\mathcal{S}_{C_p}$ with $\mathcal{S}_{C_{(p+q)}}$.

ALGORITHM 1
Pearson-correlation-based SSP local density clustering algorithm

At the beginning, all unclassified SSPs in $\mathcal{S}_{O^-}$ have the opportunity to become a new class center and they form a candidate cluster center set $\mathcal{S}_{Ct}$. The Euclidean distance between each other is calculated through (5) and stored in an Euclidean distance matrix M_{ed}. An SSP sample $S_{ct_a} \in \mathcal{S}_{Ctr}$ is randomly picked up, if the Euclidean distance between any SSP $S_{o^-} \in \mathcal{S}_{O^-}$ and the current candidate center S_{ct_a} is less than a threshold Ψ_{dis}, then the former will be a neighbor of the latter and added to a neighbor SSP set $\mathcal{S}_{Nbr}$. If the number of SSPs in $\mathcal{S}_{Nbr}$ exceeds a certain threshold Ψ_{num}, the current candidate point S_{ct_a} will be taken as the true center to establish a group $\mathcal{S}_{C_p}$, and all neighbors are added into $\mathcal{S}_{C_p}$. Otherwise, S_{ct_a} will be removed from $\mathcal{S}_{Ct}$ and a new candidate center SSP will be chosen to repeat theabove process. The whole process will be done again for SSPs in $\mathcal{S}_{O^+}$.

4.1.2 Spatio-temporal-information-based target task mapping

For a specified SSP inversion task, those historical sampled SSPs having the similar distribution with the target task is suitable for training the task inversion model. However, the sound speed distribution of the target task is not a prior information, thus the potential training SSPs can not be found according to the distribution features of SSPs. Since that the distributions of SSPs are similar when these SSPs are sampled with close spatio-temporal information, the search of suitable training data can be realized based on the similarity of spatio-temporal information.

For target task mapping, we propose an STI-KNN task mapping algorithm to find proper reference data for the task inversion model, and the prior SSP clusters $\mathcal{S}_C$ is obtained by Algorithm 1. When an inversion task is assigned, we define a spatio-temporal distance parameter ϕ to describe the similarity between the sampling regions of a reference SSP and the task, which can be expressed as:

$$\phi = \lambda \star \phi_\alpha + (1-\lambda) \star \phi_\beta, \quad (7)$$

where ϕ_α is the sampling time difference, ϕ_β is the sampling location difference, and $0 \leq \lambda \leq 1$ is a factor to balance ϕ_α and ϕ_β.

The ϕ_α is calculated by:

$$\phi_\alpha = \begin{cases} |\alpha_t - \alpha_o|, if |\alpha_t - \alpha_o| < 183 \\ 365 + \min(\alpha_t, \alpha_o) - \max(\alpha_t, \alpha_o), otherwise \end{cases} \quad (8)$$

where α_t and α_o are the time information of SSP inversion task and a random SSP in $\mathcal{S}_O$ (Algorithm 1), respectively. Due to the high similarity of SSPs sampled at the same period in different years

within a certain area, it is not necessary to distinguish the year differences, thus the sampling time information is defined in days. If an SSP is collected on February 1, the sampling time value equals to 32, because the 1st day on February is the 32th day of the year. However, it should be noted that the time difference will not exceed half a year (183 days), because the time code is cyclic. For instance, assume two SSPs are sampled on October 1 in the last year (the 244th day of a year) and January 1 in the current year (the 1st day of a year), the actual time difference is 365 + 1 − 244 = 122, but not 244 − 1 = 243. This is because the 1st day of the last year is equal to the 1st day of the current year, which could be virtually regarded as the 366th day of the last year (without lose of generality, the leap year is taken as an example).

The space information is defined by the latitude and longitude coordinate of an SSP. The ϕ_β is calculated by:

$$\phi_\beta = \sqrt{(\beta_t^x - \beta_o^x)^2 + (\beta_t^y - \beta_o^y)^2}, \tag{9}$$

where subscript t and o have the same meaning as (8), β^x and β^y represent the longitude and latitude coordinates of SSP sampling space after coding, respectively. As we focus on the distribution of sound speed in the Pacific Ocean of the Northern Hemisphere, the coded β^y equals to the SSP's latitude coordinate, while β^x is defined as:

$$\beta^x = \begin{cases} \left|\hat{\beta}^x\right| - 180, if & 0^\circ E < \hat{\beta}^x < 180^\circ E \\ 180 - \left|\hat{\beta}^x\right|, if & 0^\circ W < \hat{\beta}^x < 180^\circ W \end{cases} \tag{10}$$

where $\hat{\beta}^x$ is the original longitude coordinate of the SSP.

After comparing the spatio-temporal distance between all historical SSPs and the target SSP, the cluster which contains most of the κ nearest SSPs will be determined as the mapping result of the target task. The factor λ in (7) is determined through random verifications, which is conducted based on real sampled SSP data from WOD'18 in the Pacific Ocean with different kinds of distribution. Through these random verifications, the accuracy of mapping the target task to the exact cluster, that has similar SSP distribution with the task, will be statistically tested under different λ values, and the most appropriate λ will be determined according to the highest mapping accuracy.

The training data for an SSP inversion task can be artificially provided or automatically selected by machine learning algorithms. For automated SSP inversion system with much less human cost, the lambda will affect the probability of providing suitable training data for the task learner. Since the SSPs with different distribution compared with those of the task area will mislead the learning process of task learner, the inversion accuracy will decrease when the SSP cluster of the task is wrongly mapped. Therefore, the task mapping accuracy that corresponding to the factor λ indicates the confidence coefficient of an inverted SSP result. The STI-KNN algorithm is given in Algorithm 2.

Input: historical SSPs:
$\mathcal{S}_O = \{S_{o_1}, S_{o_2}, \ldots, S_{o_i}\}$;
SSP clusters:
$S_c = \{S_{C_1}, \ldots, S_{C_p}, \ldots, S_{C_{(p+1)}}, \ldots, S_{C_{(p+q)}}\}$;
spatio-temporal information of historical SSPs:
$\Phi = \{(\alpha_{o_1}, \beta_{o_1}^x, \beta_{o_1}^y), \ldots, (\alpha_{o_i}, \beta_{o_i}^x, \beta_{o_i}^y)\}$;
spatio-temporal information inversion task:
$\varphi = (\alpha_t, \beta_t^x, \beta_t^y)$;
number threshold of neighbors: κ.
Output: SSP cluster of the target task: S_{ct}.
Step 1: calculate the spatio-temporal distance between target task and historical SSPs by (9);
Step 2: sort the spatio-temporal distance;
Step 3: choose κ SSPs from S_O with the lowest spatio-temporal distance;
Step 4: select the cluster $\mathcal{S}_{C_p}$ or $\mathcal{S}_{C_{(p+q)}}$ containing the most of the κ SSPs as the mapping result of the target task.

ALGORITHM 2
STI-KNN task mapping algorithm

4.2 Task-driven meta-learning

To solve the over-fitting problem and increase the SSP inversion accuracy with few-shot reference samples, we propose a TDML SSP inversion model as shown in Figure 5 that includes a meta-training phase and an SSP inversion phase. There is a global learner, several base learners and a task learner in the proposed model. Through K base learners each trained with V-shot SSPs from different clusters, which is called K-way V-shot learning, a good set of initialization parameters for the global learner is found, so that the task learner initialized by the global learner could converge quickly with a few training times on the task SSP training set.

According to the SSP clustering result by the proposed PC-SLDC algorithm, the SSP distribution of the target task is either positive or negative, and the base learner trained by SSPs with the opposite gradient will contribute negatively to the global learner, which will slow down the convergence progress of the task model, even decrease the inversion accuracy. To diminish the negative transfer, the SSP clusters that having the same gradient direction with that of the task SSP set are chosen as the candidate training sets for base learners. Moreover, if the distribution of SSPs learned by the base learner k is more similar to that of the task training SSPs, the base learner k will have more influence on the parameter updating of the global learner, which is achieved by adjusting the gradient learning rate. Thus, the negative transfer could be further weakened, and the task learner could converge faster so as to avoid over-fitting on few-shot reference samples.

Concretely, we propose a TDML SSP inversion algorithm to illustrate the model training and application process. The neuron connection parameter of global learner is randomly initialized as

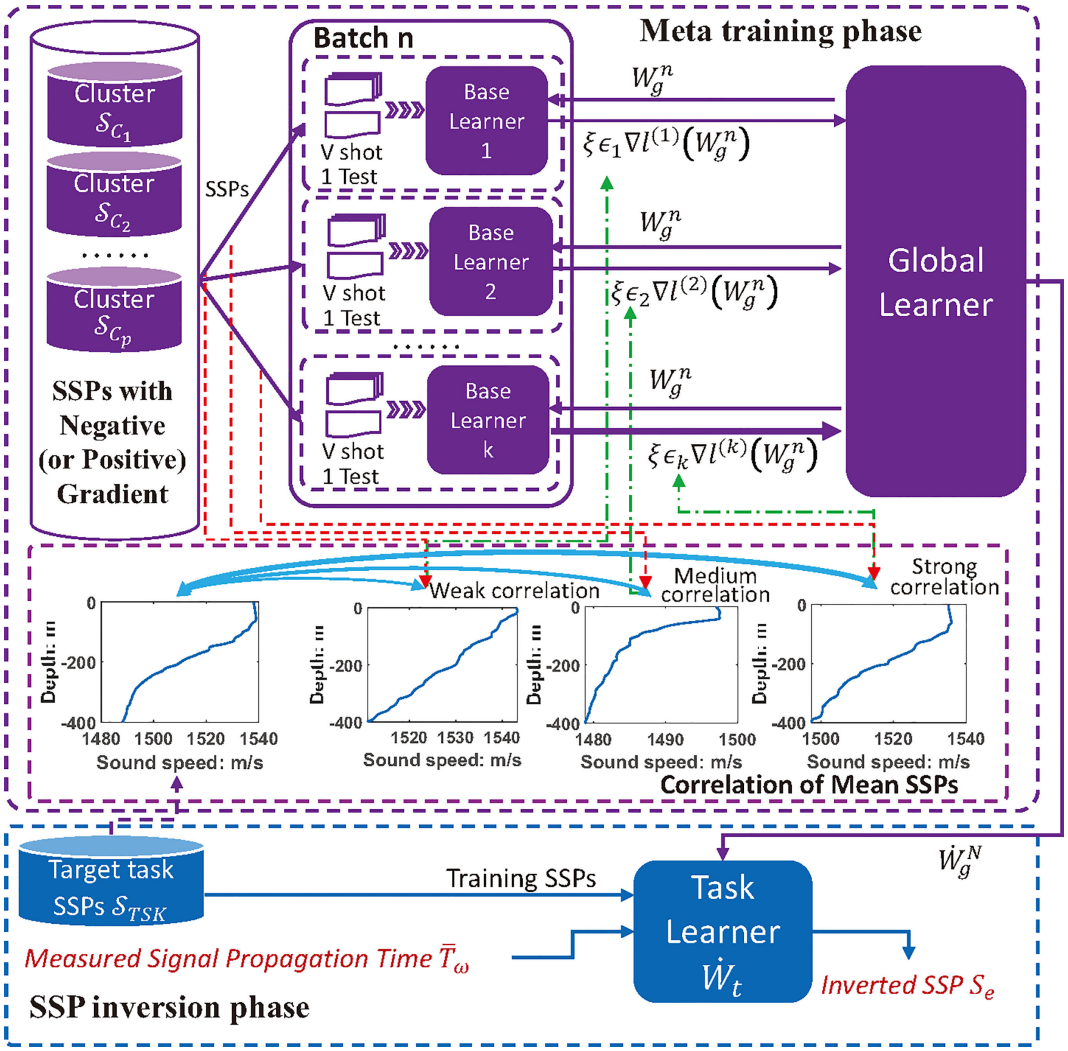


FIGURE 5
Task-driven meta-learning model for SSP inversion.

W_g^1 while updated iteratively. At the beginning of the nth batch[3], all the K base learners are initialized by the global learner W_g^n, meanwhile, K SSP clusters are randomly chosen from the P available training SSP clusters ($K \le P$) for training the K base learners respectively, each of which consists of 1 testing and V training samples. For base learner k, the V SSPs are used forone step learning with the lose function defined as:

$$l^{(k)}(W_k^n) = \sum_{v=1}^{V}\left(\frac{1}{2}\sum_{d=1}^{D}\left(s_v^d - \tilde{s}_v^d\right)^2 + \| W_k^n \|_1\right), W_k^n = W_g^n, \quad (11)$$

where s_v^d is the sound speed of the vth training SSP at depth d, $\tilde{s}_v^d$ is the corresponding inverted sound speed, and $W_k^n 1$ is the regularization item of the base learner k. Next, the local parameters are updated with back propagation (BP) algorithm by Rumelhart et al. (1986):

$$\dot{W}_k^n = W_k^n - \eta \nabla_{W_k^n} l^{(k)}(W_k^n), \quad (12)$$

where η is the learning rate of base learners. Then, the base learner is test on the left 1 SSP data S_{tst} with lose function:

$$l^{(k)}(\dot{W}_k^n) = \frac{1}{2}\sum_{d=1}^{D}\left(s_{tst}^d - \tilde{s}_{tst}^d\right)^2, \quad (13)$$

where s_{tst}^d and $\tilde{s}_{tst}^d$are the sound speed of the testing and inverted SSP, respectively. Finally, the parameters of the global learner are updated by optimizing the performance L with respect to W_g^n ($W_g^n = W_k^n$) across all base learners. The global optimization problem is expressed as follows:

$$\min_{W_g^n} L = \min_{W_g^n} \sum_{k=1}^{K} l^{(k)}(\dot{W}_k^n). \quad (14)$$

Note that the meta-optimization is performed over the initial parameter W_g^n during current iteration, whereas the objective is computed using the updated parameters $\dot{W}_k^n$. In this way, the

3 During each iteration, the base learner is trained and updated by one batch, so the total number of batches is equal to the number of iterations

sensitivity of the model could be enhanced so that one or a small number of gradient updating steps on a new task will produce maximally effective behavior on that task according to Finn et al. (2017).

To further improve the quality of initialization parameters learned by the global learner, a correlation coefficient $\epsilon_k (k = 1, 2, \ldots, K)$ is introduced into each base learner to adjust the updating speed of model parameters, which is concretely the Pearson correlation coefficient between the mean SSP of the kth meta training cluster and the mean SSP of the inversion task training SSPs. With the K-way V-shot training, the meta-optimization is actually performed through stochastic gradient descent, such that the global learner is updated by:

$$\dot{W}_g^n = W_g^n - \xi \nabla_{W_g^n} \sum_{k=1}^{K} \epsilon_k l^{(k)}(\dot{W}_k^n), \tag{15}$$

where $\dot{W}_g^n$ represents the global leaner after parameter updating, and ξ is the global learning rate. If the meta training is not over, the parameters of global learner in the $(n+1)$ th batch will be initialized as $W_g^{n+1} = \dot{W}_g^n$.

After meta-training, the parameters of global learner $\dot{W}_g^N$ is transfer as the initialization for the task learner, so that $W_t^1 = \dot{W}_g^N$. Then the task learner is trained on a few training SSPs by one or a small number of steps, and the converged model is parameterized as $\dot{W}_t$. When feeding measured sound field information such as signal propagation time into model W_t, the inverted SSP $\tilde{S}_e$ can be estimated via once forward propagation, thereby improving the inversion efficiency. The detailed TDML algorithm for SSP inversion is given in Algorithm 3.

Input: target task SSPs: S_{TSK};
SSP clusters:
$S_c = \{S_{C_1}, \ldots, S_{C_p}, \ldots, S_{C_{(p+1)}}, \ldots, S_{C_{(p+q)}}\}$;
task, global, base learners initialized by: W_g^1;
meta-training iterations: N;
target task training iterations: $\hat{N}$;
Output: SSP inversion result: S_e.
Step 1: preprocessing: get rid of SSP clusters in S_c with negative or positive gradient which is different from SSPs in S_{TSK};
Step 2: meta-training:
foreach iterations $n \leq N$ **do**
SSPs preparation for K-way V-shot learning;
foreach base learner k **do**train with V-shot SSPs and update the parameter by (13), (14);
test the base learner according to (15).
compute the global objective function by (16);
update the global learner via (17);
Step 3: task learner training:
assignment $W_t^1 \leftarrow \dot{W}_g^N$;
train task learner for $\hat{N}$ times: $W_t^1 \rightarrow \dot{W}_t$;
Step 4: task SSP inversion:
measure the signal propagation time $\bar{T}_\omega$;
invert SSP S_e by feeding $\bar{T}_\omega$.

ALGORITHM 3
Task-driven meta-learning algorithm for SSP inversion

To reduce the impact of the time measurement error on the inversion model, which is caused by inaccurate positioning of the communication system, the joint AEFMNN and ray tracing model proposed in our previous work Huang et al. (2021) is introduced as the basic learning model for the base and task learner, and the anti-noise performance of TDML is inherited. In Huang et al. (2021), the robust feature extraction performance of the autoencoder has been evaluated by comparing the variation trend of correlation coefficients on the input signals and implicit features under different levels of time measurement error, in which the positioning error of AUVs is set to be zero for simulating a single error source. The signal propagation time correlation coefficients are calculated by correlating the error-influenced signal propagation time with the ideal one, and the correlation coefficient of implicit features is obtained via correlations between the implicit features extracted when the input signal propagation time is influenced by the measurement errors or without errors. Detailed anti-noise performance of AEFMNN can be referred to Figure 12 in Huang et al. (2021).

The learning model of base and task learner is given in Figure 6. There are total 7 layers in AEFMNN model that have been described in detail by Huang et al. (2021): noisy time input layer $T_{v,\omega}$, encoding hidden layer F_{ec}, decoding hidden layer F_{dc}, decoding time output layer $\tilde{T}_v$, translating hidden layer F_{tr}, translating output SSP layer $\tilde{S}_v$, and the hidden feature layer F_{ed} shared by the encoder, decoder and translation neural network.

In Figure 6, the signal propagation time with measurement errors $T_{v,\omega}$ is simulated to reflect the real situation, while the one without errors $\tilde{T}_v$ is computed to be the labeled time information for updating the parameters of the auto-encoder. The auto-encoder and the translation neural network are updated in turn during once training. Through narrowing the gap between the estimated signal propagation time $\tilde{T}_v$ and the simulated time T_v, the auto-encoder is first trained to extract the implicit features that reduces the impact of measurement errors of the input data. Then by narrowing the gap between the inverted SSP $\tilde{S}_v$ and the labeled SSP $\hat{S}_v$, the translation neural network is trained to establish the mapping relationship from the implicit features to the sound speed distribution.

Taking the nth iteration (V-shot) for base learner k as an example, the parameters of the auto-encoder is updated by BP algorithm with the time lose function $l_t^{(k)}(W_k^n)$ expressed as:

$$l_t^{(k)}(W_k^n) = \sum_{v=1}^{V} \left(\frac{1}{2} \sum_{m=1}^{M} (\tilde{t}_v^m - t_v^m)^2 + \| W_{k,ec,dc}^n \|_1 \right), \tag{16}$$

where $\tilde{t}_v^m$ is the estimated signal propagation time of the mth receiver, t_v^m is the corresponding theoretical time information without noise, and $W_{k,ec,dc}^n$ is the regularization item related to the parameters of the auto-encoder. Then, the translation neural network transform the hidden features to sound speed distribution with lose function (11) modified as:

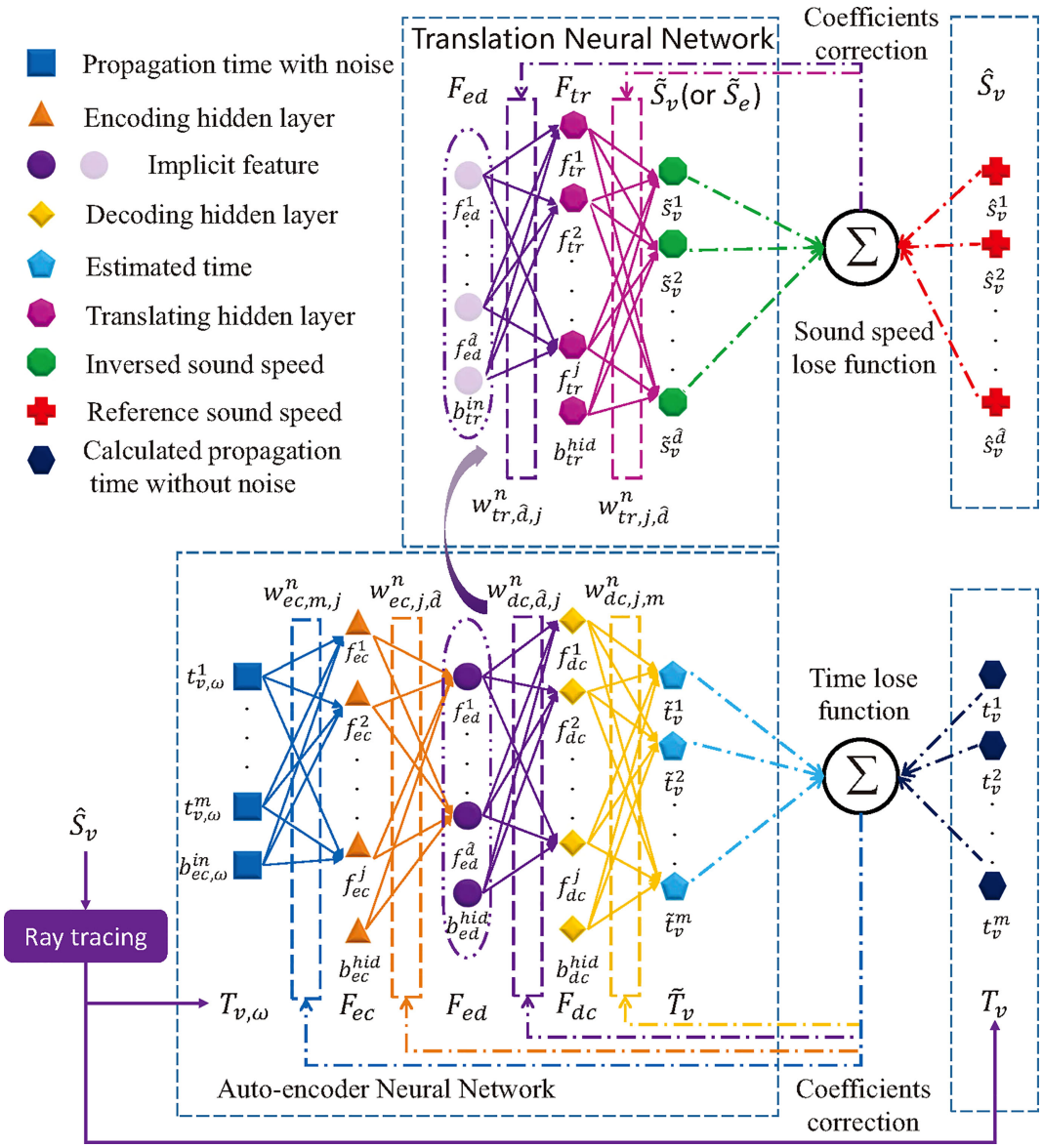


FIGURE 6
Joint Ray Tracing and AEFMNN SSP inversion model by Huang et al., 2021.

$$l^{(k)}(W_k^n) = \sum_{v=1}^{V}\left(\frac{1}{2}\sum_{d=1}^{D}\left(\tilde{s}_v^{\hat{d}} - \hat{s}_v^{\hat{d}}\right)^2 + \| W_{k,tr}^n \|_1\right), \tag{17}$$

where $\tilde{s}_v^{\hat{d}}$ is the inverted sound speed at depth $\hat{d}$, $\hat{s}_v^{\hat{d}}$ is the corresponding labeled sound speed, and $W_{k,tr}^n$ is the regularization item related to the parameters of the translation neural network. The forward propagation process of AEFMNN is done by following equations:

$$f_{ec}^j = \Gamma\left(\sum_{m=1}^{M}\left(t_{v,\omega}^m \cdot w_{ec,m,j}^n\right) + b_{ec,\omega}^{in} \cdot w_{ec,b,j}^n\right), \tag{18}$$

$$f_{ed}^{\hat{d}} = \Gamma\left(\sum_{j=1}^{J}\left(f_{ec}^j \cdot w_{ec,j,\hat{d}}^n\right) + b_{ec}^{hid} \cdot w_{ec,b,\hat{d}}^n\right), \tag{19}$$

$$f_{dc}^j = \Gamma\left(\sum_{\hat{d}=1}^{\hat{D}}\left(f_{ed}^{\hat{d}} \cdot w_{dc,\hat{d},j}^n\right) + b_{dc}^{in} \cdot w_{dc,b,j}^n\right), b_{dc}^{in} = b_{ed}^{hid}, \tag{20}$$

$$\tilde{t}_v^m = \Gamma\left(\sum_{j=1}^{J}\left(f_{dc}^j \cdot w_{dc,j,m}^n\right) + b_{dc}^{hid} \cdot w_{dc,b,m}^n\right), \tag{21}$$

$$f_{tr}^j = \Gamma\left(\sum_{\hat{d}=1}^{\hat{D}}\left(f_{ed}^{\hat{d}} \cdot w_{tr,\hat{d},j}^n\right) + b_{tr}^{in} \cdot w_{tr,b,j}^n\right), b_{tr}^{in} = b_{ed}^{hid}, \tag{22}$$

$$\tilde{s}_v^{\hat{d}} = \Gamma\left(\sum_{j=1}^{J}\left(f_{tr}^j \cdot w_{tr,j,\hat{d}}^n\right) + b_t^{hid} \cdot w_{tr,b,\hat{d}}^n\right). \tag{23}$$

where the special subscript b of the weight parameter w indicates that the weight connects the bias neuron of current layer and the neurons in next layer. Among (18) to (23), the leaky rectified linear unit (LReLU) by Maas et al. (2013) is introduced as the activation function, which is expressed as:

$$\Gamma(\tau) = \begin{cases} \tau & \tau > 0 \\ \zeta\tau & \tau \le 0 \end{cases} \tag{24}$$

where ζ is a fixed constant between -1 and 0 (0.25 in this paper).

The outputs of the translation neural network are the sound speeds at different depth, so the number of output neurons depends on the sampling depth of the SSP. If there are too many sampling points in an SSP, the required parameters of the neural network will increase significantly, thereby leading to the over-fitting problem when trained on few-shot dataset. To reduce the model complexity, an stratified-line SSP simplification algorithm proposed in our previous work Huang et al. (2019) is introduced, by which an original SSP S_v could be accurately approximated to be $\hat{S}_v$ via a few feature points.

5 Simulation and discussion

In this section, the performance of the proposed PC-SLDC algorithm for SSP clustering, the accuracy of task mapping with STI-KNN algorithm, the SSP inversion efficiency and accuracy under TDML framework are verified by simulations on historical SSP data in the shallow Pacific ocean with water depth of 400 m. However, the application is not limited to the experimental area, but also applicable to shallow or deep ocean where the sound speed distribution is consistent in a certain spatio-temporal range and the sound speed at each depth layer approximately obeys Gaussian distribution with the root-mean-square error (RMSE) on the order of a few meters per second. The experiments are done via Matlab "R2019a", and all SSPs are real sampled in the Pacific Ocean that come from the WOD'18 Boyer et al. (2021), the sonar data used for SSP inversion is simulated through ray theory.

5.1 Accuracy of target task mapping

To guarantee the convergence performance of the learning model, the i.i.d. condition of training and testing data needs to be satisfied. Therefore, similarly clustering the empirical SSPs and finding which cluster the target task belongs to become extremely important. To evaluate the performance of proposed PC-SLDC and STI-KNN algorithms, we first divide the SSPs into clusters base on PC-SLDC, then test the target task mapping accuracy by STI-KNN, finally compare the SSP similarity under different clustering criteria.

In Figure 7, the mapping accuracy of STI-KNN algorithm for target task is tested on 391 historical SSPs sampled in the Northern Pacific Ocean with each checking 7 neighbor SSP samples. When the Euclidean density distance threshold is set to be $\Psi_{dis} \le 10$ and the element number of clusters is set to be $\Psi_{num} \ge 15$ in PC-SLDC, the accuracy of STI-KNN can be up to 96% with $\lambda = 0.02$. When the Euclidean density distance threshold is set to be $\Psi_{dis} \le 8$ andthe element number of clusters is set to be $\Psi_{num} \ge 12$ in PC-SLDC, the accuracy of STI-KNN can be up to 97.85% with $0.01 \le \lambda \le 0.036$

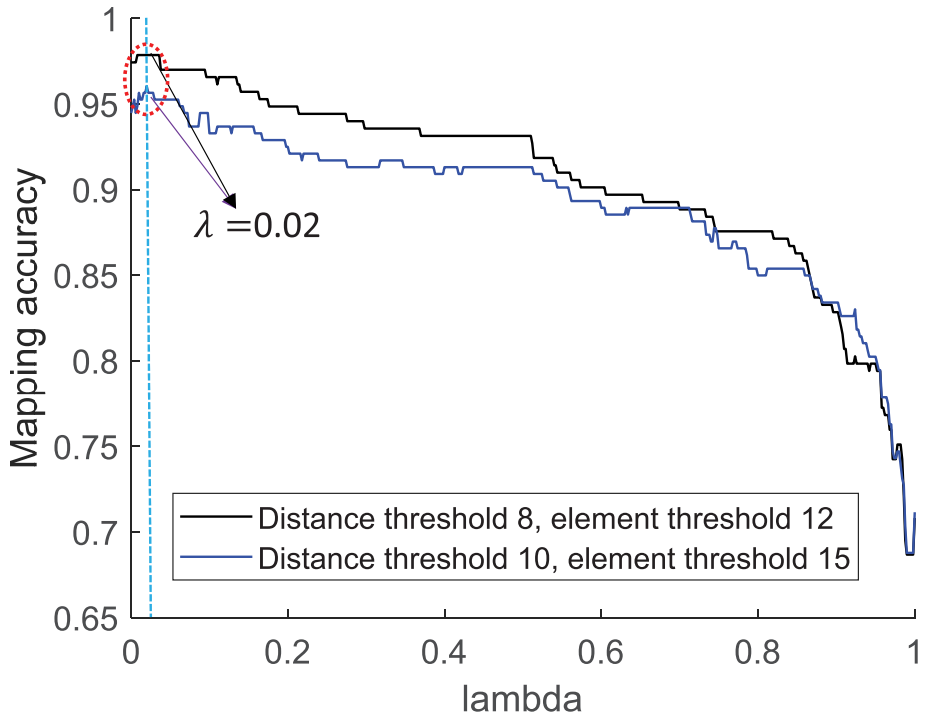


FIGURE 7
Mapping accuracy of SSP inversion task by STI-KNN algorithm.

. The λ is set to be 0.02 in our following simulations, and SSPs are clustered with $si_{dis} \leq 10, \Psi_{num} \geq 15$.

After target task mapping, we evaluate the clustering performance by testing the error distances of 20 samples to the mean SSP of the cluster that each sample maps to via STI-KNN as shown in Figure 8, the cluster of which is obtained by PC-SLDC and compared with clustering merely by SSP sampling month or location. The SSPs of items 1, 2 and 3 are the same group with negative gradients, while the SSPs of items 4, 5 and 6 are the same group with positive gradients. The location threshold is 5 longitude and latitude, and the month threshold is 1 month. From the result, the SSPs clustered through PC-SLDC are more similar to their cluster elements than SSPs clustered merely by month or location information. In particular, the RMSE of test SSPs to the mean SSP of the cluster obtained by month is much worse than the other two, this is because the empirical SSPs within each month are sampled dispersedly around the Northern Pacific Ocean, the distributions of which are obviously different.

The average SSP of each cluster can be used for roughly estimating the sound speed distribution of a certain area, however, the variation of sound speed can not be reflected in different regions or sampling date, and the estimation error of sound speed will increase with the area scale or time interval expanding. Therefore, it is necessary to further improve the accuracy of sound speed inversion by learning to establish the mapping relationship from signal propagation time to sound speed distribution.

5.2 SSP inversion under TDML framework

The TDML framework proposed in this paper aim to improve the SSP inversion accuracy while reduce the time as much as possible. In this section, we test the accuracy and time efficiency of the proposed TDML-based SSP inversion method compared with some classical SSP inversion methods as base lines.

5.2.1 Base lines and parameter settings

5.2.1.1 MFP-EOF-PSO

There are four steps included in this classical SSP inversion method: principal component extraction, candidate SSPs generation, simulated sound field calculation and sound field matching; the candidate SSP corresponding to the optimal matching sound field will be recorded as the final inversion result. Heuristic algorithms are widely used in searching for the matched item, and PSO is adopted as an example in this paper.

5.2.1.2 CS

The CS-based SSP inversion is a new method that is combined with EOF. In this method, the eigenvectors of EOF are utilized to form the compressed sensing dictionary.

5.2.1.3 AEFMNN-based single learning model

The single AEFMNN SSP inversion model has the same model structure with any base learner of TDML. Such a model is set up for comparing to evaluate the anti-over-fitting performance of TDML.

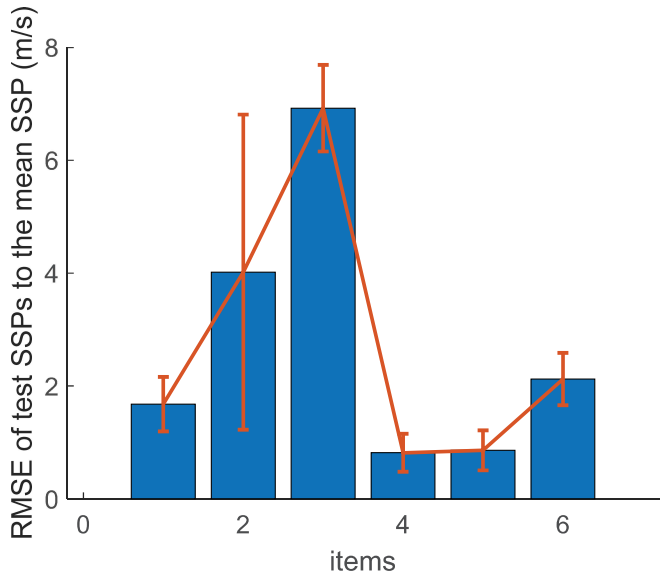


FIGURE 8
Error distances of SSPs to the average SSP of the task cluster.

5.2.1.4 TL

TL is a classical method that can be used for few-shot learning. Two AEFMNN models having the same structure with base learners of TDML are introduced. One model is trained on SSPs that are not belong to the task cluster, while the other is trained by the task cluster; the trained parameters of the former model are set to be the initialization for the later model.

5.2.1.5 ML

The difference between ML and TDML for SSP inversion is that all SSP clusters, excluding the task cluster, can be used for meta-training in ML, while only SSP clusters having the same positive or negative gradients with the task cluster can be used for meta-training in TDML. By this means, we verify the performance of TDML against the negative migration.

The parameter settings of TDML are shown in Table 1. As the position error of the bottom AUV in Figure 9 is harmful for sound field measurement, the location should be modified before the measuring process. According to our previous work on ray-tracing-based positioning correction Huang et al. (2019), the bottom AUV can be relocated with the help of those surface AUVs according to the average sound speed distribution of the task area. Under 10000 times simulation tests, the location error can be reduced through ray tracing technique based on average empirical SSP distritbution Huang et al. (2019), and follows the Gaussian distribution with average error $0m$ and standard deviation less than $0.1m$ under the time measurement error level $\sigma_c = 3ms$ (three surface AUVs form a equilateral triangle with 100 m between each other, and the position errors of surface AUVs are not considered). In reality, the location error of bottom AUV may be larger than simulated due to the topology chaning of surface AUVs and the movement of underwater flow. Since the AEFMNN is the basic inversion model introduced in this paper, the SSP inversion accuracy of all these methods will be influenced when the location error increases, however, the anti-overfitting performance and the convergence performance will still be different with these methods. Some specific parameter settings of base lines are given in Table 2.

5.2.2 Accuracy comparison

To verify the effectiveness of task mapping based on spatio-temporal information, the inversion average accuracy of TDML with 100 testing times is compared with clustering criterion by location or month in Table 3. Results show that the TDML trained with clustering by month or location is hard to converge because the training samples in every cluster may be far different from each other, thus the inversion errors are much higher than the clustering by spatio-temporal information.

To evaluate the accuracy performance of TDML, the RMSE results of SSP inversion on two different clusters with negative or positive gradients are tested as examples in Table 4 compared with other inversion methods. The inversion results in Table 4 are average results with 100 testing times. The results indicate that through leveraging sound field information such as signal propagation time, the SSP inversion accuracy behaves better than rough estimating by the average SSP of the cluster. Actually, the signal propagation time is a sensitive function of sound speed changes, while with measured signal propagation time, these changes can be seized to some extent by those SSP inversion methods.

For evaluating the anti-over-fitting ability, both the SSP inversion accuracy during task training and testing processes are tested for deep-learning-based methods. Among these methods, the TDML performs best for testing SSP samples. The accuracy of SSP

TABLE 1 Parameter settings of TDML.

TDML	
Training SSP clusters S_c	18-/4+
Base learners K	3
SSPs for base learner training S_v	9
SSPs for base learner testing S_{tst}	1
Meta-training episodes (batches) *	40
Task training episodes	40
Task training SSPs per episode	5
Maximum SSP depth	400 m
Points of simplified SSPs	8
Ideal horizontal distance (Figure 1)	80,120,…,440
	m (10 items)
Bottom AUV's location error μ_{bp}	0 m
Bottom AUV's location error σ_{bp}	0.1 m
Surface AUVs' location error μ_{sp}	0 m
Surface AUVs' location error σ_{sp}	0.1 m
Time measurement error μ_c	0 s
Time measurement error σ_c	0.003 s
Learning rate for base/meta learner	η_{ed} =0.01
	ξ_{ed} =0.01
	η_{tr} =0.00003
	ξ_{tr} =0.00003
Learning rate for task learner	$\hat{\eta}_{ed}$ = 0.01
	$\hat{\eta}_{tr}$ = 0.0001
Input layer neurons	10
Hidden layer neurons	200
Hidden feature neurons	8
Output layer neurons of auto-encoder	10
Output layer neurons of translator	8
Training SSPs in the task cluster	60%
Validating SSPs in the task cluster	20%
Testing SSPs in the task cluster	20%

* One episode corresponds to a round of parameter updating, using 3 SSP clusters that is equal to the number of base learners.

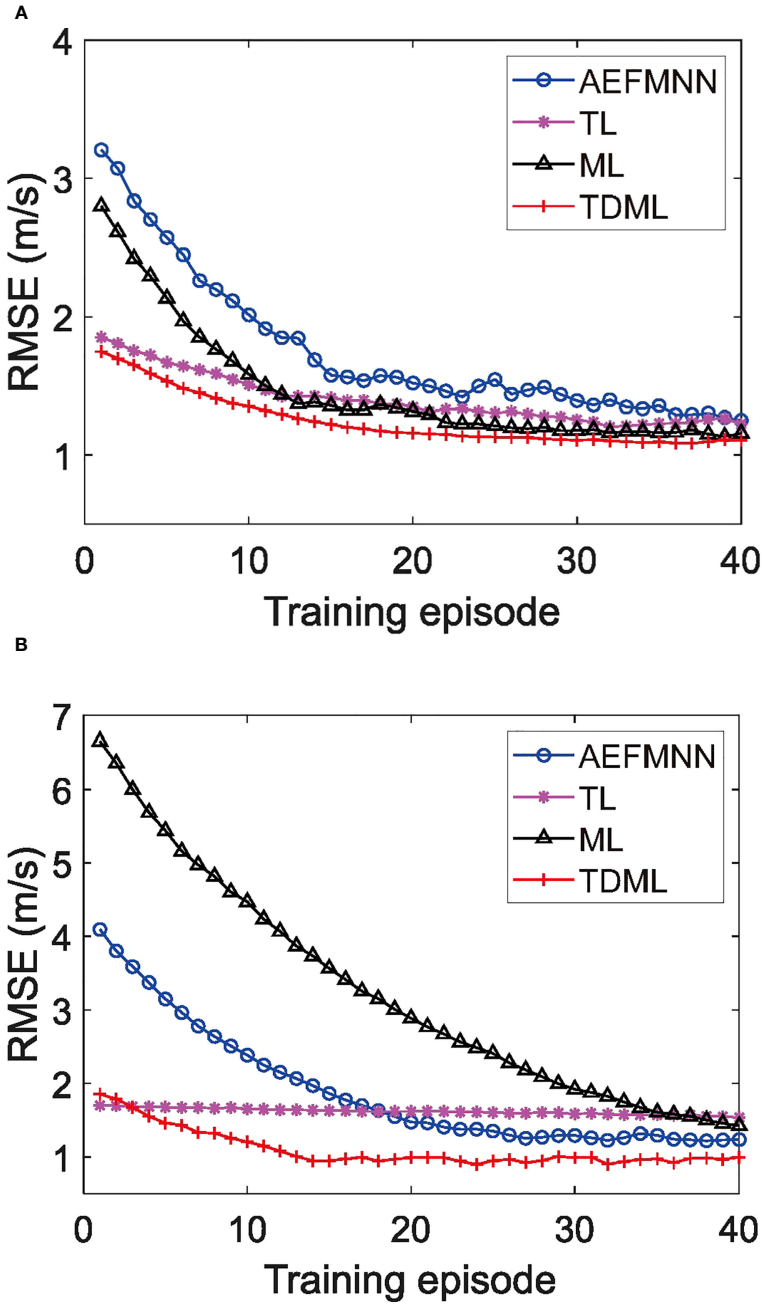


FIGURE 9
Convergence comparison of different deep learning methods for SSP inversion. **(A)** Cluster 1with negative gradients. **(B)** Cluster 2 with positive gradients.

inverted by CS is a little worse than MFP (combined with EOF and PSO) due to the first-order Taylor linear approximation at the dictionary establishing process. Because of the scarcity of training samples after clustering by PC-SLDC, the SSP inversion via AEFMNN is prone to be over-fitting, which is why the training accuracy can be extremely high but the testing accuracy will be greatly reduced, and reflected in large test-validation values. For TL, ML and TDML, the anti-over-fitting capability is improved. However, it should be noted that the inversion accuracy by TL is not as good as ML or TDML, and this is mainly because the initialization parameters of the task model are pre-trained in different ways. For TL, the model is pre-trained by another SSP cluster, which makes it retain much characteristics of pre-training cluster when transferring model parameters, thereby reducing the ability to learn new SSP distribution. On the contrary, the ML or TDML is pre-trained by meta models to learn more public features among SSP clusters, and the second-order gradient descent by (15) makes the model more sensitive to the changes of signal propagation time. Therefore, the ML-based model is not likely to be over-fitting on pre-training SSP clusters.

For cluster 1, the accuracy improvement of TDML is not obvious compared with that of ML. In fact, this phenomenon is

TABLE 2 Parameter settings of base lines.

ML	
Training SSP clusters	22
TL	
Pre-training episodes	40
Local training episodes	40
Learning rate	0.01 for auto-encoder
	0.00003 for translator
AEFMNN	
Training episodes	40
Learning rate	0.01 for auto-encoder
	0.0001 for translator
CS	
EOF feature vectors	6
CS orders	4
MFP-EOF-PSO	
EOF feature vectors	5
PSO iterations	18
PSO particles	20

related to the training SSP clusters. Among the 22 training SSP clusters for ML, 18 clusters are distributed in negative gradient, which is the same with the target task cluster. To verify the resistance ability of TDML to negative migration, SSPs with positive gradient are chosen to be the target task, and the comparison of inversion accuracy with different methods is given in cluster 2. For ML, most of pre-training SSP clusters are distributed in negative gradient, so it is difficult for the ML model to learn the common features of SSPs with positive gradient, resulting in bad learning ability on the new task. On the contrary, the pre-training clusters for TDML are all distributed in positive gradient, the accuracy performance can be guaranteed.

To give a more intuitive understanding of the negative migration in ML, the convergence of inversion tasks belonging to cluster 1 and 2 are displayed in Figures 9A, B, respectively. It can be noticed that with TDML, the model can converge after only 20 times of training, which is much faster than other methods. In few-shot learning, reducing the repeated training of samples is helpful to deal with the over-fitting problem. For task cluster 1, the initial parameters of TDML for task learner are closer to the optimal solution than ML; while for task cluster 2, the negative migration of ML is so obvious that the initial parameter is far from the optimal solution, and the convergence rate is also significantly reduced. However, for TDML in Figures 9A, B, there are decreasing processes and exist turning points that the RMSE error (m/s) becomes stable after a few of training episodes. Especially, the gap between the beginning and the convergence stage of TDML is smaller in Figure 9B, which indicates that the TDML does work and forms a good set of initial parameters of the task model.

For intuitively expressing the inversion results, the SSP inversion example through different methods is given in Figure 10. The result of TDML has better fitting with the original SSP curve.

5.2.3 Time efficiency comparison

The time efficiency of inversion method is very important for emergency tasks such as underwater rescue. As the training of learning models could be finished offline before task assignment, more attention should be paid to the time overhead on the inversion stage, which is compared in Figure 11. The match sound field information needs to be searched by heuristic algorithms in MFP, which is very time-consuming. For CS-based method, several iterations are needed to gradually reduce the residual. However, for learning-based methods, only once forward propagation is enough to obtain the inverted SSP with a well trained model, so the time efficiency is enormously improved.

6 Conclusion

To satisfy the accurate and time-efficient requirements of underwater localization applications such as emergency rescue, we propose a TDML framework for fast and accurately estimating the regional SSP that is beneficial for positioning correction. The TDML can be competent for most ocean SSP inversion tasks, especially in few-shot learning scenarios. By simultaneously learning different kinds of SSPs with several base learners, the common features of SSPs can be captured and transferred to the task learner, and the sensitivity of the task learner to the unique characteristics of task SSPs can also be maintained. Thus, the model can converge quickly in the face of new SSP inversion tasks, so as to reduce the over-fitting effect in few-shot learning.

TABLE 3 RMSE OF SSP inversion by TDML based on different clustering criterion.

	Result (m/s)			
Cluster	1		2	
Criterion	Validation	Test	Validation	Test
Location	13.945	**14.894**	11.723	**22.603**
Month	14.605	**13.621**	14.617	**13.633**
STI-KNN	1.178	**1.235**	0.998	**1.036**

The physical meaning of bold characters mainly reflects the generalization ability of the model.

10.3389/fmars.2023.1146333

TABLE 4 RMSE of SSP inversion by different methods.

Cluster	Result (m/s)							
1	Cluster mean	1.335						
1	EOF	1.320						
1	CS	1.330						
		V[1]	T[2]	Gap[3]	Space[4]	Ratio	Space	Ratio
1	AEFMNN	1.112	**1.402**	**0.290**	< 0.12	**0%**	> 0.12	**100%**
1	TL	1.130	**1.317**	**0.188**	< 0.12	**6%**	> 0.12	**94%**
1	ML	1.183	**1.266**	**0.083**	< 0.12	**72%**	> 0.12	**28%**
1	TDML	1.178	**1.235**	**0.058**	< 0.12	**97%**	> 0.12	**3%**
Cluster	Result (m/s)							
2	Cluster mean	1.241						
2	EOF	1.211						
2	CS	1.217						
		V[1]	T[2]	Gap[3]	Space[4]	Ratio	Space	Ratio
2	AEFMNN	1.028	**1.448**	**0.420**	< 0.09	**4%**	> 0.09	**96%**
2	TL	1.219	**1.263**	**0.044**	< 0.09	**73%**	> 0.09	**27%**
2	ML	1.304	**1.284**	**-0.021**	< 0.09	**90%**	> 0.09	**10%**
2	TDML	0.998	**1.036**	**0.038**	< 0.09	**94%**	> 0.09	**6%**

1 Validation mean error (m/s). 2 Test mean error (m/s). 3 Gap mean = Test mean error - Validation mean error (m/s). 4 Gap space (m/s).
The physical meaning of bold characters mainly reflects the generalization ability of the model.

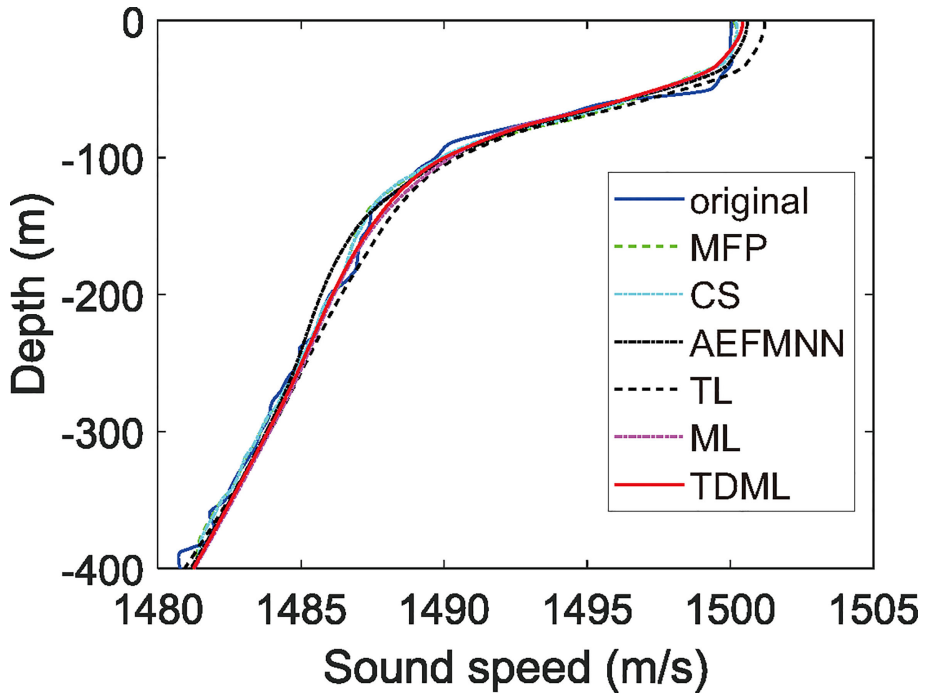


FIGURE 10
Time overhead of SSP inversion.

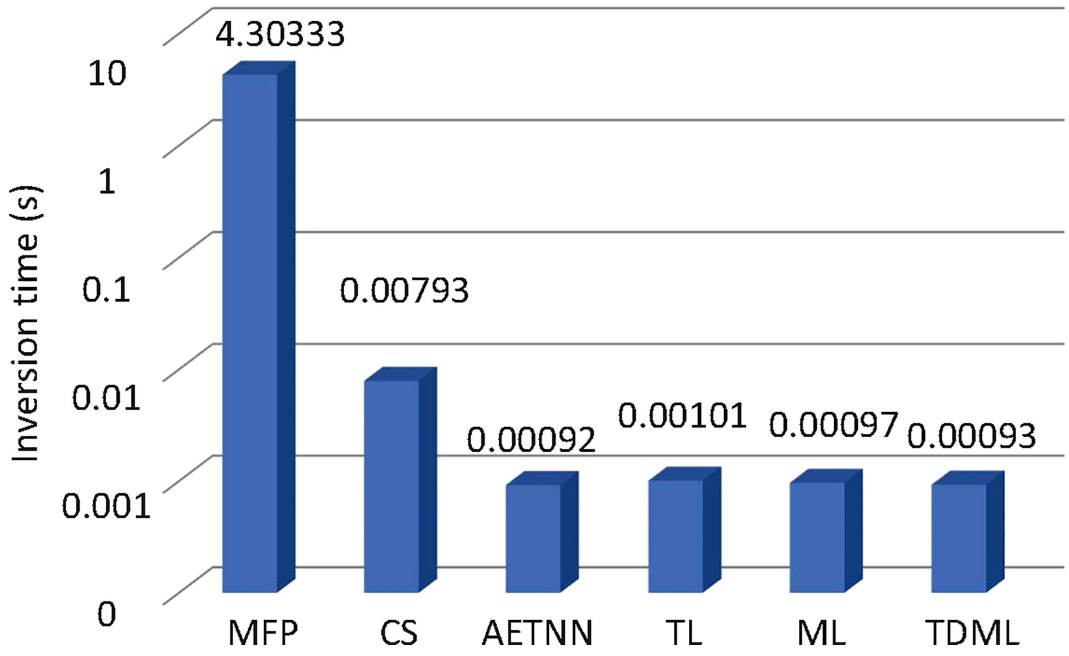


FIGURE 11
An example of SSP inversion result.

To guarantee the *i.i.d* conditions, we propose a PC-SLDC algorithm for clustering the empirical SSPs with similar distribution. Then we propose an STI-KNN algorithm to map the target inversion task, so that proper training samples for the task can be found. To address the negative learning problem in ML, only clusters having the positive correlation with the task can be chosen as training tasks, and the learning rates of different base learners change with the similarity between the meta training data and the task training data. The experiment results show that the TDML has better generalization ability compared with other learning methods for SSP inversion, that is, the good accuracy performance is not only obtained in the model training stage, but also maintained in the SSP inversion (testing) stage. Moreover, the TDML inherits the advantage of time efficiency of ANN during the inversion stage.

Although TDML has better accuracy performance compared with AEFMNN, TL, ML, there are still some factors that limit the performance of TDML. 1)High noise level of signal propagation time that beyond the bearing capacity of AEFMNN will affect the SSP inversion accuracy. 2)The mapping accuracy of a given task to the SSP distribution cluster it belongs to has great influence on the confidence coefficient performance of SSP inversion result. 3)The SSP inversion accuracy will be limited when the real SSP distribution of a given task lays out of distribution coverage of reference SSPs, though it is accurately mapped to a cluster. For example, assume the time of SSP inversion task and most of its neighbor reference SSPs with least spatio-temporal distance is ideally the same, however, the location of SSP inversion task is at the external margin of the area constructed by the sampling location of reference SSPs. In this case, the TDML will not able to accurately invert the SSP of the task due to the spatial difference of SSP distribution, the problem of which is also exist in other SSP inversion methods.

In our future work, we are going to further verify the TDML in both shallow and deep ocean experiments, and apply the TDML SSP inversion method to underwater positioning and navigation systems.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

WH: conceptualization, methodology, formal analysis, investigation, software, original draft writing. FY: conceptualization, methodology. DL: writing review and editing. HZ: supervision, writing review and editing. TX: writing review and editing, project

administration. All authors contributed to the article and approved the submitted version

Funding

This work was financially supported by Laoshan Laboratory (LSKJ202205104), China Postdoctoral Science Foundation (2022M722990), Qingdao Postdoctoral Science Foundation (QDBSH20220202061), National Natural Science Foundation of China (NSFC:62271459), National Defense Science and Technology Innovation Special Zone Project: Marine Science and Technology Collaborative Innovation Center (22-05-CXZX-04-01-02), and the Fundamental Research Funds for the Central Universities, Ocean University of China (202313036).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alet, F., Schneider, M. F., Lozano-Perez, T., and Kaelbling, L. P. (2020). "Meta-learning curiosity algorithms," in *International Conference on Learning Representation 2020 (ICLR)*, Addis Ababa, Ethiopia: OpenReview.net. 1–22. doi: 10.48550/arXiv.2003.05325

Benson, J., Chapman, N. R., and Antoniou, A. (2000). Geoacoustic model inversion using artificial neural networks. *Oceans* 1, 446–451. doi: 10.1088/0266-5611/16/6/302

Bianco, M., and Gerstoft, P. (2017). Dictionary learning of sound speed profiles. *J. Acoustical Soc. America* 140, 1749–1758. doi: 10.1121/1.4977926

Bianco, M. J., Gerstoft, P., Traer, J., Ozanich, E., Roch, M. A., Gannot, S., et al. (2019). Machine learning in acoustics: theory and applications. *J. Acoustical Soc. America* 146, 3590–3628. doi: 10.1121/1.5133944

Boyer, T., Baranova, O., Coleman, C., Garcia, H., Grodsky, A., Locarnini, A., et al. (2021) *World ocean database 2018*. Available at: https://www.ncei.noaa.gov/access/648world-ocean-database/datawodgeo.html.

Carroll, P., Mahmood, K., Zhou, S., Zhou, H., Xu, X., and Cui, J.-H. (2014). On-demand asynchronous localization for underwater sensor networks. *IEEE Trans. Signal Process.* 62, 3337–3348. doi: 10.1109/TSP.2014.2326996

Chang, H., Han, J., Zhong, C., Snijders, A. M., and Mao, J. (2018). Unsupervised transfer learning *via* multi-scale convolutional sparse coding for biomedical applications. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 1182–1194. doi: 10.1109/TPAMI.2017.2656884

Choo, Y., and Seong, W. (2018). Compressive sound speed profile inversion using beamforming results. *Remote Sens.* 10, 1–18. doi: 10.3390/rs10050704

Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. (2019). "Quantifying generalization in reinforcement learning," in *Proceedings of the 36th International Conference on Machine Learning*. (Long Beach, California, USA: PMLR) 97, 1282–1289 doi: 10.48550/arXiv.1812.02341

Dinn, D. F., Loncarevic, B. D., and Costello, G. (1995). "The effect of sound velocity errors on multi-beam sonar depth accuracy," in *'Challenges of Our Changing Global Environment'. Conference Proceedings, OCEANS'95 MTS/IEEE*. (San Diego, CA, USA: IEEE) 2, 1001–1010. doi: 10.1109/OCEANS.1995.528559

Elsken, T., Metzen, J. H., and Hutter, F. (2019). Neural architecture search: A survey. *J. Mach. Learn. Res.* 20, 1997–2017. doi: 10.5555/3322706.3361996

Erol-Kantarci, M., Mouftah, H. T., and Oktug, S. (2011). A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Commun. Surveys Tutorials* 13, 487–502. doi: 10.1109/SURV.2011.020211.00035

Finn, C., Abbeel, P., and Levine, S. (2017). "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning (ICML'17)*. (Sydney, NSW, Australia: JMLR.org.) 70, 1126–1135.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning* (Cambridge, Massachusetts, USA: MIT Press).

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. (2020). IEEE Transactions on Pattern Analysis and Machine Intelligence. *Meta-Learning in Neural Networks: A Survey arXiv:2004.05439v2*. (IEEE) 44 (9), 5149–5169. doi: 10.1109/TPAMI.2021.3079209

Huang, W., Li, D., and Jiang, P. (2018). "Underwater sound speed inversion by joint artificial neural network and ray theory," in *Proceedings of the Thirteenth ACM International Conference on Underwater Networks & Systems (WUWNet'18)*. (Shenzhen, China: ACM) 1–8. doi: 10.1145/3291940.3291972

Huang, W., Liu, M., Li, D., Cen, Y., and Wang, S. (2019). "A stratified linear sound speed profile simplification method for localization correction," in *Proceedings of the International Conference on Underwater Networks and Systems, WUWNET'19*. (New York, NY, USA: ACM) 30, 1–6. doi: 10.1145/3366486.3366517

Huang, W., Liu, M., Li, D., Yin, F., Chen, H., Zhou, J., et al. (2021). Collaborating ray tracing and ai model for auv-assisted 3-d underwater sound-speed inversion. *IEEE J. Oceanic Eng.* 46, 1372–1390. doi: 10.1109/JOE.2021.3066780

Isik, M. T., and Akan, O. B. (2009). A three dimensional localization algorithm for underwater acoustic sensor networks. *IEEE Trans. Wireless Commun.* 8, 4457–4463. doi: 10.1109/TWC

Jensen, F. B., Kuperman, W. A., Porter, M. B., and Schmidt, H. (2011). *Computational ocean acoustics: Chapter 1* (New York, NY, USA: Springer Science & Business Media). doi: 10.1008/978-1-4419-8-8

Jin, G., Liu, F., Wu, K., and Chen, C. (2020). Deep learning-based framework for expansion, recognition and classification of underwater acoustic signal. *J. Exp. Theor. Artif. Intell.* 32, 205–218. doi: 10.1080/0952813X.2019.1647560

Komen, D. F. V., Neilsen, T. B., Howarth, K., Knobles, D. P., and Dahl, P. H. (2020). Seabed and range estimation of impulsive time series using a convolutional neural network. *J. Acoustical Soc. America* 147, 403–408. doi: 10.1121/10.0001216

Kussat, N., Chadwell, C., and Zimmerman, R. (2005). Absolute positioning of an autonomous underwater vehicle using gps and acoustic measurements. *IEEE J. Oceanic Eng.* 30, 153–164. doi: 10.1109/JOE.2004.835249

Li, Z., He, L., Zhang, R., Li, F., Yu, Y., and Lin, P. (2015). Sound speed profile inversion using a horizontal line array in shallow water. *Sci. China Physics Mechanics Astronomy* 58, 1–7. doi: 10.1007/s11433-014-5526-x

Li, Q., Shi, J., Zhenglin, L., Yu, L., and Zhang, K. (2019). Acoustic sound speed profile inversion based on orthogonal matching pursuit. *Acta Oceanol. Sin.* 38, 149–157. doi: 10.1007/s13131-019-1505-4

Li, F., and Zhang, R. (2010). Inversion for sound speed profile by using a bottom mounted horizontal line array in shallow water. *Chin. Phys. Lett.* 27, 084303:1–4. doi: 10.1088/0256-307X/27/8/084303

Liu, J., Wang, Z., Cui, J.-H., Zhou, S., and Yang, B. (2015). A joint time synchronization and localization design for mobile underwater sensor networks. *IEEE Trans. Mobile Comput.* 15, 530–543. doi: 10.1109/TMC.2015.2410777

Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. *In Proc. icml.* 1, 1–6. Available at: http://robotics.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf

Michalopoulou, Z. H., Alexandrou, D., and De Moustier, C. (1993). Application of neural and statistical classifiers to the problem of seafloor characterization. *IEEE J. Oceanic Eng.* 20, 190–197. doi: 10.1109/48.393074

Misra, P., and Enge, P. (2006). *Global Positioning System-Signals, Measurements, and Performance. 2nd ed.* (Lincoln, MA, USA: Ganga-Jamuna Press).

Munk, W., and Wunsch, C. (1979). Ocean acoustic tomography: A scheme for large scale monitoring. *Deep Sea Res. Part A. Oceanog. Res. Papers* 26, 123–161. doi: 10.1016/0198-0149(79)90073-6

Munk, W., and Wunsch, C. (1983). Ocean acoustic tomography: Rays and modes. *Rev. Geophysics* 21, 777–793. doi: 10.1029/RG021i004p00777

Nichol, A., Achiam, J., and Schulman, J. (2018). On first-order meta-learning algorithms. *arXiv:1803.02999v3*, 1–15. doi: 10.48550/arXiv.1803.02999

Pan, S. J., and Yang, Q. (2010). A survey on transfer learning. *IEEE Trans. Knowledge Data Eng.* 22, 1345–1359. doi: 10.1109/TKDE.2009.191

Pérez-Rúa, J.-M., Zhu, X., Hospedales, T., and Xiang, T. (2020). "Incremental few-shot object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. (Seattle, WA, USA: IEEE). 13843–13852. doi: 10.1109/CVPR42600.2020.01386

Piccolo, J., Haramuniz, G., and Michalopoulou, Z. H. (2019). Geoacoustic inversion with generalized additive models. *J. Acoustical Soc. America* 145, 463–468. doi: 10.1121/1.5110244

Ravi, S., and Larochelle, H. (2017). "Optimization as a model for few-shot learning," in *International Conference on Learning Representation 2017 (ICLR)*. (Toulon, France: OpenReview.net.) 1–11.

Rich, C. (1997). "Multitask learning," in *Machine Learning*. (New York, NY, USA: Springer US). 28, 41–75. doi: 10.1023/A:1007379606734

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back propagating errors. *Nature* 5, 533–536. doi: 10.1038/323533a0

Shang, E. (1989). Ocean acoustic tomography based on adiabatic mode theory. *J. Acoustical Soc. America* 85, 1531–1537. doi: 10.1121/1.397355

Snell, J., Swersky, K., and Zemel, R. (2017). "Prototypical networks for few-shot learning," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*. (Red Hook, NY, USA: Curran Associates Inc.) NIPS'17, 4080–4090. doi: 10.5555/3294996.3295163

Stephan, Y., Thiria, S., and Badran, F. (1995). Inverting tomographic data with neural nets. *'Challenges Our Changing Global Environment'. Conf. Proc.* 3, 1501–1504. doi: 10.1109/OCEANS.1995.528711

Sun, W. C., Bao, J. Y., Jin, S. H., Xiao, F. M., and Cui, Y. (2016). Inversion of sound velocity profiles by correcting the terrain distortion. *Geomatics Inf. Sci. Wuhan Univ.* 41, 349–355. doi: 10.13203/j.whugis20140142

Tang, J.-f., and Yang, S.-e. (2006). Sound speed profile in ocean inverted by using travel time. *J. Harbin Eng. Univ. (In Chinese)* 27, 733–737. doi: 10.3969/j.issn.1006-7043.2006.05.022

Thomson, D. J., Dosso, S. E., and Barclay, D. R. (2018). Modeling auv localization error in a long baseline acoustic positioning system. *IEEE J. Oceanic Eng.* 43, 955–968. doi: 10.1109/JOE.2017.2771898

Tolstoy, A., Diachok, O., and Frazer, L. (1991). Acoustic tomography *via* matched field processing. *J. Acoustical Soc. America* 89, 1119–1127. doi: 10.1121/1.400647

Vanschoren, J. (2018). Meta-learning: A survey. *arXiv:1810.03548v1*, 1–29. doi: 10.48550/arXiv.1810.03548

Weiss, K., Khoshgoftaar, T. M., and Wang, D. (2016). A survey of transfer learning. *J. Big Data* 3, 1–40. doi: 10.1186/s40537-016-0043-6

Wu, Q., and Xu, W. (2017). Matched field source localization as a multiple hypothesis tracking problem. *Proc. Int. Conf. Underwater Networks Syst. (WUWNet'17) (ACM)* 25, 1–2. doi: 10.1145/3148675.3148723

Yang, Y., and Hospedales, T. (2016). Deep multi-task representation learning: A tensor factorisation approach. *International Conference on Learning Representations (ICLR)*, Toulon, France. OpenReview.net. arXiv:1605.06391v1., 1–12. doi: 10.48550/arXiv.1605.06391

Yin, F., Pan, L., Chen, T., Theodoridis, S., and Luo, Z.-Q. T. (2020). Linear multiple low-rank kernel based stationary gaussian processes regression for time series. *IEEE Trans. Signal Process.* 68, 5260–5275. doi: 10.1109/TSP.2020.3023008

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? *Proceedings of the 27th International Conference on Neural Information Processing Systems*. (Cambridge, MA, USA: MIT Press). 2, 3320–3328. doi: 10.48550/arXiv.1411.1792

Zhang, Z. M. (2005). "The Study for Sound Speed Inversion in Shallow Water on Application of Genetic and Simulated Annealing Algorithms. Master's thesis, chapter 4," (Harbin, China: Harbin Engineering University). doi: 10.7666/d.y780567

Zhang, W. (2013). *Inversion of Sound Speed Profile in Three-dimensional Shallow Water (in Chinese). Phdthesis, chapter 2* (Harbin, China: Harbin Engineering University).

Zhang, M., Xu, W., and Xu, Y. (2015). Inversion of the sound speed with radiated noise of an autonomous underwater vehicle in shallow water waveguides. *IEEE J. Oceanic Eng.* 41, 204–216. doi: 10.1109/JOE.2015.2418172

Zhang, W., Yang, S.-e., Huang, Y.-w., and Li, L. (2012). "Inversion of sound speed profile in shallow water with irregular seabed." In *AIP Conference Proceedings*. Beijing, China: AIP) 1495, 392–399.

Zheng, G. Y., and Huang, Y. W. (2017). Improved perturbation method for sound speed profile inversion. *J. Harbin Eng. Univ. (in Chinese)* 38, 371–377. doi: 10.11990/jheu.201603075

Zhou, Z., Cui, J., and Zhou, S. (2010). Efficient localization for large-scale underwater sensor networks. *Ad Hoc Networks* 8, 267–279. doi: 10.1016/j.adhoc.2009.08.005

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 22 September 2023
DOI 10.3389/fmars.2023.1186343

Check for updates

OPEN ACCESS

EDITED BY
Oliver Zielinski,
Leibniz Institute for Baltic Sea Research (LG), Germany

REVIEWED BY
Duane Edgington,
Monterey Bay Aquarium Research Institute (MBARI), United States
Nils Piechaud,
Norwegian Institute of Marine Research (IMR), Norway
Giovanni Volpe,
University of Gothenburg, Sweden
Harshith Bachimanchi,
University of Gothenburg, Sweden,
in collaboration with reviewer GV

*CORRESPONDENCE
Xuemin Cheng
chengxm@sz.tsinghua.edu.cn

†These authors have contributed equally to this work

RECEIVED 14 March 2023
ACCEPTED 24 August 2023
PUBLISHED 22 September 2023

CITATION
Yue J, Chen Z, Long Y, Cheng K, Bi H and Cheng X (2023) Toward efficient deep learning system for *in-situ* plankton image recognition.
Front. Mar. Sci. 10:1186343.
doi: 10.3389/fmars.2023.1186343

COPYRIGHT
© 2023 Yue, Chen, Long, Cheng, Bi and Cheng. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Toward efficient deep learning system for *in-situ* plankton image recognition

Junbai Yue[1†], Zhenshuai Chen[1†], Yupu Long[1], Kaichang Cheng[1], Hongsheng Bi[2] and Xuemin Cheng[1*]

[1]Shenzhen International Graduate School, Tsinghua University, Shenzhen, Guangdong, China, [2]University of Maryland Center for Environmental Science, Solomons, MD, United States

Plankton is critical for the structure and function of marine ecosystems. In the past three decades, various underwater imaging systems have been developed to collect *in-situ* plankton images and image processing has been a major bottleneck that hinders the deployment of plankton imaging systems. In recent years, deep learning methods have greatly enhanced our ability of processing *in-situ* plankton images, but high-computational demands and longtime consumption still remain problematic. In this study, we used knowledge distillation as a framework for model compression and improved computing efficiency while maintaining original high accuracy. A novel inter-class similarity distillation algorithm based on feature prototypes was proposed and enabled the student network (small scale) to acquire excellent ability for plankton recognition after being guided by the teacher network (large scale). To identify the suitable teacher network, we compared emerging Transformer neural networks and convolution neural networks (CNNs), and the best performing deep learning model, Swin-B, was selected. Utilizing the proposed knowledge distillation algorithm, the feature extraction ability of Swin-B was transferred to five more lightweight networks, and the results had been evaluated in taxonomic dataset of *in-situ* plankton images. Subsequently, the chosen lightweight model and the Bilateral–Sobel edge enhancement were tested to process *in-situ* images with high level of noises captured from coastal waters of Guangdong, China and achieved an overall recall rate of 91.73%. Our work contributes to effective deep learning models and facilitates the deployment of underwater plankton imaging systems by promoting both accuracy and speed in recognition of plankton targets.

KEYWORDS

***in-situ* plankton images, image processing, knowledge distillation, model deployment, deep learning**

1 Introduction

Plankton play a pivotal role in marine food webs and are essential for integrated ecosystem assessment (Brun et al., 2015; Piredda et al., 2017; Braz et al., 2020). For example, plankton often provide information on living resources (Wang et al., 2022), environmental conditions (Lv et al., 2022), and fisheries (Azani et al., 2021). Effective monitoring of

plankton allows researchers to deduce their dynamics and identify the underlying processes (Bi et al., 2022). Thus underwater imaging systems are increasingly being deployed to collect *in-situ* plankton images on various platforms (Davis et al., 1996; Benfield et al., 2000; Gorsky et al., 2000; Cowen and Guigand, 2008) to estimate abundances of different plankton groups and examine their spatial and temporal dynamics (Bi et al., 2013; Hermand et al., 2013; Guo et al., 2018; Luo et al., 2018). In recent years, imaging systems have increasingly been used for high-frequency long-term plankton monitoring (Campbell et al., 2020; Orenstein et al., 2020; Song et al., 2020; Bi et al., 2022).

In plankton image processing, it is difficult to balance accuracy and processing speed. To improve accuracy, researchers utilize not only advanced optical mechanisms to acquire more information (Buskey and Hyatt, 2006; Hermand et al., 2013; Guo et al., 2018) but also deep learning systems to achieve high accuracy (Li and Cui, 2016; Luo et al., 2018; Kyathanahally et al., 2021; Li et al., 2021; Kyathanahally et al., 2022). As a result of these evolutions, the speeds of computing have dropped, making it difficult to deploy excellent algorithms on site because of the following: (1) The amount of raw data increases with the continuous sampling; (2) neural networks in deep learning have a huge number of parameters and computations; (3) as data transmission is often limited in open ocean, the processing ability of underwater computing hardware is extremely limited. Therefore, it is necessary to develop portable data processing procedures for independent underwater equipment to deal with abovementioned problems. In other words, the algorithm should be improved in terms of computing speed and storage capacity while ensuring the accuracy and generalization.

In the era of deep learning, researchers try to compress the neural network models to reduce the amount of parameters and complexity of calculation. The mainstream methods include model pruning (Tanaka et al., 2020), model quantization (Fan et al., 2020), parameter sharing (Wu et al., 2018), and knowledge distillation (Hinton et al., 2015). Knowledge distillation is able to realize the interaction of parameters and features among multiple neural networks and possesses excellent performance and flexibility. In general, large-scale models tend to have better learning abilities and can accurately extract the key features of the samples in datasets. According to the core idea of knowledge distillation, large-scale models are taken as the teacher networks, and the iterative operations aim to reduce the loss function between the probability distributions or feature vectors output of the teacher networks and other smaller scale models (called the student networks). With the progress of training, the student networks gradually learn the feature extraction mechanisms guided by the teacher networks. It means that small-scale models can achieve equal accuracy in specific tasks as large-scale models through this method. Knowledge distillation was proposed by Hinton et al., 2015 and initially used Kullback–Leibler (KL) divergence as the loss function. Subsequently, various works were proposed in multiple distillation strategies. For example, Romero et al., 2014 proposed the distillation method using feature maps computed by middle layers in neural network (FitNet). Peng et al., 2019 and Tung and Mori, 2019 demonstrated the distillation processes based on correlation congruence (CC) and similarity preserving (SP), respectively. Similarly, it is also worth exploring to propose model compressing techniques in the scenarios of *in-situ* plankton image processing.

Based on the characters of PlanktonScope (an *in-situ* underwater imaging system proposed by Bi et al., 2022 and attached algorithm pipeline), the present study attempts to introduce knowledge distillation method and demonstrate efficient detection and recognition tasks on *in-situ* plankton images. We designed and implemented an inter-class similarity distillation algorithm based on feature prototype projection (prototype projection distillation, PPD) to realize the compression of forward calculation model. In order to seek the appropriate teacher network and ensure the original accuracy, we carried out a comparative study and examined the accuracy of five convolution neural networks (CNNs) and three Transformer architectures. Combined with transfer learning, the Swin-B network model (from Transformer architectures) was found to express the highest accuracy and was selected as the preliminary algorithm for classification (teacher network). Meanwhile, a Bilateral–Sobel edge enhancement method was proposed to highlight the edge pixel regions of targets to suppress the noise and background of *in-situ* images. This technique aimed to solve the segmentation difficulties caused by noise stickiness and edge destruction. Finally, the selected student networks and Bilateral–Sobel edge enhancement were integrated into algorithm pipeline, and these schemes were evaluated in accuracy and time consumption on the dataset captured via PlanktonScope in the coastal areas of Guangdong, China.

2 Materials and methods

The knowledge distillation and edge enhancement method are employed in the procedures of recognition and detection in algorithm pipeline, respectively. In *Section 2.1*, the algorithm pipeline of PlanktonScope is presented and the datasets applied in experiments are described. In *Section 2.2*, the basic theory and mathematical model of the proposed inter-class similarity knowledge distillation method based on feature prototype projection (PPD) are illustrated in details. In *Section 2.3*, as candidates for the teacher network in distillation, Transformer and CNN model families are described. In *Section 2.4*, the Bilateral–Sobel edge enhancement algorithm used to improve effect of detection is presented.

2.1 Description of algorithm pipeline and datasets

2.1.1 Basic algorithm pipeline of PlanktonScope

Figure 1 presents the content of algorithm pipeline. Plankton image detection and recognition include two stages: extraction and classification (Bi et al., 2015). Extraction is to extract the pixel regions of the targets from the *in-situ* images to separate the targets and background. Classification is to extract the features of the

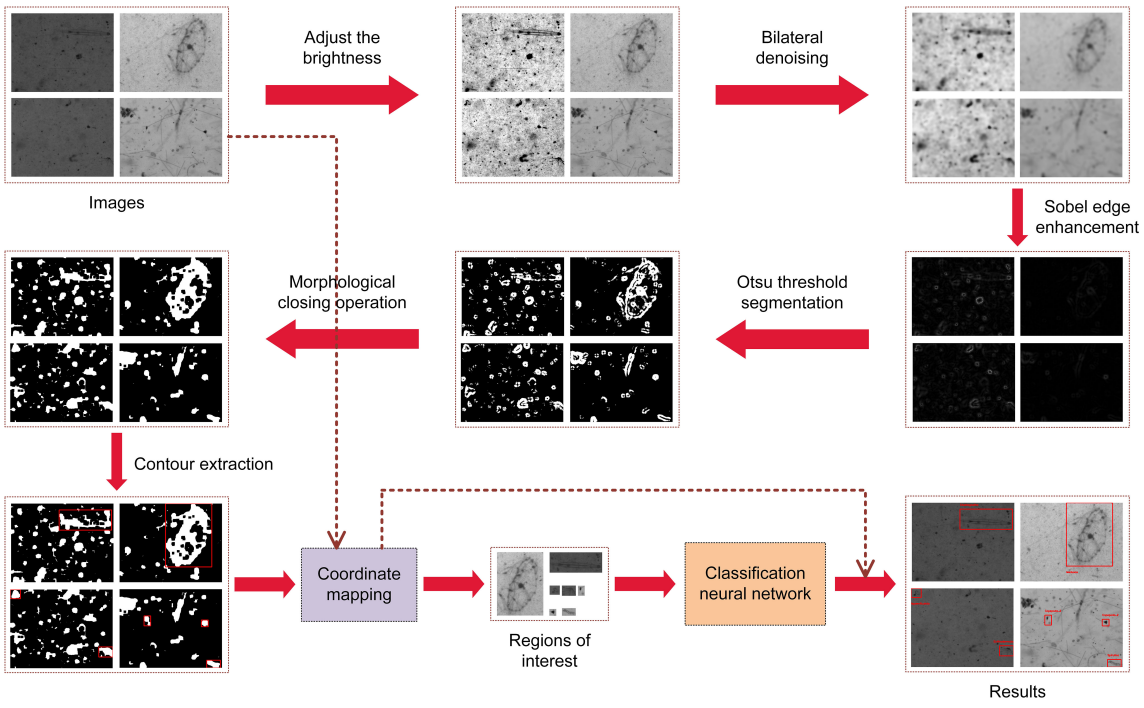


FIGURE 1
The flowchart of the algorithm pipeline and related results.

segmented targets and judge the class of the targets according to their features. The steps can be summarized as follows: (1) input the *in-situ* image and adjust the brightness; (2) operate denoising and edge enhancement; (3) implement the threshold segmentation proposed by (Otsu, 1979) based on maximum between-cluster variance to finish binarization; (4) demonstrate the morphological closing operation (Said et al., 2016) to fill the discontinuities, holes, and edge breaks; (5) implement the contour extraction based on boundary tracking (Suzuki, 1985; Marini et al., 2018) to obtain the regions of interest (ROIs) of the targets; (6) classify the detected targets using the selected calculation model; and (7) operate statistics of the quantity and species of plankton. The contributions of PPD method and Bilateral–Sobel edge enhancement are in steps (6) and (2), respectively.

2.1.2 Test dataset for detection and recognition tasks

The dataset for efficiency test of the proposed methods was collected by PlanktonScope in the coastal area of Guangdong, China. This dataset contains 209 *in-situ* images (2180 × 1635) for testing. These images are all 8-bit, and the whole set contains a total of 494 plankton targets, of which 258 are *Medusaes*. In addition, the other classes include *Copepoda*, *Spirulina*, *Appendicularia*, *Chaetognatha*, and *Echinodermata* (in Figure 2). The ground truths of ROIs are manually annotated. As the result of deep diving depth and illumination conditions of the monitoring system, the collected *in-situ* images are relatively dark, with pixel value of brightness ranging from 22 to 163. Even the human eye cannot distinguish a target in such weak contrast. Therefore, brightness adaptive processing is carried out for images:

$$I'_{u,v} = \begin{cases} p_{\max} & I_{u,v} > p_{\max} \\ I_{u,v} & p_{\max} \geq I_{u,v} \geq p_{\min} \\ p_{\min} & I_{u,v} < p_{\min} \end{cases} \tag{1}$$

$$I''_{u,v} = \frac{255(I'_{u,v} - p_{\min})}{p_{\max} - p_{\min}} \tag{2}$$

When the pixel values of one image are sorted, if the first 1% and last 1% pixel values are removed, $p_{\min}$ to $p_{\max}$ is the value range of rest pixels. Moreover, Equation 1 removes the extreme values, and Equation 2 normalizes the other values to obtain the final result of brightness adjustment. Figures 2A, B show a pair of original and processed images.

2.1.3 Plankton dataset for classification training

To train and evaluate the classification networks, we used a large-scale and standardized taxonomic dataset of plankton captured in the South China Sea. This dataset was created over a long period via PlanktonScope, and it has 30,720 segmented targets, which have been divided into 12 classes. Each class contains 2,560 images (8-bit), of which 2,048 are in the training set, and 512 in the test set. In addition, the size span of ROIs is in the range of 15^2–1200^2 (pixels). The actual field of view corresponding to one image is 4.796 cm × 3.597 cm, and one pixel converts to 22 microns. Figures 2C–N show examples from different classes.

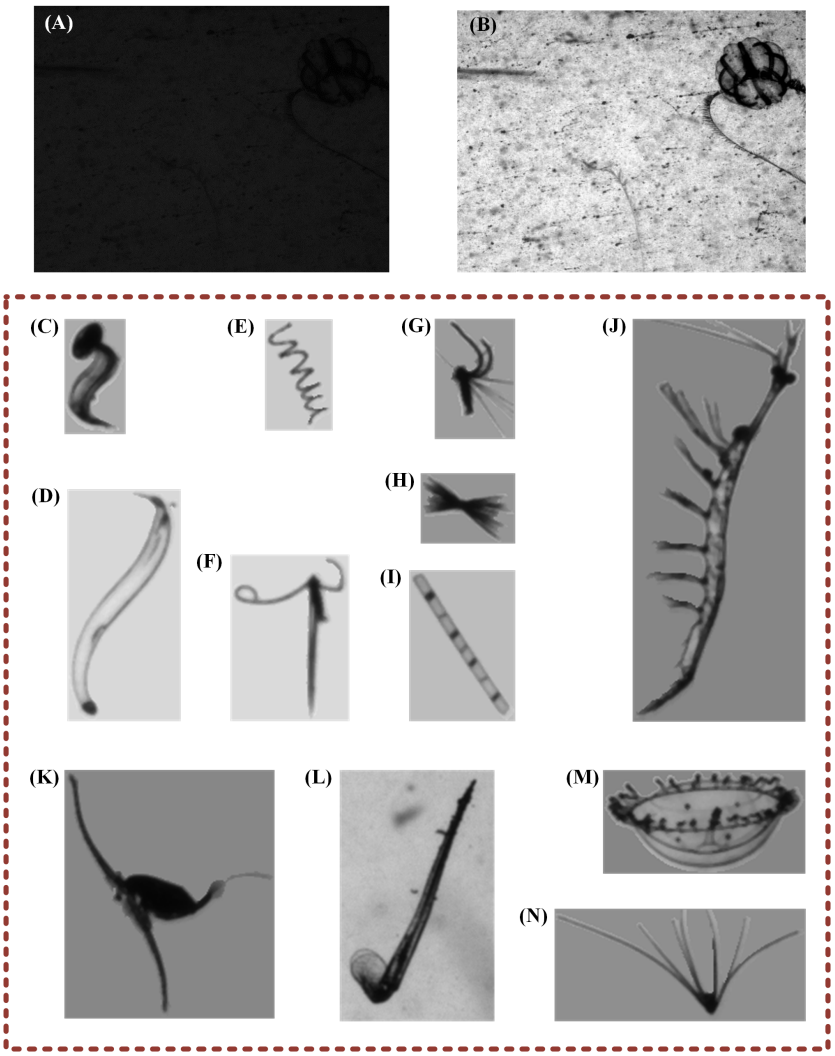


FIGURE 2
Samples of datasets. **(A)** Original image before brightness adjustment; **(B)** processed result after brightness adjustment; **(C–N)** examples of taxonomic dataset captured in South China Sea: **(C)** *Appendicularia*; **(D)** *Chaetognatha*; **(E)** *Spirulina*; **(F)** *Copepoda_1*; **(G)** *Copepoda_3*; **(H)** Unknown classes; **(I)** *Skeletonema*; **(J)** *Euphausiids*; **(K)** *Copepoda_2*; **(L)** *Creseis*; **(M)** *Medusae*; and **(N)** *Echinodermata*.

2.2 Knowledge distillation framework for model deployment

2.2.1 Basic theory of the method

Processing image on-site often suffers from limited computing hardware. Therefore, it is necessary to reduce the number of parameters and improve computing speed. An inter-class similarity distillation method based on feature prototypes projection (PPD) was proposed for model compression. This method can reduce the scale of parameters and time consumption under the maintenance of accuracy.

The intermediate data output by the hidden layers of neural networks are abstract representations after undergoing nonlinear calculations and feature transformations. These data are the results of feature extractions, and the corresponding calculations are the expected knowledge. The core idea of knowledge distillation is to impart expected knowledge from the teacher network (usually with large parameters and high-recognition performance) to the student network (usually with small parameters and high-computing speed). The expected knowledge is generally the intermediate or final result (feature or probability, etc.) from the teacher network (Romero et al., 2014; Hinton et al., 2015; Peng et al., 2019; Tung and Mori, 2019). Therefore, the loss function in the training of the student network consists of two parts: one is the cross-entropy (CE) loss L_1 between the real label and the logical value output from the student network and the other is the difference L_2 of the intermediate or final result between teacher and student networks. The linear combination of these two parts constitutes the final loss function $L(L = \alpha L_1 + (1 - \alpha)L_2)$ to guide the training, where the weight α balances the loss of the two parts and it is a hyperparameter which needs to be selected artificially. This hyperparameter α would bring great uncertainty to the distillation effect, so we proposed a distillation method without this hyperparameter through the experiments on the plankton *in-situ* images.

Figure 3 shows the overview of our distillation process. First, the teacher network was trained on taxonomic dataset and converged after multiple epochs. Then, we used the trained teacher network to calculate (extract) the features of all samples, and took the arithmetic mean value of features in each class as the respective feature prototypes c. Subsequently, the training of student network started. On forward calculation, both the teacher and student network operated the calculation (extraction) of all samples to obtain the feature expression t_i ands_i(the vectors output from hidden layers). Then, the cosine similarity between the features of all samples and the feature prototypes of each class is calculated to obtain $\varphi^{(Teacher)}$ and $\varphi^{(Student)}$. Therefore, we could arrange the results and obtain the inter-class similarity matrix of both teacher and student networks. Next, we took the mean square error (MSE) between the two matrices as loss function and operated back propagation.

The inter-class similarity matrix of the teacher network was regarded as the expected knowledge, so we only updated the parameters of the student network to learn the distribution of inter-class similarity. This resulted in the gradual improvement of the recognition accuracy of the student network. Compared with the classical knowledge distillation methods, the advantages of our method are as follows: (1) The selection of feature prototype helps to avoid the interference of feature outliers. (2) Only one loss function relying on inter-class similarity is used, without extra calculation of classification loss. (3) There is no need to set hyperparameters α, which reduces the impact of manual factors on performance.

2.2.2 Mathematic details of the model

The learning mechanism of neural network can be understood as the mapping from the sample space (input data) to the high-dimensional feature space. Using x_i and f_i to represent the sample and feature vectors, respectively, the cosine similarity between two feature vectors is defined as follows:

$$\sigma_{i,j} = \frac{f_i f_j^T}{\| f_i \|_2 \| f_j \|_2} \tag{3}$$

For classification tasks, the ideal situation is that the feature vectors of different classes are orthogonal to each other, and those of the same classes are toward the common direction, corresponding to 0 and 1 in similarity, respectively. The network is aimed at reducing the inter-class similarity and increasing the intra-class similarity. A trained network which satisfies the test standard is considered to satisfy the aforementioned requirements. The network can be regarded as a feature extractor$\mathcal{F}$to encode the sample vectors:

$$t_i = \mathcal{F}(x_i) \tag{4}$$

for the class labeled by$k$, we calculate the means of all vectors t_k in feature space $\mathcal{T}_k$ and normalize them by l_2-norm to obtain the feature prototype:

$$\boldsymbol{c}_k = \frac{\boldsymbol{t}'_k}{\| \boldsymbol{t}'_k \|_2} = \frac{\boldsymbol{t}'_k}{\sqrt{\sum_{j=1}^{D} (\boldsymbol{t}'_{k,j})^2}}, \ \boldsymbol{t}'_k = \frac{1}{M_k} \sum_{t_i \in \mathcal{T}_k} t_i \tag{5}$$

$$C = \mathrm{Concat}(c_1, c_2, \ldots, c_K) \tag{6}$$

where M_k donates the number of vectors labeled by class k and D donates the dimension of t_k. Equation 6 is the matrix representation of the combination of all classes' feature prototypes.

Furthermore, the inner product of the teacher feature t_i (also standardized by l_2-norm), and the feature prototype c_k is performed to obtain the cosine similarity distance, which is the expected knowledge in distillation, as shown in Equation 7. To simplify the calculation, the cosine similarity calculation between the teacher feature t_i and all feature prototypes can be obtained in the form of matrices, as shown in Equation 8.

$$\varphi_{i,k} = \frac{t_i c_k^T}{\| t_i \|_2} = \frac{\sum_{j=1}^{D} t_{i,j} c_{k,j}}{\sqrt{\sum_{j=1}^{D} t_{i,j}^2}} \tag{7}$$

$$\Phi(t_i) = \frac{t_i C^T}{\| t_i \|_2} = \frac{t_i C^T}{\sqrt{\sum_{j=1}^{D} (t_{i,j})^2}} \tag{8}$$

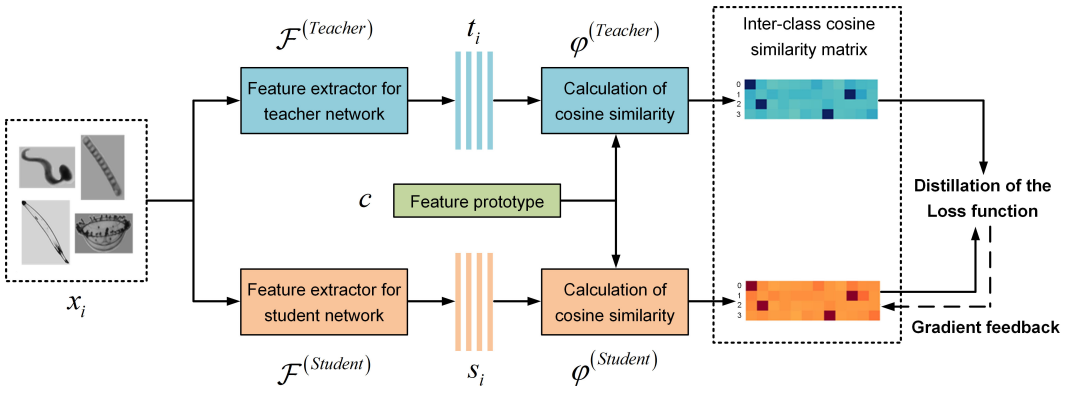


FIGURE 3
Process overview of the proposed PPD methods.

For the student network, the untrained encoder is considered unreliable. However, it can calculate the student feature s_i initially. Using Equations 7, 8, we obtain Equations 9, 10 to calculate the cosine similarity of student features:

$$\varphi_{i,k} = \frac{s_i c_k^T}{\| s_i \|_2} = \frac{\sum_{j=1}^{D} s_{i,j} c_{k,j}}{\sqrt{\sum_{j=1}^{D} s_{i,j}^2}} \tag{9}$$

$$\Phi(s_i) = \frac{s_i C^T}{\| s_i \|_2} = \frac{s_i C^T}{\sqrt{\sum_{j=1}^{D} (s_{i,j})^2}} \tag{10}$$

We calculate the MSE loss and use the gradient descent algorithm to guide the student network to learn the similarity between the individual samples encoded by the teacher network and finally improve the recognition ability of the lightweight networks. The loss function is expressed as follows:

$$\mathcal{L}_{PPD-MSE} = \frac{1}{N} \sum_{i=1}^{N} \| \Phi(t_i) - \Phi(s_i) \|_2^2 \tag{11}$$

2.3 Transformer models

Underwater plankton images are often acquired under suboptimal imaging conditions. Despite the complete extraction of ROIs, targets often remain visually unclear. A CNN model can continuously be iterated into a forward computing graph for feature extraction through gradient descent. The spatial perception of CNN is the regular expansion of receptive field with the convolutional layers increasing. This implies a fixed interaction mode of global and local information of the image and causes a trend of overfitting and parameter redundancy. Therefore, plagued with complex features and high requirements of data processing, new neural network architecture, that is, the Transformer was chosen to improve the recognition accuracy at the beginning of teacher networks' training. This network architecture has demonstrated its strong performance over CNN in ecological automatic classification (Kyathanahally et al., 2022).

Transformer was proposed by Google in 2017 (Vaswani et al., 2017) and has achieved great success in the field of natural language processing (NLP). It employs a multi-head attention mechanism to extract features at any distance in the entire text, so that a single piece of information can flexibly implement multi-position and cross-scale interactive encodings. In 2020, Vision Transformer (ViT) was proposed (Dosovitskiy et al., 2020), and the encoder part of the initial Transformer was applied to extract image features. This scheme achieved the highest results in various computer vision (CV) tasks. To further incorporate the characteristics of image processing, the hierarchy of feature interactions in sub-regions of image (tokens) and their internal pixels were considered, which led to the proposal of Swin Transformer (Liu et al., 2021). This network shows better performance in characterization process and improves computational efficiency, which renders it potentially applicable to various fields.

This study focuses on the performance of Transformer architectures on the plankton taxonomic dataset (*Section 2.1.3*). We utilized several CNN and Transformer neural networks to evaluate the classification accuracy and computing speed. Furthermore, given the effectiveness of transfer learning (Pan and Yang, 2010) in plankton classification studies (Orenstein and Beijbom, 2017; Lumini and Nanni, 2019), we introduced transfer learning to provide pre-trained models (PTMs) for neural networks. These PTMs showed excellent performance in general CV scenarios, and their parameters experienced many iterations on large-scale public datasets. In some applications with specific requirements, these models can reach the accuracy by secondly training on the small datasets and fine-tuning the parameters. Under traditional training modes, the same accuracy needs a large amount of data and training times. The pre-training is beneficial to save computing resources and reduce data consumption.

2.4 Bilateral–Sobel edge enhancement

We proposed an edge enhancement method for fragile image texture to preprocess the images. The edge enhancement was divided into two steps: suppression of high-frequency noise and highlight of visual edge. The kernel of Bilateral filtering (Tomasi and Manduchi, 1998; Bhonsle et al., 2012) was used, and on the basis of Gaussian kernel which considers the spatial relationship of pixels, it pays extra attention to the value distribution of adjacent pixels. Therefore, Bilateral filtering can protect the weak edge while denoising, so we choose it as the denoising procedure. In an odd-order Bilateral filtering kernel, the weights of matrix are set as follows:

$$G_{x,y} = \frac{1}{\tau_G} \exp\left(-\frac{x^2 + y^2}{2\sigma_G^2}\right) \tag{12}$$

$$W_{x,y,u,v} = \frac{1}{\tau_W} \exp\left(-\frac{(I_{u+x,u+y} - I_{u,v})^2}{2\sigma_W^2}\right) \tag{13}$$

where (u, v) denotes the global position of the central pixel; x and y represent the local coordinates of adjacent pixels; σ_G and σ_W are the standard deviations of the normal distribution; τ_G and τ_W are weight coefficients applied to ensure the sum of the weights in the kernels are 1; and I is the pixel value before processing. As one can see, in the spatial kernel G and value kernel W, the closer the adjacent pixel to the central pixel in Euclidean distance and grayscale value, respectively, the greater its contribution to smoothing calculation. Furthermore, the final kernel function B is the inner product of the two matrices.

The above design can prevent the smooth denoising from breaking slight and thin edges and, thus, preserve the complete foreground information within *in-situ* images. However, the foreground and background remain indistinguishable in case of close pixel values of areas. To extract the objects submerged into the background, we further applied the Sobel operator (Vincent and Folorunso, 2009) to completely separate the edge part in the

gradient dimension for the images obtained after bilateral filtering. The gradient values in the two directions of images, S_x and S_y are calculated using standard Sobel kernels D_x and D_y, respectively, and synthesize into the final result S_{xy} through vector addition. The entire process is expressed as follows:

$$D_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, D_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, S_{xy} = \sqrt{S_x^2 + S_y^2} \quad (14)$$

After the gradient image S_{xy} is obtained through the above steps, and it is used as input for steps (3) to (7) in the algorithm pipeline described in *Section 2.1.1*.

3 Results

In order to examine the effectiveness of proposed methods and their contribution to the performance of algorithm pipeline, in this part, we designed a set of experiments and present the results. The sequence of results is shown in the order of algorithm pipeline. In *Section 3.1*, the effects of Bilateral–Sobel edge enhancement on *in-situ* images from test dataset (*Section 2.1.2*) are shown. In *Section 3.2*, we compared the performance of CNN and Transformer families on taxonomic dataset (*Section 2.1.3*), and took the outstanding model as the teacher network to verify the superiority of the knowledge proposed distillation method over the traditional ones in *Section 3.3*. Finally, in *Section 3.4*, we validated the selected methods using test dataset, and paid extra attention to the results on gelatinous plankton (*Medusae*). These experiments were conducted on the same computing hardware, using an Intel Core i7-8750H processor, 16GB of RAM, and Nvidia GeForce GTX 1060 graphics cards.

3.1 Effects of Bilateral–Sobel edge enhancement

3.1.1 Visualization of Gaussian, Bilateral, and Sobel processing results

First, we compared Gaussian and Bilateral operators to filter an image of an individual of *Medusae* and evaluated the results of subsequent binarization. As shown in Figure 4A, the boundary on both sides of the upper part in the raw image is weakly connected. Upon the application of Gaussian filtering, as shown in Figure 4C, the concerned edge breaks, whereas Bilateral filtering retains the shape of the edge to the best extent (Figure 4B).

Figures 4D–G show the independent and united results of the Bilateral and Sobel operators. As shown in Figure 4E, it is obvious that the single gradient calculation cannot suppress the high-frequency noise of the background. Although a single Bilateral filter can preserve the weak edges as much as possible while denoising, some too weak edges are still stick together with the background (Figure 4F). This will make some background regions be recognized as part of ROIs. Therefore, we used a combination of Bilateral–Sobel edge enhancement to perform a comprehensive operation in spatial, value, and gradient domains, so as to achieve complete segmentation of the target in binarization step.

3.1.2 Comparative experiments on edge enhancement

In order to quantitatively analyze the effect of Bilateral–Sobel edge enhancement and other preprocessing methods on target extraction, we used steps (1)–(5) of the algorithm pipeline described in *Section 2.1.1* for target extraction. We used the find contours function in the OpenCV library for target extraction at step (5). In addition, we set the

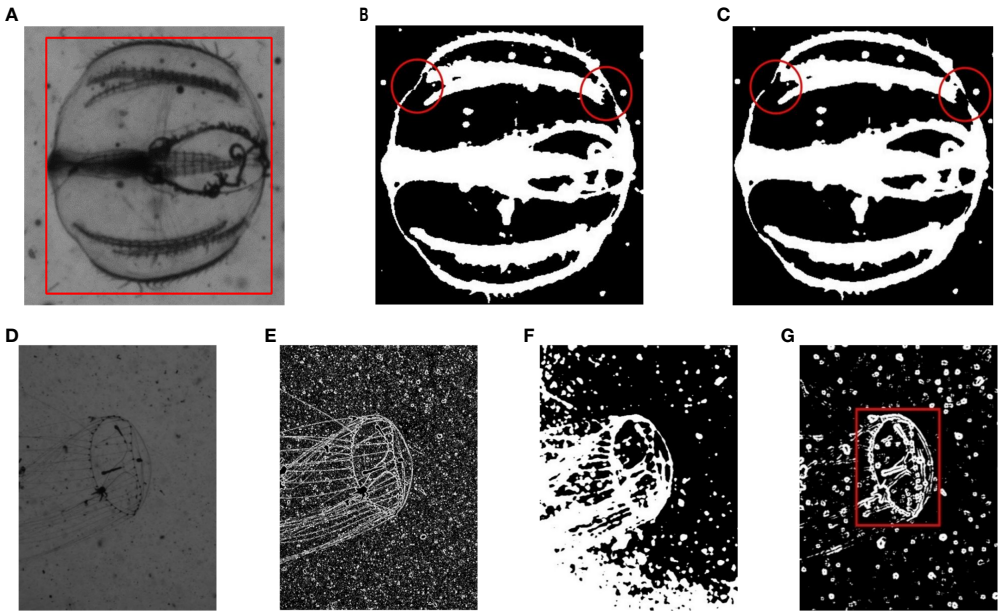


FIGURE 4
Visual evaluation of enhancement methods. **(A)** Before processing; **(B)** Bilateral filtering; **(C)** Gaussian filtering; **(D)** before processing; **(E)** operation by Sobel kernel only; **(F)** operation by Bilateral kernel only; and **(G)** operation by Bilateral and Sobel kernels; **(B–G)** experienced subsequent binarization.

denoising and edge enhancement operations for step (2) as follows: no denoising and edge enhancement, only Gaussian filtering, only Bilateral filtering, combination of Gaussian filtering and Sobel gradient calculation, and combination of Bilateral filtering and Sobel gradient calculation. All experimental subjects were raw images from the dataset presented in *Section 2.1.2*. The evaluation indicators were the precision (the quantity ratio of complete ROIs to extracted ROIs) and recall (the quantity ratio of complete ROIs to the total targets), as well as the extraction speed [number of images processed by steps (1) to (5) within 1 s]. In *in-situ* images, some targets have blurry edges, which can easily cause edge breaks during the process of extraction, resulting in one target being divided into multiple ROIs. The complete ROI refers to the fact that the specific target does not have broken pixel connections, which means that the complete ROI does not share a target with other ROIs. The results present in Table 1 show that our preferred method exhibits the best extraction result, implying our edge enhancement renders the target much easier to be detected.

3.2 Performance of CNN and Transformer schemes

In this section, we compared the performance of neural networks with extensive parameter volumes both in CNN and Transformer families. We demonstrated the test on the taxonomic dataset from South China Sea (*Section 2.1.3*). Furthermore, the effectiveness of the parameters pre-trained by the ImageNet dataset (Ridnik et al., 2021) was verified in the plankton classification task. In this experiment, MobileNet V2 (Sandler et al., 2018), ShuffleNet V2 (Ma et al., 2018), ResNet50, ResNet101, and ResNet152 (He et al., 2016) were selected from the CNN architectures; Swin-T (Liu et al., 2021), ViT-B (Dosovitskiy et al., 2020), and Swin-B (Liu et al., 2021) were selected from the Transformer architectures. We conducted two types of training modes: (1) direct training on the taxonomic dataset and (2) loading the pre-training model and then fine-tuning by the taxonomic dataset. The accuracy (the number of correctly classified samples divided by the total number of samples) results and the size (quantified as storage memories) of models are presented in Table 2. The CE loss function was used in the training process.

As shown in Table 2, the best performance is reached by pre-trained Swin-B with an accuracy of 94.34%. Furthermore, for both two network families, transfer learning yields higher accuracy than direct training. In addition, the performances of the Transformer variants are inferior to that of the CNN variants in direct training when the network is initialized by random parameters. Thus, the Transformer architectures may not be suitable for medium and small-scale datasets without any priori information, and its feature perception is not as experienced as the mode of CNNs in this case. However, pre-training may equivalently improve the amount of data in the source domain, and resulted in the Transformers' performance exceeding that of the CNNs. We have discussed this situation at the end of this paper. From the results, we considered that the pre-trained Swin-B model stood out in the application of plankton classification and planned to integrate it into the following knowledge distillation algorithm.

3.3 Experimental results of the proposed knowledge distillation method

3.3.1 Comparison with classical knowledge distillation methods

The trained Swin-B model in *Section 3.2* was selected as the teacher network to guide the convergence of student network. This model occupies storage of 87M and its reasoning speed is 26 targets per second. We compared the proposed knowledge distillation method with the other four classic technologies reported in recent years mentioned in *Part 1*, including: KD: knowledge distillation (Hinton et al., 2015); FitNet (Romero et al., 2014); SP: similarity preserving (Tung and Mori, 2019); CC: correlation congruence(Peng et al., 2019); and CE: cross-entropy (Ferdous et al., 2020). Five neural networks with different parameter volumes and reasoning speeds were used as student networks. In addition, a multi-layer perceptron structure was used to match the output dimensions of student networks with the teacher network. Using the dataset described in *Section 2.1.3*, the final results of the five methods are presented in Table 3.

The column of CE (baseline) represents classification training by using cross entropy loss function, without any knowledge distillation processes. The accuracy achieved in this column is taken as the baseline. As shown in the table, the proposed method (PPD) guides five student networks to improve the accuracy (number of correctly classified samples divided by the total number of samples), and achieves a higher or nearly equal increase compared with other methods. Moreover, the accuracy of

TABLE 1 Results of comparative experiments on edge enhancement.

Methods	Precision (%)	Recall (%)	Extraction speed (images/s)
No denoising and edge enhancement	89.21	50.40	19
Gaussian	89.47	69.11	18
Bilateral	93.41	69.10	8
Sobel	79.25	17.07	16
Gaussian–Sobel	87.34	84.14	14
Bilateral–Sobel	98.73	94.71	7

TABLE 2 Performance of different neural networks and training strategies on taxonomic dataset.

Neural network		Size (megabytes)	Accuracy(%)	
			Random initialization of parameters	Pre-trained model
CNNs	MobileNet V2	0.3	86.47	90.97
	ShuffleNet V2	1.3	88.26	92.35
	Res50	24	90.84	93.23
	Res101	43	91.05	93.42
	Res152	58	89.55	92.99
Transformer	Swin-T	27	89.70	93.93
	ViT-B	86	88.54	94.09
	Swin-B	87	89.13	94.34

ShuffleNet V2 with the help of PPD (93.13%) exceeds ResNet50 under traditional training (93.02%), whereas the parameter volume of the former is only 5% of the latter. This implies that our method can make the lightweight network show better recognition ability than large scale neural networks under traditional training.

All networks use Adam (Kingma and Ba, 2014) as the training optimizer. After each epoch of training and validation, the model parameters were saved once, and the current highest validation accuracy rate was recorded. If the highest validation accuracy rate remained unchanged for several epochs, the learning rate was reduced (the learning rates of ShuffleNet V2 and MobileNet V2 are reduced by 10 times; the learning rate of ResNet50, Swin-T and ResNet101 are reduced by four times.) and load the model parameters corresponding to the highest accuracy to continue the training.

3.3.2 Evaluation of different loss functions

One of the key points of the proposed method is the similarity enhancement of feature descriptions between teacher and student networks. In the experiments above, we used MSE as the loss function (Equation 11), which usually appears in regression tasks. In this section, we discussed other two common loss functions from classification tasks: the CE and KL divergence loss functions. ShuffleNet V2 and Swin-T with better performance in Table 3 were used as the student network and the results are presented in Table 4. It can be seen that the MSE was most applicable to our frameworks, implying that the learning of our defined knowledge should be regarded as a regression process. The reasons for the poor performance of the other two loss functions can be inferred as follows: The dot product of the similarity matrix and the one-hot coding resulted in the loss of the relationship information between classes, leading the degradation of the final effect; Because the features are processed by the l_2-norm during distillation, the value of similarity was distributed in a narrow range of [−1,1], and both CE and KL loss need to perform softmax operation on the outputs similarity value; thus, they caused the output probability distribution being excessively smooth and weakening the positive response of intra-class features.

3.4 Examination of the update of algorithm pipeline

We finally demonstrated the experiments to examine the upgrade of algorithm pipeline, hoping that the quality of image processing can reach excellent performance. As for the

TABLE 3 Comparative experimental results of different knowledge distillation methods.

Student networks	Size (megabytes)	Classification speed (targets/s)	Accuracy(%)					
			CE (baseline)	PPD (ours)	CE + KD	CE + FitNet	CE + SP	CE + CC
ShuffleNet V2	1.3	301	92.35	93.13 (+0.78)	92.94 (+0.59)	92.92 (+0.57)	92.48 (+0.13)	93.15 (+0.80)
MobileNet V2	1.6	310	91.59	92.46 (+0.87)	92.51 (+0.92)	92.56 (+0.97)	91.94 (+0.35)	92.35 (+0.76)
ResNet50	26	76	93.02	93.62 (+0.60)	93.02 (+0.00)	93.41 (+0.39)	93.25 (+0.24)	93.12 (+0.10)
Swin-T	28	68	93.82	94.21 (+0.39)	94.22 (+0.40)	93.86 (+0.04)	94.04 (+0.22)	93.98 (+0.16)
ResNet101	45	36	93.23	93.82 (+0.59)	93.59 (+0.36)	93.15 (−0.08)	92.55 (+0.32)	93.20 (−0.03)

TABLE 4 Effect of PPD method with different loss functions.

Student networks	Accuracy(%)			
	CE	PPD-CE	PPD-KL	PPD-MSE
ShuffleNet V2	92.35	91.94 (−0.41)	92.30 (−0.05)	93.13 (+0.78)
Swin-T	93.82	93.73 (−0.09)	93.59 (−0.23)	94.21 (+0.39)

segmentation stage, the Bilateral–Sobel edge enhancement aided in the target extraction and location. In the stage of classification, we further verified the three student networks that performed well in the previous experiments (*Section 3.3.1*) and the selected teacher network, Swin-B (*Section 2.1.2*). In addition, *Medusae* is difficult in target extraction and classification due to its weak edge connection and similar gray value to background and so forth, so we paid extra attention to the detection effect of *Medusae*. The results are presented in Table 5.

It can be summarized that the trained Swin-B still exhibits the best performance. However, the model is very large and the processing time is more than 1 s, which is not suitable for terminal deployment. ShuffleNet V2 and Swin-T, which were guided by Swin-B with the proposed PPD, also perform better. The lightweight ShuffleNet V2 exhibits better performance than ResNet50 and requires only 273 milliseconds to process one *in-situ* image. Swin-T exhibits a better accuracy and also satisfies the acceptable storage capacity and processing speed.

4 Discussion

4.1 Deep understanding of the operations on plankton features

We applied knowledge distillation and updated the algorithm pipeline to pursue better detection and recognition effects of targets in plankton *in-situ* images. Here, it should be emphasized that our design inspirations of the methods focus on the mathematical operations on plankton features. In order to explain understandably, we define two temporary terms of plankton ROIs: (1) regional features, which represent the relative spatial position of ROIs in the background, and (2) classification features, which represent the class properties (including shape, texture features, etc.). Regional features and classification features are the features of ROIs in space and as objects, respectively. According to the steps of algorithm pipeline, we enhance the regional features and extract the classification features.

Bilateral–Sobel edge enhancement enhances the regional features of targets and makes them be easily separated. In the previous segmentation tasks, it is challenged to distinguish the targets, interference noise, and chaotic background. For example, as for gelatinous plankton, their narrow edges of and dense noises possess the same spatial frequency, and the gray scale of interest pixel region and background are visually fused. Therefore, ROIs, noises, and background are mixed in regional features and cannot be separated by single methods such as filtering. To solve these problems, we combined the distinguishing abilities of the filter kernel functions (Bilateral–Sobel operator) in the spatial, value, and gradient domains, to reduce the correlation of the mixed region features. In addition, the subsequent separation can be easily realized to obtain the complete ROIs. The verified experiments of the complete extraction reached the accuracy and recall rate of 98.73% and 94.73%, respectively.

For the classification steps, the discrimination of classification features of extracted ROIs is weak. However, neural networks can be used to map them to high-dimensional expressions, which can be easily distinguished. According to the experimental results, the best way for us to demonstrate the extraction of classification features is to fine-tune the calculation model of the pre-trained Swin-B on the taxonomic dataset, with the best accuracy of 94.34%. Moreover, the multi-head attention mechanism of the Transformer variants implements global and long-distance perception, which is different from the layer-by-layer expansion of CNN. The

TABLE 5 Performance of different models on test dataset.

Networks	Size (megabytes)	Time (ms/image)	All classes		Medusae	
			Precision (%)	Recall (%)	Precision (%)	Recall (%)
ShuffleNet V2 (93.13%)	1.3	273	89.23	87.91	100.00	89.72
ResNet50 (93.03%)	26	596	88.78	86.73	100.00	89.23
Swin-T (94.21%)	28	617	92.38	91.73	100.00	92.76
Swin-B (94.34%)	87	1343	93.37	92.85	100.00	93.87

Transformer variants require sufficient training data, and the performance of the Transformers was inferior to that of CNNs without transfer learning. However, the perception mechanisms of neural network to ordinary images and *in-situ* images are naturally similar; thus, the application of the pre-training network is equivalent to increasing the size of the dataset. Consequently, the Transformer variants can fully explore their potentials. To illustrate this inference, we used principal components analysis (PCA) to compress the output features from Swin-B before and after fine-tuning to two-dimensional representations, as shown in Figure 5. The network pre-trained by large ordinary image datasets exhibits a certain ability to distinguish the plankton targets. After transfer learning, it can further realize the feature clustering in small datasets and make each class region preserve sufficient feature distance. Therefore, the method we adopted has the potential to be applied in various specific scenarios.

Knowledge distillation is to transplant the extraction ability of classification features. Here, we discuss the characteristics of the proposed PPD method, classical knowledge distillation method, and traditional supervised learning. For the classification tasks, traditional supervised learning utilizes the cross-entropy loss to push the outputs close to the extreme values of 1 and 0. Whereas, the classical knowledge distillation methods attempt to learn the information of probability distribution output by the teacher networks and promote the student networks' perception of inter-class similarity. The proposed PPD method demonstrates the similarity calculation of classification features via interactions between an independent sample and a complete class. Our distillation mode combined intermediate feature learning with the generation of classification probabilities by using inter-class similarity. So, the gradient descent can simultaneously perform feature learning and supervised classification. The feature prototype extracted from the teacher network Swin-B and the sample features output from the teacher network were compressed into two-dimensional representation through PCA, and the results are shown in Figure 6. It can be seen that the feature prototypes are located in the centers of each cluster, which fully have the enough ability to express the features of each class. More importantly, some individual outlier features do not have obvious influences on the feature prototypes. Therefore, it can be seen that the average features as the characteristic prototypes are in line with the mathematical expectation. The interference from an outlier value is avoided and the damage of noise data to the classification performance is reduced. The proposed knowledge distillation method was tested through sufficient comparative experiments and obtained satisfactory results, and our novel method can be considered in wide range of applications.

4.2 Prospects for the development of *in-situ* monitoring

According to extensive experiments conducted above, our proposed methods have updated the algorithm pipeline and achieved satisfactory results on the test dataset. The lightweight neural networks can reach high accuracy and be appropriate to be deployed. The excellent effects and the practicability of Transformer variants and the proposed PPD method are verified in the plankton *in-situ* images.

The image processing for the current algorithm pipeline can be developed continuously. We are considering designing end-to-end deep learning object detection frameworks in our systems as many works have done in CV field. In addition, as the qualities of *in-situ* images are generally not ideal, it is necessary to build a large-scale plankton object detection dataset in the next period. Furthermore, unsupervised learning for plankton classification may be discussed and unlabeled data may be used to improve the representation ability of the models. In addition, the use of computer programs to assist in labeling and cleaning *in-situ* data are also expected to rapidly expand the database. For recognition tasks, compared with CNN in most recognition tasks, Transformer has not been saturated with the growth of network parameters and dataset size (Vaswani et al., 2017). Therefore, we still believe that with the continuous surge of underwater data, the Transformer will have a broader prospect in plankton monitoring applications. In terms of model compression, in addition to knowledge distillation, pruning is another kind of effective method. In recent years, researchers have

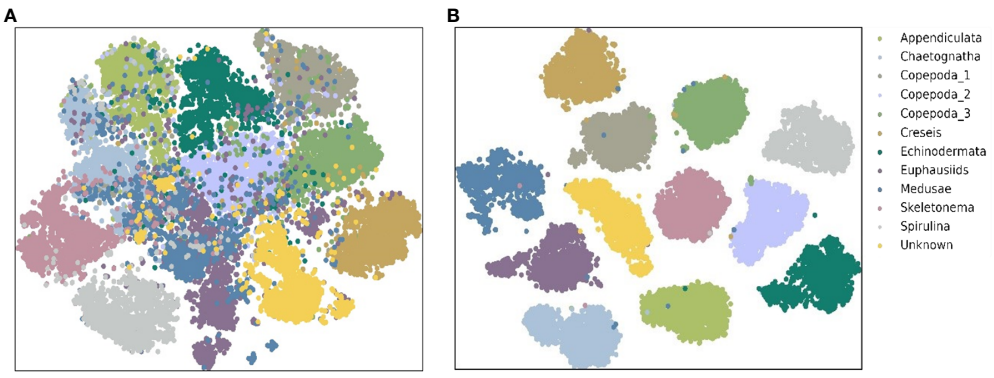


FIGURE 5
Visual evaluation of the ability to distinguish features before **(A)** and after **(B)** fine-tuning of the pre-training model.

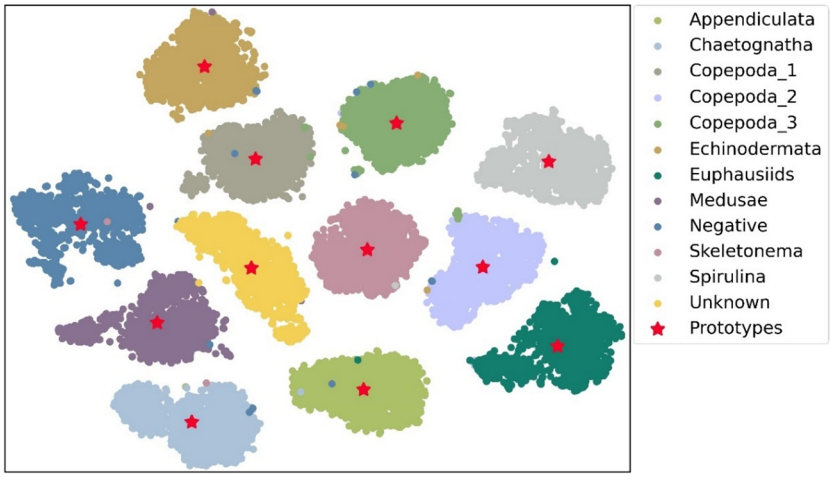


FIGURE 6
The visualization results of the feature prototype and the features of each sample output by the teacher network.

explored how to effectively combine the two methods, and related works have been carried out (Park and No, 2022; Liang et al., 2023), revealing the excellent effect that the combination schemes can bring.

In addition, the quality of dataset at the sensor side should be also focused on, especially the development of high-quality underwater optical imaging system. The adaptability of the imaging systems to the coastal, estuarine, and other complex water areas especially with high turbidity and water velocity need to be improved. A sincere suggestion is to introduce new hardware aids from the perspective of optical design, and the high quality of the source information will greatly reduce the difficulty of subsequent image processing.

4.3 Conclusions

This study proposed and demonstrated a novel knowledge distillation method and synchronously equipped new algorithm system for target detection and recognition regarding *in-situ* images of plankton. The experiments were based on the datasets captured by the experienced underwater imaging system PlanktonScope. Furthermore, the method expanded the analytical ability to gelatinous plankton, which has been a challenge till now, and achieved high recognition recall rate and short processing time. Especially, a new inter-class similarity distillation algorithm based on feature prototypes was proposed. For the first time, we used the similarity assessment of features among independent samples and complete classes as a regression task to realize knowledge distillation. Consequently, better performance was shown on the taxonomic dataset of plankton. Moreover, through experiments and comparisons with classical methods, we formed the final update of algorithm pipeline and discussed the work results and inner principle. The improvement of optical imaging and the exploration in image processing in the field of deep learning will be the two main focus points of future work.

Data availability statement

The data and the code used for algorithm implementation will be made available by the authors, without undue reservation.

Author contributions

JY, ZC, and YL completed the background investigation, method design, and experiments, and led the writing of the paper. KC provided materials of the original algorithm pipeline and participated in the comparative experiments. HB and XC provided valuable suggestions for the whole work and revised the paper. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by the National Key Research and Development Program of China (No. 2017YFC1403602) and the Shenzhen Science and Technology Innovation Program (Nos. KCXFZ20211020163557022, JSGG20191129110031632, JCYJ20170412171011187), and the National Natural Science Foundation of China (Nos. 61527826, 51735002), and the Major Scientific and Technological Innovation Project of the Shandong Provincial Key Research and Development Program (2019JZZY020708).

Acknowledgments

The authors express their sincere gratitude to the reviewers and editors who provided valuable comments and assistance for the publication.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Azani, N., Ghaffar, M., Suhaimi, H., Azra, M., Hassan, M., Jung, L., et al. (2021). "The impacts of climate change on plankton as live food: A review," in *IOP Conf. Ser.: Earth Environ. Sci.* (Virtual, Indonesia: IOP Science) 869(1), 012005. doi: 10.1088/1755-1315/869/1/012005

Benfield, M. C., Shaw, R. F., and Schwehm, C. J. (2000). Development of a vertically profiling, high-resolution, digital still camera system. *Louisiana State Univ. Baton Rouge Dept Oceanogr. Coast. Sci.* 2000. doi: 10.21236/ADA609777

Bhonsle, D., Chandra, V., and Sinha, G. R. (2012). "Medical image denoising using bilateral filter," in *Int. J. Image Graph. Sign. Proces* (MECS Publisher) 4(6). 36–43. doi: 10.5815/ijigsp.2012.06.06

Bi, H., Cook, S., Yu, H., Benfield, M. C., and Houde, E. D. (2013). Deployment of an imaging system to investigate fine-scale spatial distribution of early life stages of the ctenophore Mnemiopsis leidyi in Chesapeake Bay. *J. Plankton Res.* 35 (2), 270–280. doi: 10.1093/plankt/fbs094

Bi, H., Guo, Z., Benfield, M. C., Fan, C., Ford, M., Shahrestani, S., et al. (2015). A semi-automated image analysis procedure for in *situ* plankton imaging systems. *PloS One* 10 (5), e0127121. doi: 10.1371/journal.pone.0127121

Bi, H., Song, J., Zhao, J., Liu, H., Cheng, X., Wang, L., et al. (2022). Temporal characteristics of plankton indicators in coastal waters: High-frequency data from PlanktonScope. *J. Sea. Res.* 189, 102283. doi: 10.1016/j.seares.2022.102283

Braz, J. E. M., Dias, J. D., Bonecker, C. C., and Simoes, N. R. (2020). Oligotrophication affects the size structure and potential ecological interactions of planktonic microcrustaceans. *Aquat. Sci.* 82 (3), 1–10. doi: 10.1007/s00027-020-00733-z

Brun, P., Vogt, M., Payne, M. R., Gruber, N., O'brien, C. J., Buitenhuis, E. T., et al. (2015). Ecological niches of open ocean phytoplankton taxa. *Limnol. Oceanogr.* 60 (3), 1020–1038. doi: 10.1002/lno.10074

Buskey, E. J., and Hyatt, C. J. (2006). Use of the FlowCAM for semi-automated recognition and enumeration of red tide cells (Karenia brevis) in natural plankton samples. *Harmful Algae* 5 (6), 685–692. doi: 10.1016/j.hal.2006.02.003

Campbell, R. W., Roberts, P. L., and Jaffe, J. (2020). The Prince William Sound Plankton Camera: a profiling in *situ* observatory of plankton and particulates. *ICES J. Mar. Sci.* 77 (4), 1440–1455. doi: 10.1093/icesjms/fsaa029

Cowen, R. K., and Guigand, C. M. (2008). *In situ* ichthyoplankton imaging system (ISIIS): system design and preliminary results. *Limnol. Oceanogr.-Meth.* 6 (2), 126–132. doi: 10.4319/lom.2008.6.126

Davis, C. S., Gallager, S. M., Marra, M., and Stewart, W. K. (1996). Rapid visualization of plankton abundance and taxonomic composition using the Video Plankton Recorder. *Deep-Sea Res. Pt. II* 43 (7-8), 1947–1970. doi: 10.1016/S0967-0645(96)00051-3

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. doi: 10.48550/arxiv.2010.11929

Fan, A., Stock, P., Graham, B., Grave, E., Gribonval, R., Jegou, H., et al. (2020). Training with quantization noise for extreme model compression. *arXiv preprint arXiv*. doi: 10.48550/arXiv.2004.07320

Ferdous, R. H., Arifeen, M. M., Eiko, T. S., and Mamun, S. A. (2020). "Performance analysis of different loss function in face detection architectures," in *Proc. Int. Conf. Trends in Comput. Cognit. Eng.* 659–669. doi: 10.1007/978-981-33-4673-4_54

Gorsky, G., Picheral, M., and Stemmann, L. (2000). Use of the Underwater Video Profiler for the study of aggregate dynamics in the North Mediterranean. *Estuar. Coast. Shelf Sci.* 50 (1), 121–128. doi: 10.1006/ecss.1999.0539

Guo, B., Yu, J., Liu, H., Xu, W., Hou, R., and Zheng, B. (2018). Miniaturized in *situ* dark-field microscope for in *situ* detecting plankton. *Ocean Opt. Inf. Technol.* 10850, 243–250. doi: 10.1117/12.2505639

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* (Las Vegas, USA: IEEE), 770–778.

Hermand, J. P., Randall, J., Dubois, F., Queeckers, P., Yourassowsky, C., Roubaud, F., et al. (2013). "*In-situ* holography microscopy of plankton and particles over the continental shelf of Senegal," in *2013 Ocean Elec. (SYMPOL)*. (Kochi, India: IEEE), 1–10. doi: 10.1109/SYMPOL.2013.6701926

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. doi: 10.48550/arxiv.1503.02531

Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv* [Preprint] arXiv.1412.6980.

Kyathanahally, S. P., Hardeman, T., Merz, E., Bulas, T., Reyes, M., Isles, P., et al. (2021). Deep learning classification of lake zooplankton. *Front. Microbiol* 12. doi: 10.3389/fmicb.2021.746297

Kyathanahally, S. P., Hardeman, T., Reyes, M., Merz, E., Bulas, T., Brun, P., et al. (2022). Ensembles of data-efficient vision transformers as a new paradigm for automated classification in ecology. *Sci. Rep.* 12 (1), 18590. doi: 10.1038/s41598-022-21910-0

Li, X., and Cui, Z. (2016). Deep residual networks for plankton classification. *Oceans 2016 MTS/IEEE Monterey IEEE*, 1–4. doi: 10.1109/OCEANS.2016.7761223

Li, Y., Guo, J., Guo, X., Zhao, J., Yang, Y., Hu, Z., et al. (2021). Toward in *situ* zooplankton detection with a densely connected YOLOV3 model. *Appl. Ocean Res.* 114, 102783. doi: 10.1016/j.apor.2021.102783

Liang, C., Jiang, H., Li, Z., Tang, X., Yin, B., Zhao, T., et al (2023). HomoDistil: homotopic task-agnostic distillation of pre-trained transformers. *arXiv preprint arXiv: 2302.09632*. doi: 10.48550/arxiv.2302.09632

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., et al. (2021). "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. Proc. IEEE Int. Conf. Comput. Vis.* (Montreal, Canada: IEEE), 10012–10022.

Lumini, A., and Nanni, L. (2019). Deep learning and transfer learning features for plankton classification. *Ecol. Inform.* 51, 33–43. doi: 10.1016/j.ecoinf.2019.02.007

Luo, J. Y., Irisson, J. O., Graham, B., Guigand, C., Sarafraz, A., Mader, C., et al. (2018). Automated plankton image analysis using convolutional neural networks. *Limnol. Oceanogr.-Meth* 16 (12), 814–827. doi: 10.1002/lom3.10285

Lv, Z., Zhang, H., Liang, J., Zhao, T., Xu, Y., and Lei, Y. (2022). Microalgae removal technology for the cold source of nuclear power plant: A review. *Mar. pollut. Bull.* 183, 114087. doi: 10.1016/j.marpolbul.2022.114087

Ma, N., Zhang, X., Zheng, H.-T., and Sun, J. (2018). "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proc. Eur. Conf. Comput. Vis.* (Munich, Germany: Springer), 116–131.

Marini, S., Fanelli, E., Sbragaglia, V., Azzurro, E., Del Rio Fernandez, J., and Aguzzi, J. (2018). Tracking fish abundance by underwater image recognition. *Sci. Rep.* 8 (1), 1–12. doi: 10.1038/s41598-018-32089-8

Orenstein, E. C., and Beijbom, O. (2017). "Transfer learning and deep feature extraction for planktonic image data sets," in *IEEE Winter Conf. App. Comput. Vis.* (Santa Rosa, USA: IEEE), doi: 10.1109/WACV.2017.125

Orenstein, E. C., Kenitz, K. M., Roberts, P. L. D., Franks, P. J. S., Jaffe, J. S., and Barton, A. D. A. (2020). Semi-and fully supervised quantification techniques to improve population estimates from machine classifiers. *Limnol. Oceanogr.-Meth.* 18 (12), 739–753. doi: 10.1002/lom3.10399

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Tran. Syst. Man Cybern.* 9 (1), 62–66. doi: 10.1109/TSMC.1979.4310076

Pan, S. J., and Yang, Q. (2010). "A survey on transfer learning," in *IEEE Tran. Knowl. Data Eng.* (IEEE). Vol. 22. 1345–1359. doi: 10.1109/TKDE.2009.191

Park, J., and No, A. (2022). "Prune your model before distill it," in *Proc. Eur. Conf. Comput. Vis.* (Tel-Aviv, Israel: Springer), 120–136.

Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., et al. (2019). "Correlation congruence for knowledge distillation," in *Proc. IEEE Int. Conf. Comput. Vis.* (Seoul, South Korea: IEEE), 5007–5016.

Piredda, R., Tomasino, M. P., D'Erchia, A. M., Manzari, C., Pesole, G., Montresor, M., et al. (2017). Diversity and temporal patterns of planktonic protist assemblages at a Mediterranean Long Term Ecological Research site. *FEMS Microbiol. Ecol.* 93 (1). doi: 10.1093/femsec/fiw200

Ridnik, T., Ben-baruch, E., Noy, A., and Zelnik-manor, L. (2021). Imagenet-21k pretraining for the masses. doi: 10.48550/arxiv.2104.10972

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. (2014). Fitnets: Hints for thin deep nets. doi: 10.48550/arxiv.1412.6550

Said, K. A. M., Jambek, A. B., and Sulaiman, N. (2016). A study of image processing using morphological opening and closing processes. *Int. J. Control Theor. App.* 9 (31), 15–21.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*. (Salt Lake City, USA: IEEE), 4510–4520.

Song, J., Bi, H., Cai, Z., Cheng, X., He, Y., Benfield, M. C., et al. (2020). Early warning of Noctiluca scintillans blooms using *in-situ* plankton imaging system: an example from Dapeng Bay, PR China. *Ecol. Indic.* 112, 106123. doi: 10.1016/j.ecolind.2020.106123

Suzuki, S. (1985). Topological structural analysis of digitized binary images by border following. *Comput. Gr. Image Process.* 30 (1), 32–46. doi: 10.1016/0734-189X(85)90016-7

Tanaka, H., Kunin, D., Yamins, D. L. K., and Gnguli, S. (2020). Pruning neural networks without any data by iteratively conserving synaptic flow. *Proc. Adv. Neural Inf. Process. Syst.* 33, 6377–6389. doi: 10.48550/arXiv.2006.05467

Tomasi, C., and Manduchi, R. (1998). "Bilateral filtering for gray and color images," in *6th Int. Conf. Comput. Vis.* (Mumbai, India: IEEE). doi: 10.1109/ICCV.1998.710815

Tung, F., and Mori, G. (2019). "Similarity-preserving knowledge distillation," in *Proc. IEEE Int. Conf. Comput. Vis.* (Seoul, South Korea: IEEE), 1365–1374.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., et al. (2017). Attention is all you need. *Proc. Adv. Neural Inf. Process. Syst.* 30. doi: 10.48550/arXiv.1706.03762

Vincent, O. R., and Folorunso, O. (2009). "A descriptive algorithm for sobel image edge detection," in *Proc. Inf. Sci. IT Educ. Conf*, Vol. 40. 97–107.

Wang, Y., Liu, Y., Guo, H., Zhang, H., Li, D., Yao, Z., et al. (2022). Long-term nutrient variation trends and their potential impact on phytoplankton in the southern Yellow Sea, China. *Acta Oceanol. Sin.* 41 (6), 54–67. doi: 10.1007/s13131-022-2031-3

Wu, J., Wang, Y., Wu, Z., Veeraraghavan, A., and Lin, Y. (2018). Deep k-means: Re-training and parameter sharing with harder cluster assignments for compressing deep convolutions. doi: 10.48550/arXiv.1806.09228

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 25 September 2023
DOI 10.3389/fmars.2023.1158548

Check for updates

OPEN ACCESS

EDITED BY
Hongsheng Bi,
University of Maryland, United States

REVIEWED BY
Suja Cherukullapurath Mana,
Sathyabama Institute of Science and Technology, India
Katalin Blix,
UiT The Arctic University of Norway, Norway

*CORRESPONDENCE
Zhimin Wang
wangzhimin@ouc.edu.cn
Yongjian Gu
yjgu@ouc.edu.cn

†These authors have contributed equally to this work and share first authorship

RECEIVED 04 February 2023
ACCEPTED 11 September 2023
PUBLISHED 25 September 2023

CITATION
Shi S, Wang Z, Shang R, Li Y, Li J, Zhong G and Gu Y (2023) Hybrid quantum-classical convolutional neural network for phytoplankton classification. *Front. Mar. Sci.* 10:1158548. doi: 10.3389/fmars.2023.1158548

COPYRIGHT
© 2023 Shi, Wang, Shang, Li, Li, Zhong and Gu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Hybrid quantum-classical convolutional neural network for phytoplankton classification

Shangshang Shi†, Zhimin Wang*†, Ruimin Shang, Yanan Li, Jiaxin Li, Guoqiang Zhong and Yongjian Gu*

Faculty of Information Science and Engineering, Ocean University of China, Qingdao, China

The taxonomic composition and abundance of phytoplankton have a direct impact on marine ecosystem dynamics and global environment change. Phytoplankton classification is crucial for phytoplankton analysis, but it is challenging due to their large quantity and small size. Machine learning is the primary method for automatically performing phytoplankton image classification. As large-scale research on marine phytoplankton generates overwhelming amounts of data, more powerful computational resources are required for the success of machine learning methods. Recently, quantum machine learning has emerged as a potential solution for large-scale data processing by harnessing the exponentially computational power of quantum computers. Here, for the first time, we demonstrate the feasibility of using quantum deep neural networks for phytoplankton classification. Hybrid quantum-classical convolutional and residual neural networks are developed based on the classical architectures. These models strike a balance between the limited function of current quantum devices and the large size of phytoplankton images, making it possible to perform phytoplankton classification on near-term quantum computers. Our quantum models demonstrate superior performance compared to their classical counterparts, exhibiting faster convergence, higher classification accuracy and lower accuracy fluctuation. The present quantum models are versatile and can be applied to various tasks of image classification in the field of marine science.

KEYWORDS

hybrid quantum-classical neural network, quantum convolutional neural network, phytoplankton classification, parameterized quantum circuit, ansatz

1 Introduction

Phytoplankton is the most important primary producer in the aquatic ecosystem. As the main supplier of dissolved oxygen in the ocean, phytoplankton plays a vital role in the energy flow, material circulation and information transmission in the marine ecosystem (Barton et al., 2010; Gittings et al., 2018). The species composition and abundance of phytoplankton are key factors in marine ecosystem dynamics, exerting a direct influence on

global environment change. As such, much attention has been paid to the identification and classification of phytoplankton (Zheng et al., 2017; Pastore et al., 2020; Fuchs et al., 2022).

With the rapid development of imaging devices for phytoplankton (Owen et al., 2022), a huge number of phytoplankton images can now be collected in a short time. However, it has become impossible to classify and count these images using traditional manual methods, i.e. expert-based methods. To increase the efficiency of processing these images, machine learning methods has been introduced, including support vector machine (Hu and Davis, 2005; Sosik and Olson, 2007), random forest (Verikas et al., 2014; Faillettaz et al., 2016), k-nearest neighbor (Glüge et al., 2014), and artificial neural network (Mattei et al., 2018; Mattei and Scardi, 2020). In particular, convolutional neural network (CNN), which achieves state-of-the-art performance on image classification, has become widely used in this field in recent years. A variety of CNN-based architectures have been proposed to identify and classify phytoplankton with high efficiency and precision (Dai et al., 2017; Cui et al., 2018; Wang et al., 2018; Fuchs et al., 2022).

To conduct large-scale research on marine phytoplankton, more powerful computational resources are desired to ensure the success of machine learning methods for handling the overwhelmingly increasing volume of data. Along with the remarkable progress in the field of quantum computing (Arute et al., 2019; Zhong et al., 2020; Bharti et al., 2022; Madsen et al., 2022), quantum machine learning (QML) has emerged as a potential solution for large-scale data processing (Biamonte et al., 2017). There is a growing consensus that even the near-term NISQ (noisy intermediate-scale quantum) devices may find advantageous applications (Preskill, 2018), one of which is the quantum neural network (QNN) (Jeswal and Chakraverty, 2019; Kwak et al., 2021). The QNN takes the parameterized quantum circuit (PQC) as a learning model (Benedetti et al., 2019), and can be naturally extended to a quantum deep neural network with the flexible multilayer architecture. The quantum convolutional neural network (QCNN) is a typical model of quantum deep neural networks that has recently received a lot of attention and achieved significant developments. QCNN has demonstrated its success in processing both quantum and classical data, including quantum many-body problems (Cong et al., 2019), identification of high-energy physics events (Chen et al., 2022), COVID-19 prediction (Houssein et al., 2022) and MNIST dataset classification (Oh et al., 2020).

In this work, we explore the potential of QCNN for performing phytoplankton classification. There are two typical architectures of QCNN: the fully quantum parameterized QCNN (Cong et al., 2019) and the hybrid quantum-classical CNN (Liu et al., 2021). Due to the large size of phytoplankton images and the limited number of qubits and quantum operations available on current quantum devices, it is currently impractical to learn the images using fully quantum parameterized QCNN. Therefore, we adopt the hybrid quantum-classical convolutional neural network (QCCNN) architecture to achieve good multi-classification of the phytoplankton dataset. QCCNN integrates the PQC into the classical CNN architecture by replacing the classical feature map with the quantum feature map. This makes QCCNN friendly to current NISQ devices in terms of both the number of qubits and circuit depths, while retaining important features of classical CNN, such as nonlinearity and scalability (Liu et al., 2021).

Moreover, the QCCNN may face challenges such as the barren plateau problem (i.e. vanishing gradient) and degradation problem (i.e. saturated accuracy with increasing depth) (Deng, 2021). To address these issues, we further propose a hybrid quantum-classical residual network (QCResNet) that incorporates a residual architecture to enhance the QCCNN's performance.

It is worth noting that the visual transformer has recently achieved remarkable performance in image processing (Dosovitskiy et al., 2020) by identifying long-range dependencies and obtaining global information. Its success has led to its application in classifying plankton datasets (Baek et al., 2022; Dagtekin and Dethlefs, 2022; Kyathanahally et al., 2022; Shao et al., 2022). In the future, it will be intriguing to develop quantum visual transformer models based on the quantum self-attention mechanism (Li et al., 2022; Shi et al., 2023; Zhao et al., 2022), and explore their potential for phytoplankton classification.

The main contribution of this work is as follows:

(1) For the first time, the feasibility of using quantum deep neural networks for phytoplankton classification is demonstrated. This represents a concrete example of the application of quantum machine learning methods in the field of marine science.

(2) Several specific architectures for QCCNN and QCResNet are developed, which are accessible on near-term NISQ devices. Particularly, the QCResNet architecture is proposed to enhance the QCCNN's performance. These models are versatile and can be directly applied to other image classification tasks.

(3) The QCCNN and QCResNet models demonstrate exceptional performance in phytoplankton classification compared to template CNN and ResNet models. Moreover, the impact of PQC's expressibility and entangling capability on QCCNN's performance is explored.

The rest of the paper is organized as follows. Section 2 provides introduction to the preliminaries of QNN. In section 3, we discuss the architectures of QCCNN and QCResNet. Section 4 describes the phytoplankton dataset used in the experiment. Section 5 presents the experimental results, including the performance of QCCNN and QCResNet, as well as the impact of ansatz circuit on QCCNN's performance. Finally, conclusions are given in section 6.

2 Quantum neural network

QNN is a type of variational quantum algorithm, which is also the hybrid quantum-classical algorithm. Typically, QNN consists of four parts: data encoding, forward transformation performed by the ansatz, quantum measurement and parameter optimization routine, as illustrated in Figure 1. It's worth noting that the first three parts

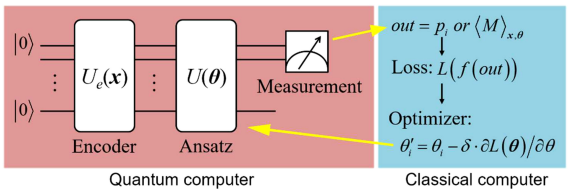


FIGURE 1
Architecture of QNN model. QNN is a hybrid quantum-classical algorithm. The forward transformation is implemented by the quantum computer, while the optimization of parameters is done by the classical computer.

are implemented on the quantum device, while the optimization routine is executed on the classical computer, which then feeds the updated parameters back into quantum device.

Data encoding is the process of embedding classical data into quantum states by applying a unitary transformation, i.e.$|\boldsymbol{x}\rangle = U_e|0\rangle^{\otimes n}$where $|\boldsymbol{x}\rangle$is proportional to the data vector $\boldsymbol{x}$. Data encoding can be regarded as a quantum feature map that maps the data space to the quantum Hilbert space (Schuld and Killoran, 2019). QNNs leverage this exponentially large Hilbert space as the feature space, making it extremely difficult to simulate using classical resources (Havlíček et al., 2017). One of the most commonly used encoding method in QNN is the angle encoding. It embeds classical data into the rotation angles of the quantum rotation gates. For example, given a normalized data vector $\boldsymbol{x} = (x_1, \ldots x_N)^T$ with $x_i \in [0, \quad 1)$, angle encoding can embed it into

$$R_y^{\otimes N}(\boldsymbol{x})|0\rangle^{\otimes N} = \bigotimes_{i=1}^{N}\left(\cos\frac{x_i}{2}|0\rangle + \sin\frac{x_i}{2}|1\rangle\right), \tag{1}$$

where R_y is the rotation gate about the $\hat{y}$ axes, i.e. $R_y(x_i) = [\cos\frac{x_i}{2}, \ -\sin\frac{x_i}{2}; \ \sin\frac{x_i}{2}, \ \cos\frac{x_i}{2}]$. For more information on data encoding strategies, please refer to (Hur et al., 2022).

The ansatz can be seen as a quantum analogue of feedforward neural network, which utilizes the quantum unitary transformation to implement the feature map of data. Essentially, the ansatz is a PQC with adjustable quantum gates. These adjustable parameters are optimized to approximate the target function that maps features into different domains representing different classes. Therefore, the structure of ansatz circuit plays a crucial role in specific learning tasks. In most cases, the hardware-efficient ansatz is adopted in QNN, which uses a limited set of quantum gates and a particular qubit connection topology that is specific to the quantum devices on hand. The gate set usually contains three single-qubit gates and one two-qubit gates. An arbitrary single-qubit gate can be expressed as a combination of rotation gates about the $\hat{x}$, and $\hat{z}$ axes. For example, using the X-Z decomposition, a single-qubit gate can be represented as

$$U_{1q}(\alpha, \beta, \gamma) = R_x(\alpha)R_z(\beta)R_x(\gamma), \tag{2}$$

where α, β, and γ are the rotation angles. The two-qubit gates are utilized to create entanglement between qubits. There are fixed two-qubit gates without adjustable parameters, such as the CNOT gate, and the ones with adjustable parameters, such as the controlled $R_x(\theta)$ and $R_z(\theta)$ gates. A comprehensive discussion of the properties of different ansatz circuits is presented in (Sim et al., 2019).

Quantum measurement produces an output value that can be used as a prediction for the data. The measurement operation corresponds to a Hermitian operator M, which can be decomposed as $M = \sum_i \lambda_i |i\rangle\langle i|$, where λ_i is the ith eigenvalue and $|i\rangle$ is the corresponding eigenvector. When a measurement is performed, the quantum state $|\psi\rangle$ will collapse to one of the eigenstates $|i\rangle$ with a probability $p_i = |\langle i|\psi\rangle|^2$. Then, the expectation value of the measurement outcome is

$$\langle M \rangle = \sum_i \lambda_i \cdot p_i = \sum_i \lambda_i |\langle i|\psi\rangle|^2. \tag{3}$$

The most fundamental measurement outcomes are the probabilities $\{p_i\}$ and the expectation $\langle M \rangle$. The commonly used measurement in quantum computing is the computational basis measurement, also known as the Pauli-Z measurement, with the Hermitian operator

$$\sigma_z = (+1)|0\rangle\langle 0| + (-1)|1\rangle\langle 1|. \tag{4}$$

When performing the σ_z measurement, a qubit will collapse to the state $|0\rangle$ ($|1\rangle$) with the probability. $p_0 = |\langle 0|\psi\rangle|^2$ ($p_1 = |\langle 1|\psi\rangle|^2$), and the corresponding eigenvalue is +1 (−1). The expectation value $\langle \sigma_z \rangle$ is a value within the range [-1, 1]. Due to the collapse principle of quantum measurement, in practice the probability and the expectation value are estimated using s samples of measurement, where s is known as the number of shots.

Optimization routine is used to update the parameters of the ansatz circuit. These parameters correspond to the adjustable rotation angles of gates and are updated based on the data. Optimizing the parameters $\boldsymbol{\theta}$ is in fact the process of minimizing the loss function $L(\boldsymbol{\theta})$. Similar to classical models, QNN can use various loss functions such as mean squared error loss and cross-entropy loss. For example, the multi-category cross-entropy loss can be expressed as

$$L(\theta) = -\frac{1}{N}\sum_{j=1}^{N}\sum_{c=1}^{C}[y_{jc} \cdot f(p_{i=c})]. \tag{5}$$

In this equation, N is the batch size; C is the number of categories; $y_{jc} \in \{0, 1\}$ is the class label; $p_{i=c}$ is the probability of measuring the eigenstates $|i\rangle$ corresponding to the category c; and $f(\cdot)$ represents the post-processing of the measurement outcome, which is used to associate the outcome to the label y_{jc}.

Similar to classical neural networks, the parameters in QNN can be updated based on the gradient of the loss function. For instance, the gradient descent method can be used to update the ith parameter θ_i as follows:

$$\theta_i' = \theta_i - \delta \cdot \partial L(\boldsymbol{\theta})/\partial \boldsymbol{\theta}_i, \tag{6}$$

where δ is the learning rate. In quantum computing, there is no backpropagation algorithm to directly calculate the gradient of the loss function. Instead, derivatives are typically evaluated using the difference method or the parameter shift rule on the quantum devices (Wierichs et al., 2022).

3 Methods

3.1 Quantum-classical convolutional neural network

The QCCNN can be constructed based on classical CNN models. Specifically, using the CNN architecture presented in the supplementary material (Supplementary Figure 1) as a template, the QCCNN can be designed by implementing the convolutional layers with PQC. Figure 2 shows two possible QCCNN architectures. In Figure 2A, the QCCNN consists of one quantum convolutional layer and one classical convolutional layer, and Figure 2B shows a QCCNN with two quantum convolutional layers.

The models in Figure 2A and Figure 2B are named QCCNN-1 and QCCNN-2, respectively. Below, we delve into the details of the two architectures.

3.1.1 Quantum convolutional layer

The architecture of the quantum convolution layer #1 and quantum convolution layer #2 used in Figure 2 is illustrated in Figures 3A, B respectively. They consist of similar components as QNN, including the encoding circuit, ansatz circuit and quantum measurement.

In the quantum convolution layer #1, the filter window size is set to 2×2, and the four elements are embedded using four qubits through four $R_y(\theta)$ gates; while in the quantum convolution layer #2, the filter window size is set to 3×3, and the nine elements are embedded using nine qubits through nine $R_y(\theta)$ rotation gates. The ansatz is implemented using two typical hardware-efficient circuits, as shown in Figure 4. Figure 4A depicts the all-to-all configuration of two-qubit gates, which has the larger expressibility and entangling capability but the higher circuit complexity, while Figure 4B depicts the circuit-block configuration of two-qubit gates, which has the smaller expressibility and entangling capability but the lower circuit complexity (Sim et al., 2019). The expressibility and entangling capability of the ansatz can be increased by stacking the circuit as multi layers.

Expressibility and entangling capability are two key characteristics that describe the representative capability of a PQC in the exponentially large Hilbert space (Sim et al., 2019). It's

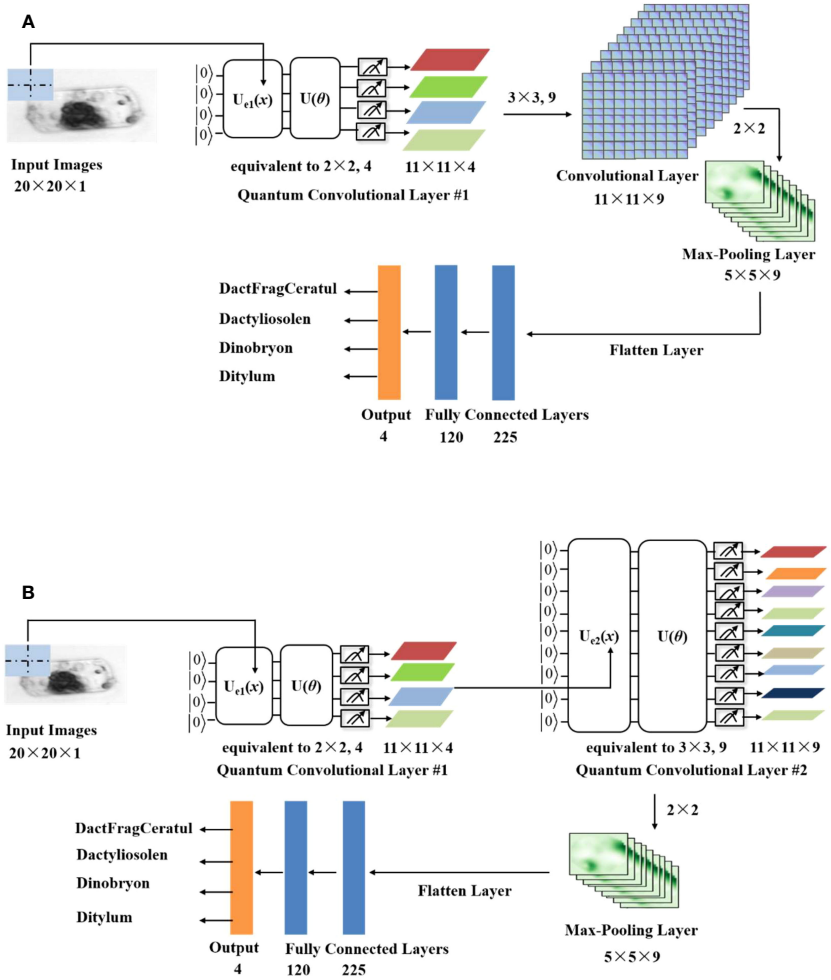


FIGURE 2
Architecture of the QCCNN with **(A)** one quantum and one classical convolutional layer (named QCCNN-1) and **(B)** two quantum convolutional layers (named QCCNN-2).

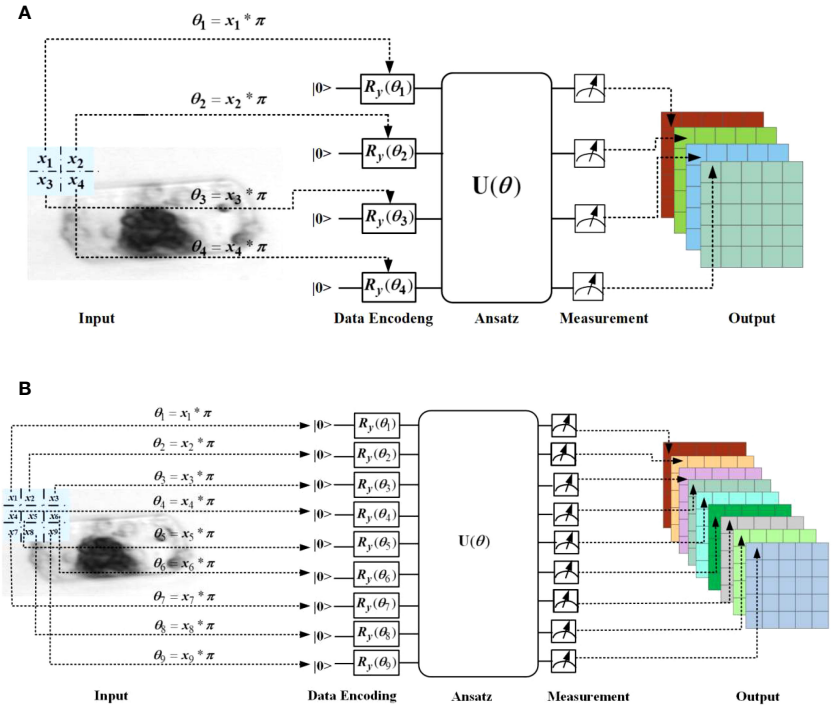


FIGURE 3
Architecture of the quantum convolutional layer #1 **(A)** and #2 **(B)** used in Figure 2.

important to note that the Hilbert space serves as the feature space of QCCNN, which means that the difference in the representative capability of the ansatz circuit can significantly affect the performance of QCCNN. However, the specific impact of this difference remains ambiguous. In the experiment section, we explore this impact in more detail.

For the quantum measurement in Figure 3, the four (nine) qubits are measured individually using the σ_z operator. The

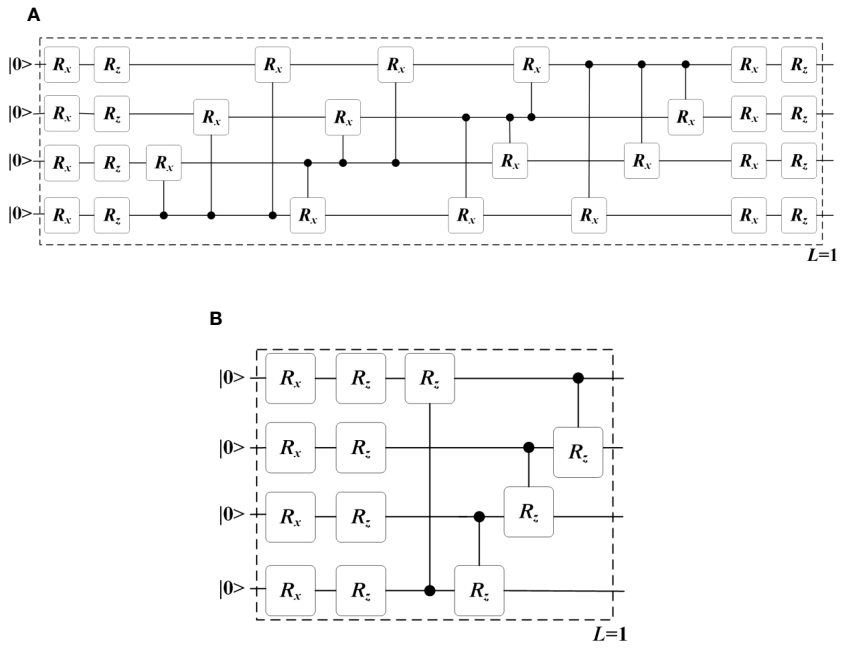


FIGURE 4
Two typical ansatz circuits with **(A)** all-to-all configuration and **(B)** circuit-block configuration of two-qubit gates. These circuits are used as a single layer, i.e. $L = 1$. Multiple layers can be stacked to increase the expressibility and entangling capability of the circuit.

resulting probabilities of each qubit collapsing to state $|0\rangle$ are then used as four (nine) feature channels for the next layer. It's worth noting that the quantum convolutional layer does not have an activation function, and the nonlinearity arises from the process of data encoding and quantum measurement. This is a significant difference between QNN and classical models.

3.1.2 Classical operations

The classical operations of QCCNN include classical convolutional layers, pooling layers, and fully connected layers, which follow the typical operations of CNN. Specifically, in the convolutional layers, a window size of 3×3 is used, and the activation function is the ReLu function. A Max Pooling layer is employed to reduce the number of trainable parameters. Finally, at the end of QCCNN, two fully connected layers are used to connect the convolutional and output layer.

3.2 Quantum residual network

Similar to the method used to design QCCNN, QCResNet can be constructed based on the template ResNet presented in the supplementary material (Supplementary Figure 2). Figure 5 illustrates two possible architectures for QCResNet. In Figure 5A, the QCResNet consists of one quantum residual unit and one classical residual unit, while Figure 5B has two quantum residual units. The two models are named QCResNet-1 and QCResNet-2, respectively.

As shown in Figure 5, both quantum residual unit #1 and quantum residual unit #2 utilize one quantum convolutional layer. It is worth noting that the quantum convolutional layer in quantum residual unit #1 uses a filter window size of 3×3, but outputs three feature channels, which differs from the one shown in Figure 3B. The architecture of the quantum convolutional layer used in quantum residual unit #1is presented in the supplementary material (Supplementary Figure 3). On the other hand, the quantum convolutional layer used in quantum residual unit #2 is identical to the one shown in Figure 3B.

4 Datasets and networks

The image dataset of phytoplankton used in this work was obtained by analyzing water from Woods Hole Harbor using a custom-built imaging-in-flow cytometer (Sosik and Olson, 2007). Sampling was conducted between late fall and early spring in 2004

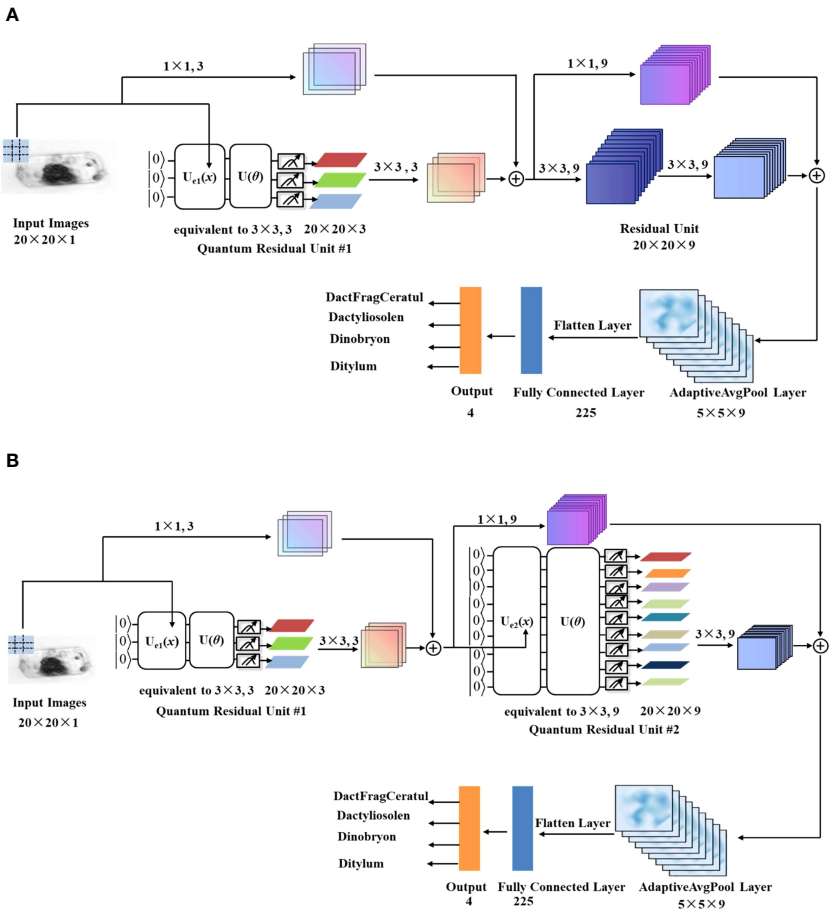


FIGURE 5
Architecture of the QCResNet with **(A)** one quantum residual unit (named QCResNet-1) and **(B)** two quantum residual units (named QCResNet-2).

and 2005. The dataset consists of 6600 images that were visually inspected and manually identified, with an even distribution across 22 categories, resulting in 300 images per category. Example images of the 22 categories are shown in Figure 6. All images were randomly divided into training set and test set, with each set containing 150 images for each category. This results in a balanced distribution of images across the categories.

In the experiment, the QCCNN and QCResNet were simulated on the classical computer, which required significant computational resources. As a result, it was not practical to train our quantum models using the full dataset of 6600 images. To address this issue, we compiled a sub-dataset consisting of 1200 images across four categories of phytoplankton, which are *DactFragCeratul*, *Dactyliosolen*, *Dinobryon* and *Ditylum*. In addition, to make the images accessible to the QCCNN and QCResNet models, all images are resized to 20×20 pixels. It is important to note that these limitations are only due to the difficulty of simulating the quantum circuit with a large number of qubits on the classical computer. The dataset used in the experiments is available on GitHub (Shi, 2023).

In the experiment, six neural networks are evaluated using the phytoplankton dataset. These networks include the template CNN (Supplementary Figure 1), template ResNet (Supplementary Figure 2), QCCNN-1 and QCCNN-2 (Figure 2), QCResNet-1 and QCResNet-2 (Figure 5). The specific architectures of these models are discussed in Section 3. A detailed comparison of their parameters is presented in the Supplementary Material (Section 3). In general, the quantum convolutional layer uses fewer parameters than the classical models, resulting in the faster convergence of the quantum models, as demonstrated in the following experiments.

In this work, the quantum and classical neural networks are implemented using the PennyLane software (Bergholm et al., 2018) and Pytorch framework, respectively. PyTorch-compatible quantum nodes in PennyLane are used to construct the hybrid quantum-classical neural networks. The loss function used is the cross-entropy function, as shown in Eq. (5). The parameters in the quantum and classical layers are trained together and updated based on the SGD method. The number of shots used in the quantum measurement is set to 1500, as discussed in the Supplementary Material (Section 4). The six neural networks have learning rates ranging from 0.05 to 0.1, with a batch size of 15 and trained for 50 epochs each.

5 Experimental results and discussions

5.1 Training loss and classification accuracy

To compare the performance of classical and quantum models, we first analyze the models' training loss and classification accuracy. Figure 7 displays the curves of the training loss and test classification accuracy of the template CNN and QCCNN models. It is clear from the curves that the QCCNN model converges much faster than the CNN model. Furthermore, the classification accuracy of QCCNN-1 is 93.67%, which is almost the same as that of CNN. However, the accuracy fluctuation of QCCNN is much smaller, indicating that QCCNN has better generalization.

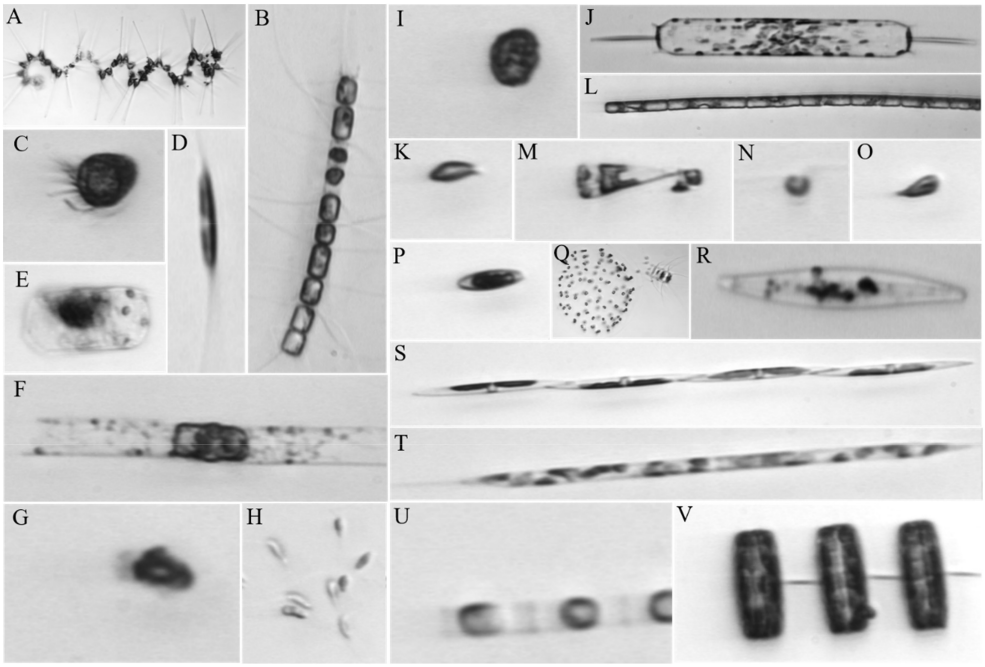


FIGURE 6
Example images of the 22 categories of phytoplankton: **(A)**, Asterionellopsis; **(B)**, Chaetoceros; **(C)**, Ciliate; **(D)**, Cylindrotheca; **(E)**, DactFragCeratul; **(F)**, Dactyliosolen; **(G)**, Detritus; **(H)**, Dinobryon; **(I)**, Dinoflagellate; **(J)**, Ditylum; **(K)**, Euglena; **(L)**, Guinardia; **(M)**, Licmophora; **(N)**, Nanoflagellate; **(O)**, other cells< 20μm; **(P)**, Pennate; **(Q)**, Phaeocystis; **(R)**, Pleurosigma; **(S)**, Pseudonitzschia; **(T)**, Rhizosolenia; **(U)**, Skeletonema; **(V)**, Thalassiosira.

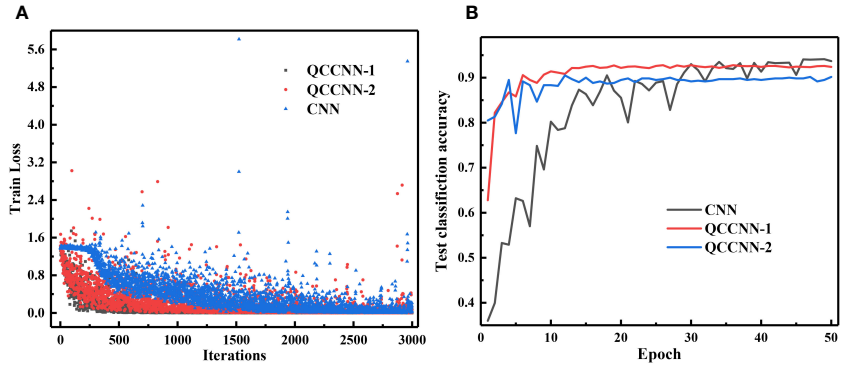


FIGURE 7
Curves of the training loss **(A)** and test classification accuracy **(B)** of CNN and QCCNN for phytoplankton classification.

The stronger performance of QCCNN can be attributed to the unique feature space of QCCNN, that is, the exponentially large Hilbert space created by the quantum circuit. This quantum feature space enables QCCNN to capture more abstract information from the data and generalize better. As the number of qubits and depth of quantum circuit increase, the quantum feature space will become completely intractable for classical computers, leading to a quantum advantage for QCCNN.

In addition, it's interesting to note that the accuracy of QCCNN-1 is higher than that of QCCNN-2. The experiments show that adding more quantum convolutional layers to QCCNN does not necessarily improve the model's performance. This is likely because more quantum convolutional layers significantly increase the feature space, making it more difficult to train the model. Therefore, the number and position of quantum convolutional layers used in QCCNN should be optimized for the specific learning tasks. Similar results have also been observed in the quantum-inspired CNN (Shi et al., 2022).

The curves for the training loss and test classification accuracy of the template ResNet and QCResNet models are shown in Figure 8. Similar to the findings for QCCNN, QCResNet exhibits similar features. QCResNet converges faster than ResNet; the classification accuracy of QCResNet-1 is 94.5%, which is higher than ResNet's 91.5%; QCResNet shows much smaller fluctuations in accuracy compared to ResNet. The larger fluctuations in the training loss and accuracy curves of ResNet, compared to CNN, can be reduced by increasing the depth of the networks. Additionally, the performance of QCResNet-1 is better than that of QCResNet-2, indicating that the number and position of quantum convolutional layers used in QCResNet should be optimized for the specific learning tasks, as it is for QCCNN.

5.2 Confusion matrix and other evaluation metric

In order to conduct a more comprehensive evaluation of the model's classification performance, we compute the confusion matrices of the results obtained by the six neural networks, as shown in Figure 9. A confusion matrix is an $N \times N$ matrix, where N represents the number of target categories. It summarizes the correct and incorrect predictions generated by the models on the multiple-class classification task.

Furthermore, based on the confusion matrices, we calculate additional evaluation metrics, in addition to classification accuracy, to analyze the generalization ability of the six neural networks.

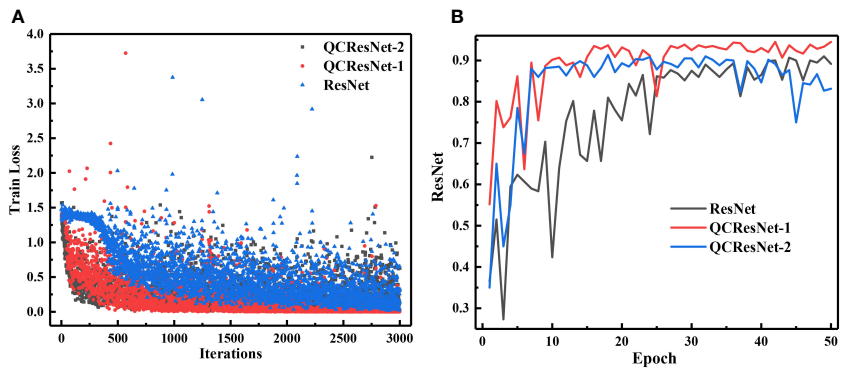


FIGURE 8
Curves of the training loss **(A)** and test classification accuracy **(B)** of ResNet and QCResNet for phytoplankton classification.

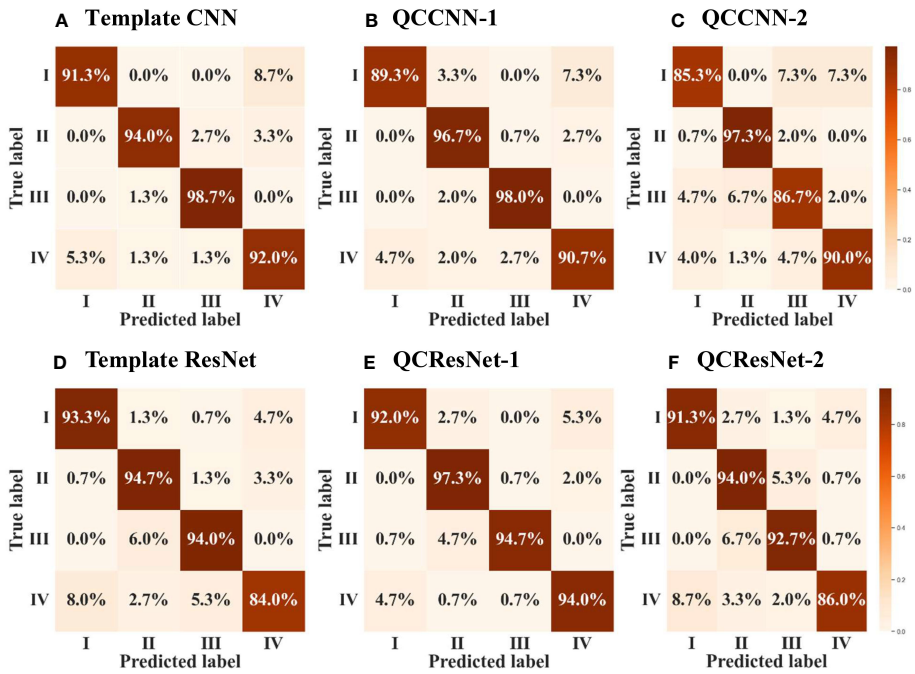


FIGURE 9
Confusion matrix of the results obtained by the six neural networks, namely **(A)** template CNN, **(B)** QCCNN-1, **(C)** QCCNN-2, **(D)** template ResNet, **(E)** QCResNet-1 and **(F)** QCResNet-2. The classes I, II, III and IV represent DactFragCeratul, Dactyliosolen, Dinobryon and Detritus, respectively.

These metrics include precision, recall, F1-score, specificity, false positive rate (FPR), false discovery rate (FDR) and false negative rate (FNR). The definitions of these metrics can be found in the Supplementary Material (Section 5). Table 1 presents the results, where the metrics are computed as the arithmetic mean of the metric values for each class, namely the macro metric, as illustrated in the supplementary.

Now we can analyze the performance of the six neural networks in greater detail, based on the confusion matrix in Figure 9 and the evaluation metrics presented in Table 1. Firstly, QCResNet-1 outperforms the other models in all evaluation metrics, indicating that the use of a residual architecture effectively enhances the performance of QCCNN. In particular, when compared to ResNet, QCResNet-1 exhibits significantly stronger performance on type IV phytoplankton (i.e. Detritus), as shown in Figure 9. However, QCResNet-1 performs poorly on type III phytoplankton (i.e. Dinobryon) when compared to the CNN and QCCNN-1 models. In general, QCResNet-1, QCCNN-1 and CNN models achieve comparable performance in terms of evaluation metrics; however, their classification outcomes differ significantly across the four phytoplankton categories.

Secondly, the QCCNN-2, QCResNet-2 and ResNet models exhibit poor performance, which is consistent with the results shown in Figures 7 , 8. As per Figure 9, the primary weakness of QCCNN-2 is its poor performance on type III phytoplankton, with a prediction accuracy that is approximately 10 percentage points lower. On the other hand, QCResNet-2 performs poorly on type IV phytoplankton. In future work, it would be interesting to compare

TABLE 1 Results of the eight evaluation metrics for the six neural networks.

Metrics	CNN	QCCNN-1	QCCNN-2	ResNet	QCResNet-1	QCResNet-2
Accuracy	0.94	0.9367	0.8983	0.915	0.945	0.91
Precision	0.9407	0.9368	0.8981	0.915	0.9458	0.9110
Recall	0.94	0.9367	0.8983	0.915	0.945	0.91
F1-Score	0.9403	0.9367	0.8982	0.915	0.9454	0.9105
Specificity	0.98	0.9789	0.9661	0.9717	0.9817	0.97
FPR	0.02	0.0211	0.0339	0.0283	0.0183	0.03
FDR	0.0593	0.0631	0.1019	0.085	0.0557	0.0891
FNR	0.06	0.0633	0.1017	0.085	0.055	0.09

the performance of these models on other datasets and learning tasks.

5.3 Influence of ansatz circuit for QCCNN

The quantum convolutional layer is a crucial element of both QCCNN and QCResNet. Its primary function is to utilize the ansatz circuit, i.e. a PQC, as a filer to perform the forward transformation in CNN. Therefore, the features of the ansatz circuit have a significant impact on the performance of QCCNN and QCResNet. Analyzing this relationship can help improve the performance of QNN models.

The ansatz circuit can be quantitatively characterized by its expressibility and entangling capability (Sim et al., 2019). Expressibility refers to the ability of a circuit to generate states that are highly representative of the Hilbert space. One way to calculate expressibility is by comparing the distribution of states generated by sampling the PQC's parameters to the uniform distribution of states in the Haar-random state ensemble. On the other hand, entangling capability describes the correlation between multiple qubits, that is, the inherent correlation within the quantum state. The entangling capability of an ansatz circuit can be quantified using the entanglement measures, such as the Meyer-Wallach measure. Generating highly entangled states with low-depth circuits can provide significant advantages for QNN, such as the ability to capture non-trivial correlations in quantum data.

There should be a relationship between the expressibility and entangling capability of the ansatz circuit and the performance of the corresponding QNNs. Below, we use QCCNN-1 as the basic model to exploit this dependence. Note that in the experiments discussed above, QCCNN-1 uses the circuit shown in Figure 4A as its ansatz. As mentioned in Section 3.1.1, Figure 4A circuit has higher expressibility and entangling capability, while Figure 4B circuit is lower but can be stacked to increase its expressibility and entangling capability. By replacing the ansatz of QCCNN-1 with multiple layers of Figure 4B circuit, we obtain five versions of QCCNN-1.

The classification accuracy of the five versions of QCCNN-1 is shown in Figure 10. The figure shows that QCCNN-1 using Figure 4A circuit as the ansatz achieves higher accuracy compared to that using Figure 4B. This suggests that higher expressibility and entangling capability of the ansatz circuit can indeed result in better performance of the QCCNN model.

However, for QCCNN-1 using multi-layers of Figure 4B as the ansatz, the accuracy does not always increase with the number of layers. Specifically, the accuracy of QCCNN-1 with 2 layers is the highest, while those with 1, 3 and 4 layers are close. Note that the circuit using 4 layers of Figure 4B achieves similar expressibility and entangling capability as that of Figure 4A, as presented in (Sim et al., 2019). Therefore, this suggests that in addition to the properties of expressibility and entangling capability, there are other influential factors on the models' performance.

One such factor is the number of trainable parameters. When the number of layers is increased, the expressibility and entangling capability increase, but so does the number of trainable parameters.

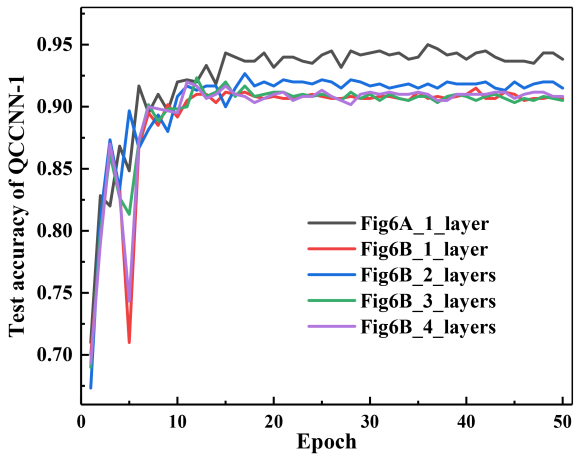


FIGURE 10
Classification accuracy of the five versions of QCCNN-1, which utilize the circuit of Figure 4A and 1, 2, 3 and 4 layers of Figure 4B as the ansatz.

More parameters make the model more difficult to train, which can decrease its generalization and offset the positive effect of increasing the expressibility and entangling capability. Another factor is the topological structure of the ansatz circuit. A quantitative method for characterizing the architecture of PQC and its correlation to the performance of the corresponding QCCNN need to be exploited in detail. We leave this for future work.

6 Conclusion

In this work, we develop several hybrid quantum-classical convolutional and residual neural networks and demonstrate their efficiency for phytoplankton classification. The QCCNN and QCResNet models are constructed by incorporating quantum-enhanced forward transformations into classical CNN and ResNet models. These hybrid architectures strike a good balance between the limited functionality of current NISQ devices and the large-size images of phytoplankton.

QCResNet outperforms classical models in terms of prediction performance, while QCCNN performs comparably to its classical counterparts. More remarkably, both QCCNN and QCResNet exhibit much faster convergence and more stable classification accuracy curves, with less fluctuation. We also find that the performance of QCCNN and QCResNet depends on several factors, including the expressibility, entangling capability and topological structure of the ansatz circuit, as well as the number of training parameters. By considering all these factors, the model's performance can be improved. Our QCCNN and QCResNet models are versatile and can be easily expanded for other image classification tasks.

In the future, we plan to optimize the architecture of QCCNN and QCResNet from both quantum and classical perspectives. This includes optimizing the structure of quantum convolutional layer and the template CNN and ResNet architecture. Additionally, due to computational resources limitations, in this work we construct a

mini model of QCNN and evaluate its performance using a relatively small dataset. In the future, it will be necessary to demonstrate the scalability of our models and find practical and advantageous applications in more marine science tasks.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

SS and ZW developed the algorithms and wrote the first draft. SS, RS, YL and JL wrote the codes and carried out the numerical experiments. YG, GZ and ZW planned and designed the project. All authors discussed the results and reviewed the manuscript. All authors contributed to the article and approved the submitted version

Funding

The present work is supported by the Natural Science Foundation of Shandong Province of China (ZR2021ZD19) and the National Natural Science Foundation of China (12005212).

Acknowledgments

We are grateful to the support of computational resources from the Marine Big Data Center of Institute for Advanced Ocean Study of Ocean University of China.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fmars.2023.1158548/full#supplementary-material

References

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., et al. (2019). Quantum supremacy using a programmable superconducting processor. *Nature* 574 (7779), 505–510. doi: 10.1038/s41586-019-1666-5

Baek, S. S., Jung, E. Y., Pyo, J., Pachepsky, Y., Son, H., and Cho, K. H. (2022). Hierarchical deep learning model to simulate phytoplankton at phylum/class and genus levels and zooplankton at the genus level. *Water Res.* 218, 118494. doi: 10.1016/j.watres.2022.118494

Barton, A. D., Dutkiewicz, S., Flierl, G., Bragg, J., and Follows, M. J. (2010). Patterns of diversity in marine phytoplankton. *Science* 327 (5972), 1509–1511. doi: 10.1126/science.1184961

Benedetti, M., Lloyd, E., Sack, S., and Fiorentini, M. (2019). Parameterized quantum circuits as machine learning models. *Quantum Sci. Technol.* 4 (4), 043001. doi: 10.1088/2058-9565/ab4eb5

Bergholm, V., Izaac, J., Schuld, M., Gogolin, C., Ahmed, S., Ajith, V., et al. (2018). Pennylane: Automatic differentiation of hybrid quantum-classical computations. arXiv:1811.04968. doi: 10.48550/arXiv.1811.04968

Bharti, K., Cervera-Lierta, A., Kyaw, T. H., Haug, T., Alperin-Lea, S., Anand, A., et al. (2022). Noisy intermediate-scale quantum algorithms. *Rev. Modern Phys.* 94 (1), 15004. doi: 10.1103/RevModPhys.94.015004

Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., and Lloyd, S. (2017). Quantum machine learning. *Nature* 549, 195. doi: 10.1038/nature23474

Chen, S. Y. C., Wei, T. C., Zhang, C., Yu, H., and Yoo, S. (2022). Quantum convolutional neural networks for high energy physics data analysis. *Phys. Rev. Res.* 4 (1), 13231. doi: 10.1103/PhysRevResearch.4.013231

Cong, I., Choi, S., and Lukin, M. D. (2019). Quantum convolutional neural networks. *Nat. Physics.* 15 (12), 1273–1278. doi: 10.1038/s41567-019-0648-8

Cui, J., Wei, B., Wang, C., Yu, Z., Zheng, H., Zheng, B., et al. (2018). "Texture and shape information fusion of convolutional neural network for plankton image classification," in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, Kobe, Japan. 1–5. doi: 10.1109/OCEANSKOBE.2018.8559156

Dagtekin, O., and Dethlefs, N. (2022). "Modelling phytoplankton behaviour in the north and irish sea with transformer networks," in *Proceedings of the Northern Lights Deep Learning Workshop*, Vol. 3. doi: 10.7557/18.6229

Dai, J., Yu, Z., Zheng, H., Zheng, B., and Wang, N. (2017). "A hybrid convolutional neural network for plankton classification," in *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops*, Vol. 102-114. doi: 10.1007/978-3-319-54526-4_8

Deng, D. (2021). Quantum enhanced convolutional neural networks for NISQ computers. *Sci. China Phys. Mech. Astron.* 64 (10), 100331. doi: 10.1007/s11433-021-1758-0

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929.. doi: 10.48550/arXiv.2010.11929

Faillettaz, R., Picheral, M., Luo, J. Y., Guigand, C., Cowen, R. K., and Irisson, J. O. (2016). Imperfect automatic image classification successfully describes plankton distribution patterns. *Methods Oceanogr.* 15, 60–77. doi: 10.1016/j.mio.2016.04.003

Fuchs, R., Thyssen, M., Creach, V., Dugenne, M., Izard, L., Latimier, M., et al. (2022). Automatic recognition of flow cytometric phytoplankton functional groups using convolutional neural networks. *Limnol. Oceanogr. Methods*. 20 (7), 387–399. doi: 10.1002/lom3.10493

Gittings, J. A., Raitsos, D. E., Krokos, G., and Hoteit, I. (2018). Impacts of warming on phytoplankton abundance and phenology in a typical tropical marine ecosystem. *Sci. Rep.* 8 (1), 1–12. doi: 10.1038/s41598-018-20560-5

Glüge, S., Pomati, F., Albert, C., Kauf, P., and Ott, T. (2014). "The challange of clustering flow cytometry data from phytoplankton in lakes," in *Nonlinear dynamics of electronic systems. NDES 2014. Communications in computer and information science*, vol. 438 . Eds. V. M. Mladenov and P. C. Ivanov (Cham: Springer). doi: 10.1007/978-3-319-08672-9_45

Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., et al. (2017). Supervised learning with quantum-enhanced feature spaces. *Nature* 567, 209. doi: 10.1038/s41586-019-0980-2

Houssein, E. H., Abohashima, Z., Elhoseny, M., and Mohamed, W. M. (2022). Hybrid quantum-classical convolutional neural network model for COVID-19 prediction using chest X-ray images. *J. Comput. Design Eng.* 9 (2), 343–363. doi: 10.1093/jcde/qwac003

Hu, Q., and Davis, C. (2005). Automatic plankton image recognition with co-occurrence matrices and support vector machine. *Mar. Ecol. Prog. Series.* 295, 21–31. doi: 10.3354/meps295021

Hur, T., Kim, L., and Park, D. K. (2022). Quantum convolutional neural network for classical data classification. *Quantum Mach. Intell.* 4 (1), 1–18. doi: 10.1007/s42484-021-00061-x

Jeswal, S. K., and Chakraverty, S. (2019). Recent developments and applications in quantum neural network: a review. *Arch. Comput. Methods Eng.* 26 (4), 793–807. doi: 10.1007/s11831-018-9269-0

Kwak, Y., Yun, W. J., Jung, S., and Kim, J. (2021). "Quantum neural networks: Concepts, applications, and challenges," in *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN)*. 413–416. doi: 10.1109/ICUFN49451.2021.9528698

Kyathanahally, S. P., Hardeman, T., Reyes, M., Merz, E., Bulas, T., Brun, P., et al. (2022). Ensembles of data-efficient vision transformers as a new paradigm for automated classification in ecology. *Sci. Rep.* 12 (1), 18590. doi: 10.1038/s41598-022-21910-0

Li, G., Zhao, X., and Wang, X. (2022). Quantum self-attention neural networks for text classification. arXiv:2205.05625. doi: 10.48550/arXiv.2205.05625

Liu, J., Lim, K. H., Wood, K. L., Huang, W., Guo, C., and Huang, H. L. (2021). Hybrid quantum-classical convolutional neural networks. *Sci. China Phys. Mech. Astron.* 64 (9), 1–8. doi: 10.1007/s11433-021-1734-3

Madsen, L. S., Laudenbach, F., Askarani, M. F., Rortais, F., Vincent, T., Bulmer, J. F. F., et al. (2022). Quantum computational advantage with a programmable photonic processor. *Nature* 606, 75–81. doi: 10.1038/s41586-022-04725-x

Mattei, F., Franceschini, S., and Scardi, M. (2018). A depth-resolved artificial neural network model of marine phytoplankton primary production. *Ecol. Model.* 382, 51–62. doi: 10.1016/j.ecolmodel.2018.05.003

Mattei, F., and Scardi, M. (2020). Embedding ecological knowledge into artificial neural network training: A marine phytoplankton primary production model case study. *Ecol. Model.* 421, 108985. doi: 10.1016/j.ecolmodel.2020.108985

Oh, S., Choi, J., and Kim, J. (2020). "A tutorial on quantum convolutional neural networks (QCNN)," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. 236–239. doi: 10.1109/IDAACS53288.2021.9661011

Owen, B. M., Hallett, C. S., Cosgrove, J. J., Tweedley, J. R., and Moheimani, N. R. (2022). Reporting of methods for automated devices: A systematic review and recommendation for studies using FlowCam for phytoplankton. *Limnol. Oceanogr. Methods* 20 (7), 400–427. doi: 10.1002/lom3.10496

Pastore, V. P., Zimmerman, T. G., Biswas, S. K., and Bianco, S. (2020). Annotation-free learning of plankton for classification and anomaly detection. *Sci. Rep.* 10 (1), 1–15. doi: 10.1038/s41598-020-68662-3

Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum* 2, 79. doi: 10.22331/q-2018-08-06-79

Schuld, M., and Killoran, N. (2019). Quantum machine learning in feature Hilbert spaces. *Phys. Rev. Lett.* 122 (4), 40504. doi: 10.1103/PhysRevLett.122.040504

Shao, R., Bi, X. J., and Chen, Z. (2022). A novel hybrid transformer-CNN architecture for environmental microorganism classification. *PloS One* 17 (11), e0277557. doi: 10.1371/journal.pone.0277557

Shi, S. (2023) *QCCNN-dataSet*. Available at: https://github.com/sunyyds2020/QCCNN-DataSet.git.

Shi, S., Wang, Z., Cui, G., Wang, S., Shang, R., Li, W., et al. (2022). Quantum-inspired complex convolutional neural networks. *Appl. Intell.* 52, 17912–17921. doi: 10.1007/s10489-022-03525-0

Shi, S., Wang, Z., Li, J., Li, Y., Shang, R., Zheng, H., et al. (2023). A natural NISQ model of quantum self-attention mechanism. arXiv:2305.15680. doi: 10.48550/arXiv.2305.15680

Sim, S., Johnson, P. D., and Aspuru-Guzik, A. (2019). Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Adv. Quantum Technol.* 2 (12), 1900070. doi: 10.1002/qute.201900070

Sosik, H. M., and Olson, R. J. (2007). Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. *Limnol. Oceanogr. Methods* 5 (6), 204–216. doi: 10.4319/lom.2007.5.204

Verikas, A., Gelzinis, A., Bacauskiene, M., Olenina, I., and Vaiciukynas, E. (2014). An integrated approach to analysis of phytoplankton images. *IEEE J. Oceanic Eng.* 40 (2), 315–326. doi: 10.1109/JOE.2014.2317955

Wang, C., Zheng, X., Guo, C., Yu, Z., Yu, J., Zheng, H., et al. (2018). "Transferred parallel convolutional neural network for large imbalanced plankton database classification," in *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*. 1–5. doi: 10.1109/OCEANSKOBE.2018.8558836

Wierichs, D., Izaac, J., Wang, C., and Lin, C. Y. Y. (2022). General parameter-shift rules for quantum gradients. *Quantum* 6, 677. doi: 10.22331/q-2022-03-30-677

Zhao, R. X., Shi, J., Zhang, S., and Li, X. L. (2022). QSAN: A near-term achievable quantum self-attention network. arXiv:2207.07563. doi: 10.48550/arXiv.2207.07563

Zheng, H., Wang, R., Yu, Z., Wang, N., Gu, Z., and Zheng, B. (2017). Automatic plankton image classification combining multiple view features *via* multiple kernel learning. *BMC Bioinf.* 18 (16), 1–18. doi: 10.1186/s12859-017-1954-8

Zhong, H., Wang, H., Deng, Y., Chen, M., Peng, L., Qin, J., et al. (2020). Quantum computational advantage using photons. *Science* 370, 1460–1463. doi: 10.1126/science.abe8770

frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 21 February 2024
DOI 10.3389/fmars.2024.1151868

Check for updates

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
J. Xavier Prochaska,
University of California, Santa Cruz, United States
Ming Yang,
Tianjin University, China

*CORRESPONDENCE
Eugenio Cutolo
e.cutolo@protonmail.com

RECEIVED 26 January 2023
ACCEPTED 22 January 2024
PUBLISHED 21 February 2024

CITATION
Cutolo E, Pascual A, Ruiz S, Zarokanellos ND and Fablet R (2024) CLOINet: ocean state reconstructions through remote-sensing, *in-situ* sparse observations and deep learning. *Front. Mar. Sci.* 11:1151868. doi: 10.3389/fmars.2024.1151868

COPYRIGHT
© 2024 Cutolo, Pascual, Ruiz, Zarokanellos and Fablet. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

CLOINet: ocean state reconstructions through remote-sensing, *in-situ* sparse observations and deep learning

Eugenio Cutolo[1,2]*, Ananda Pascual[1], Simon Ruiz[1], Nikolaos D. Zarokanellos[2] and Ronan Fablet[3]

[1]IMEDEA (CSIC-UIB), Esporles, Spain, [2]Balearic Islands Coastal Observing and Forecasting System (SOCIB), Palma, Spain, [3]IMT Atlantique, CNRS UMR Lab-STICC, INRIA team Odyssey, Brest, France

Combining remote-sensing data with *in-situ* observations to achieve a comprehensive 3D reconstruction of the ocean state presents significant challenges for traditional interpolation techniques. To address this, we developed the CLuster Optimal Interpolation Neural Network (CLOINet), which combines the robust mathematical framework of the Optimal Interpolation (OI) scheme with a self-supervised clustering approach. CLOINet efficiently segments remote sensing images into clusters to reveal non-local correlations, thereby enhancing fine-scale oceanic reconstructions. We trained our network using outputs from an Ocean General Circulation Model (OGCM), which also facilitated various testing scenarios. Our Observing System Simulation Experiments aimed to reconstruct deep salinity fields using Sea Surface Temperature (SST) or Sea Surface Height (SSH), alongside sparse *in-situ* salinity observations. The results showcased a significant reduction in reconstruction error up to 40% and the ability to resolve scales 50% smaller compared to baseline OI techniques. Remarkably, even though CLOINet was trained exclusively on simulated data, it accurately reconstructed an unseen SST field using only glider temperature observations and satellite chlorophyll concentration data. This demonstrates how deep learning networks like CLOINet can potentially lead the integration of modeling and observational efforts in developing an ocean digital twin.

KEYWORDS

deep-learning, ocean, remote-sensing, SST, SSH, gliders, OSSE

1 Introduction

Nowadays, there is an increased consciousness of the role played by the ocean in many crucial aspects of human safety, health, and well-being due to the cumulative impacts of climate change, unsustainable exploitation of marine resources, pollution, and uncoordinated development (Ryabinin et al., 2019; Pascual et al., 2021). In response to these challenges,

which UNESCO has encapsulated in 10 objectives for the Ocean Decade (2021-2030), the European Union is endeavoring to develop a digital twin of the ocean. The concept of digital twins involves creating a digital representation of real-world entities or processes, based on both real-time and historical observations, to depict the past and present and to model potential future scenarios.

In the ocean case and especially to address climate change-related concerns, one major challenge is understanding the state and evolution of the ocean's interior. Its stratification significantly influences large-scale integrated variables like ocean heat content, acidification, and oxygenation (Durack et al., 2014; Wang et al., 2018). Moreover, numerous studies have highlighted the importance of resolving submesoscale dynamics to account for the majority of vertical ocean transport, which is vital for carbon export, fisheries, nutrient availability, and pollution displacement (Pascual et al., 2017). These challenges underscore the need for high-resolution, three-dimensional representations of the ocean state. High-resolution numerical models and data assimilation techniques, which align model outputs with actual observations, are currently the most common solutions (Mourre et al., 2004; Carrassi et al., 2018).

Operational simulations now assimilate near-real-time observations, including *in-situ* (ship-based observations, underwater gliders, and floats) and remote sensing data (Hernandez-Lasheras and Mourre, 2018). Satellite observations provide frequent global snapshots of the sea surface, for instance Sea Surface Temperature and Chlorophyll concentration images offer resolutions as fine as 1 km on a daily basis. In contrast, the current capabilities of remote altimeters are limited to a 200 km wavelength for the global ocean at mid-latitudes and about 130 km for the Mediterranean Sea (Ballarotta et al., 2019), though significant advancements are upcoming with the Surface Water and Ocean Topography (SWOT) mission successfully launched in December 2022 (Morrow et al., 2019). Notably, Sea Surface Height (SSH) data are unaffected by cloud cover. Even with such observations about the surface, the uncertainties regarding the ocean interior remain significant due to the sparse distribution of *in-situ* observations in time and space (Siegelman et al., 2019). As a result, while data-assimilating models adhere to physical balances, they still lack accuracy (Arcucci et al., 2021).

The ocean twin strategy proposes data-driven approaches as a complementary method for revealing the ocean state. In previous oceanographic studies, multivariate methods allowed to elaborate three-dimensional hydrographic fields relying on their vast *in-situ* measurements collected during ocean campaigns (Gomis et al., 2001; Cutolo et al., 2022). However, these methods are not easily scalable to a global observing system due to the big number of parameters involved, such as correlation lengths. Machine learning techniques offer a solution to these scalability issues, as the models are directly learned from the data. A key challenge for these techniques is the need for a substantial quantity of realistic training data. General circulation and process study models could then play a new role here, providing a cost-effective way to generate large datasets that adhere to ocean physics in what is usally called a Observing System Simulation Experiment (OSSE) (Arnold and Dey, 1986). Even datasets that only approximate the true state of the ocean can be valuable, as long as they cover a broad range of scenarios. This aspect is particularly important to prevent the risk of deep neural networks merely memorizing the input climatology instead of learning to capture the actual dynamics of the ocean. Training the networks on a wide range of scenarios ensures that they can accurately interpret and adapt to situations that substantially differ from the norm, rather than being limited to recognizing repetitive patterns. Additionally, to effectively generalize beyond their training data, neural networks must be meticulously designed to maintain the integrity of relevant input features throughout their layers. In this context, explainable AI aims to advance beyond the black-box applications typical in ocean remote sensing studies, promoting a deeper understanding of the models data-flows (Zhu et al., 2017).

Despite these difficulties, recent studies have demonstrated the potential of deep-learning methods for various dynamical system tasks. These range from idealized situations (Fablet et al., 2021) to realistic case studies, such as interpolating missing data in satellite-derived observations of sea surface dynamics (Barth et al., 2020; Fablet et al., 2020; Manucharyan et al., 2021). With regard to reconstructing hydrographic profiles from satellite data, there's a spectrum of approaches: from proof-of-concept studies using self-organizing maps (SOMs) and neural networks (Charantonis et al. (2015); Gueye et al. (2014)) and feed-forward or long short-term memory (LSTM) neural networks (Sammartino et al., 2020; Contractor and Roughan, 2021; Fablet et al., 2021; Jiang et al., 2021) as well as (Pauthenet et al., 2022) relying instead on multilayer perceptron. Even considering these past works the interpolation of temperature and salinity profiles given some *in-situ* and sea surface information is an open challenge.

In this study, we introduce an innovative modular neural network designed to seamlessly integrate remote-sensing images with *in-situ* observations for a complete 3D reconstruction of the ocean state. This integration is based on the Optimal Interpolation (OI) scheme's mathematical principles (Gandin, 1966). However, our method differs from traditional applications of OI that usually estimate correlations between points using Euclidean distance. Instead, we calculate distances within a custom-designed latent space. Specific modules within our neural network transform both the input remote-sensing fields and the *in-situ* measurements information into this latent space made of 'clusters'. Within these clusters, multi-variate and non-local correlations become more easily identifiable and can be effectively applied to enhance the correlation matrix. Like attention mechanisms in advanced neural models (Vaswani et al., 2017), which focus on key aspects in large datasets for tasks such as language processing or image recognition, our neural network module similarly identifies crucial correlational patterns through the latent space of clusters.

We privileged a network structure composed of independent nested modules to facilitate the understanding and analysis of its internal information flow from the input data to the covariance structure. To the best of our knowledge, this is the first work in

which neural networks achieve the most optimal combination of remote-sensing and *in-situ* observations without previous knowledge of the study area's climatology. This study is structured as follows: section 2 presents the main synthetic dataset that we used for the training and testing and some real observations for some preliminary use case scenario. All the details regarding the network architecture can be found in section 3, while the results are presented and discussed in section 4.

2 Data

Neural networks need large amounts of data to be trained appropriately. A common choice in oceanography where such a significant quantity of actual observations are unavailable is relying on numerical models. In our case, we chased NATL60, a simulation based on the Nucleus for European Modelling of the Ocean described. We used the fields of this model to simulate both remote-sensing and *in-situ* observations in a so-called Observing System Simulation Experiment (OSSE) (Arnold and Dey, 1986). The model output is sampled in these experiments to replicate the different types of partial observations available. The advantage is that we can quickly check the obtained improvements since the model output also provides the ground truth we aim to reconstruct. The danger of what is usually called "supervised learning" only aiming to minimize the discrepancy with the provided ground truth is that the network weights memorize the "right answers" so in our context the model climatology. We faced this problem, including two self-supervised terms in our loss function as we describe later but also accurately selecting a highly varying training and test dataset as presented here in subsection 2.1.

Finally, we proved the generalization capabilities of our network, testing it with actual multi-platform observations. In particular, we used the remote-sensing products of Sea Surface Temperature (SST) and Chlorophyll-a concentration (CHL) from CMEMS, together with temperature observations from gliders, as described in subsection 2.2.

2.1 eNATL60 based OSSE

Our primary experiments utilized the eNATL60 configuration of the Nucleus for the European Modelling of the Ocean (NEMO) model (Gurvan et al., 2022), featuring a 1/60° horizontal resolution and 300 vertical levels across the North Atlantic. This high-resolution configuration is essential for understanding ocean dynamics, particularly for surface oceanic motions down to 15 km, which aligns with SWOT observations (Ajayi et al. (2020)). We direct readers to this work for a detailed understanding of NATL60's capabilities. Additionally, numerous studies have employed the non-extended version of NATL60 for resolving fine-scale dynamical processes (Amores et al., 2018; Fresnay et al., 2018; Metref et al., 2019; Metref et al., 2020).

For our training and testing data, we utilized daily averages of Sea Surface Temperature (SST) and Sea Surface Height (SSH), both individually and combined, from the eNATL60 simulation spanning an entire year. Alongside these, we gathered *in-situ* salinity observations at three specific depths: 5 m, 75 m, and 150 m. Our focus was then to reconstruct the 2D salinity fields at these depths. In particular our analysis predominantly focused on the 5 m and 150 m depths, selected to assess the robustness of our model both within and beyond the mixed-layer depth. To ensure that our network's training and testing *in-situ* observations mirrored real oceanographic conditions, we adopted two distinct sampling strategies: random and regular. This approach allowed us to evaluate the network's performance in various realistic observational scenarios. The random strategy selects N domain points based on a uniform distribution, while the regular strategy uses a homogeneous grid sampling with a fixed spacing of δx. By varying N and δx, we conducted different experiments to observe metric variations.

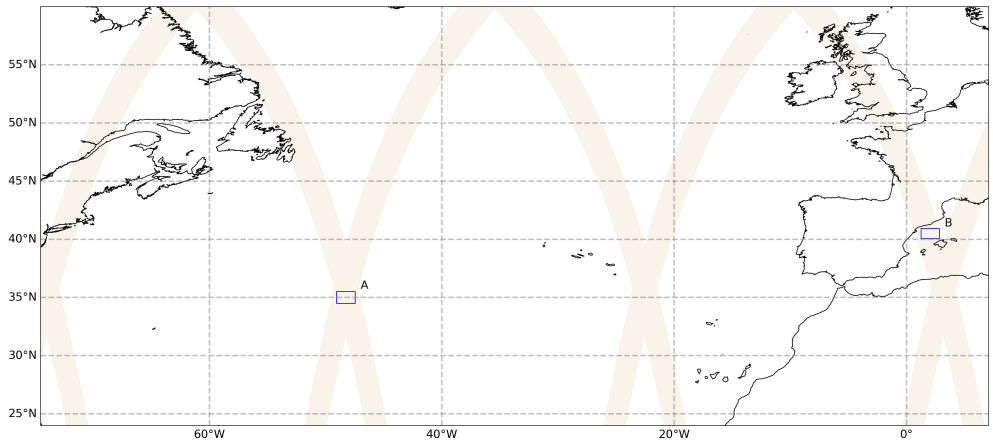


FIGURE 1
Training area **(A)** and testing area **(B)** presented with the SWOT passages in the fast-sampling phase. The coordinates of the SWOT passages comes from the simulated SWOT product from the MITgcm LLC4320 model (L2 LR SSH), available on the AVISO website: http://doi.org/10.24400/527896/a01-2021.006.

Our focus was on two marine areas: the subpolar northwest Atlantic for training, and the Western Mediterranean Sea for testing. Both regions are notable for SWOT passages during its rapid-sampling phase (see Figure 1). The Mediterranean region, in particular, is known for its dynamic oceanographic characteristics and has been extensively studied through *in-situ* and remote-sensing methods (Ruiz et al., 2009). Using different regions for training and testing helps prevent overfitting in the neural network. Overfitting occurs when a model learns the specifics and noise in the training data to an extent that interfere with the model's performance on new data. Since the climatology of the northwest Atlantic differs significantly from that of the Western Mediterranean Sea, we ensure that our network is not just memorizing patterns from the training data but is effectively learning to generalize across different oceanographic contexts. Additionally, we diversified the dataset by sampling the same day with varying N or δx values.

For simplicity, our approach assumes a synoptic scenario, where all observations occur simultaneously. Future work will address the non-synoptic nature of actual sampling and explore how the network accommodates this. Furthermore, in this study, we did not incorporate simulated noise or measurement errors into our data, opting to explore these aspects in subsequent research. Despite this, the practical effectiveness of our network is demonstrated through tests using actual observational data, details of which are provided in the following subsection.

2.2 Real observations

2.2.1 Remote-sensing observations

In our study, we have used Sea Surface Temperature (SST) and Ocean Color (CHL) imagery from the 18th of February, 2022, distributed by CMEMS. The CHL has 1 km spatial resolution, and it is a level-3 product obtained by multi-Sensor processing from OceanColor (Volpe et al., 2019). The SST also has a 1 km spatial resolution and it is based on level-2 product based on multi-channel sea surface temperature (SST) retrievals, which it has generated in real-time from the Infrared Atmospheric Sounding Interferometer (IASI) on the European Meteorological Operational-A (MetOp-A) satellite.

2.2.2 Glider observations

Gliders are autonomous underwater vehicles that allow sustained collection at high spatial resolution (1 km) and low costs compared to conventional oceanographic methods. Many studies confirmed the feasibility of using coastal and deep gliders to monitor the spatial and low-frequency variability of the coastal ocean (Alvarez et al., 2007; Heslop et al., 2012; Ruiz et al., 2019; Zarokanellos et al., 2022). In this work we used the observations from two gliders in the Balearic Sea as a part of the Calypso 2022 experiment. The two gliders carried out a suite of sensors that measure temperature, conductivity and pressure (CTD), dissolved oxygen (oxygen optode), Chlorophyll fluorescence and Turbidity (FNLTU). The two gliders were programmed to profile from the surface up to 700 m with a vertical speed of 0.18 ± 0.02 m/s and moved horizontally at approximately 20–24 km per/day. Data were processed following the methodology described in Troupin et al. (2015). In this study, we have used the temperature data at 15 m from the 10th of February until the 18th of February.

3 Methods

When sparse observations are available, the most common technique that has been adopted in oceanography and in different fields of science using a gridded product is Optimal Interpolation (OI) (Gandin, 1966). The technique relies on a solid mathematics basis and has been the state-of-the-art approach for many geophysical products until now. Since the proposed neural approach and specifically our prior builds over the OI framework we reviewed it in subsection 3.1. Then, we introduce CLOINet our neural approach and its submodules in subsection 3.2. Lastly, we present the metrics we used for bench-marking purposes.

3.1 Baseline: OINet

A common approach to explain the OI math start considering $\mathbf{y}$ as the vector containing all the observations we have of the true state $\mathbf{x}$, which is unknown. We can relate them with the following observation model Equation 1:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \epsilon \tag{1}$$

where $\mathbf{H}$ is the observation (or masking) operator, and ϵ is the observation error. Under Gaussian hypotheses for ϵ and the prior on $\mathbf{x}$, we can obtain the best possible estimation of true state $\mathbf{x}_s$ given the observations $\mathbf{y}$ through a linear operator $\mathbf{K}$ (the Kalman gain see Welch and Bishop (1995)):

$$\mathbf{x}^s = \mathbf{K}\mathbf{y} \tag{2}$$

$$\mathbf{K} = \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1} \tag{3}$$

where $\mathbf{R}$ is the observation error covariance matrix and $\mathbf{B}$ is the error covariance specific of the analysis. In Equation 3, we are assuming an a-priori knowledge of both $\mathbf{R}$ and $\mathbf{B}$, which could be theoretically obtained by repeating the same experiments many times. Practically, a parameterized covariance matrix is often used to substitute the complete climatology covariances Gaspari and Cohn (1999). The most common parametrization for this matrix is a Gaussian-shaped correlation, depending only on the points' distances and pre-determined correlation lengths. So for two generic position vectors r_i and r_j, we have:

$$\mathbf{B}_{i,j} = cov(r_i, r_j) = e^{-\sum_{n=1}^{3} \frac{(r_{i,n} - r_{j,n})^2}{2c_n^2}} \tag{4}$$

where the sum for dimension n considers the squared difference of the components of the position vectors $r_{i,n}$ and $r_{j,n}$ divided by the squared nth correlation length c_n. Regarding the observation error matrix, we assume from now on that it is diagonal Equation 5:

$$\mathbf{R}_{i,j} = \mathbf{I}_{i,j}\epsilon \tag{5}$$

A different case where the observation errors are correlated is possible. However, **R** is often assumed diagonal to reduce computational costs (Miyoshi and Kondo, 2013). Finally, inserting **B** and **R** in Equation 3 and then in Equation 2 we can compute our estimated field $\mathbf{x}^s$.

In our experiments, we established a baseline method with OINet, a simple neural network, that automatically discover the OI correlation lengths among different variables (SST, SSH, and salinity) and dimensions. OINet is then provided with the same input data as CLOINet, including surface fields (SST and/or SSH) and *in-situ* salinity observations. It operates in a two-step process: the first step involves transforming the multivariate surface fields into a unified field, making it compatible for being used with the salinity observations. The second step is to estimate the three correlation lengths specific to the current set of observations. While the first step involves 2 convolutional layers the second one is a simple feed-forward neural networks able to process a generic number of *O* observations (see the bottom part of Figure 2).

Beyond the parameter estimation this module is simply realizing an OI using the formula in Equation 4 to calculate the covariances. Notably this approach not only automates the tuning of parameters but also leverages GPU power for more efficient interpolation computations.

3.2 CLuster enhanced Optimal Interpolation Network

Ocean dynamics often display non-local and anisotropic patterns, which traditional Optimal Interpolation (OI) methods struggle to account for effectively. The main challenge with OI lies in its correlation function assumptions, which may not accurately reflect the actual physical conditions of the ocean. For instance, as seen in Equation 4 OI typically presumes that points in close proximity are strongly correlated, while distant points are not. However, oceanographic phenomena can exhibit the opposite behavior. For example, ocean fronts, characterized by narrow zones with strong horizontal density gradients, act as boundaries

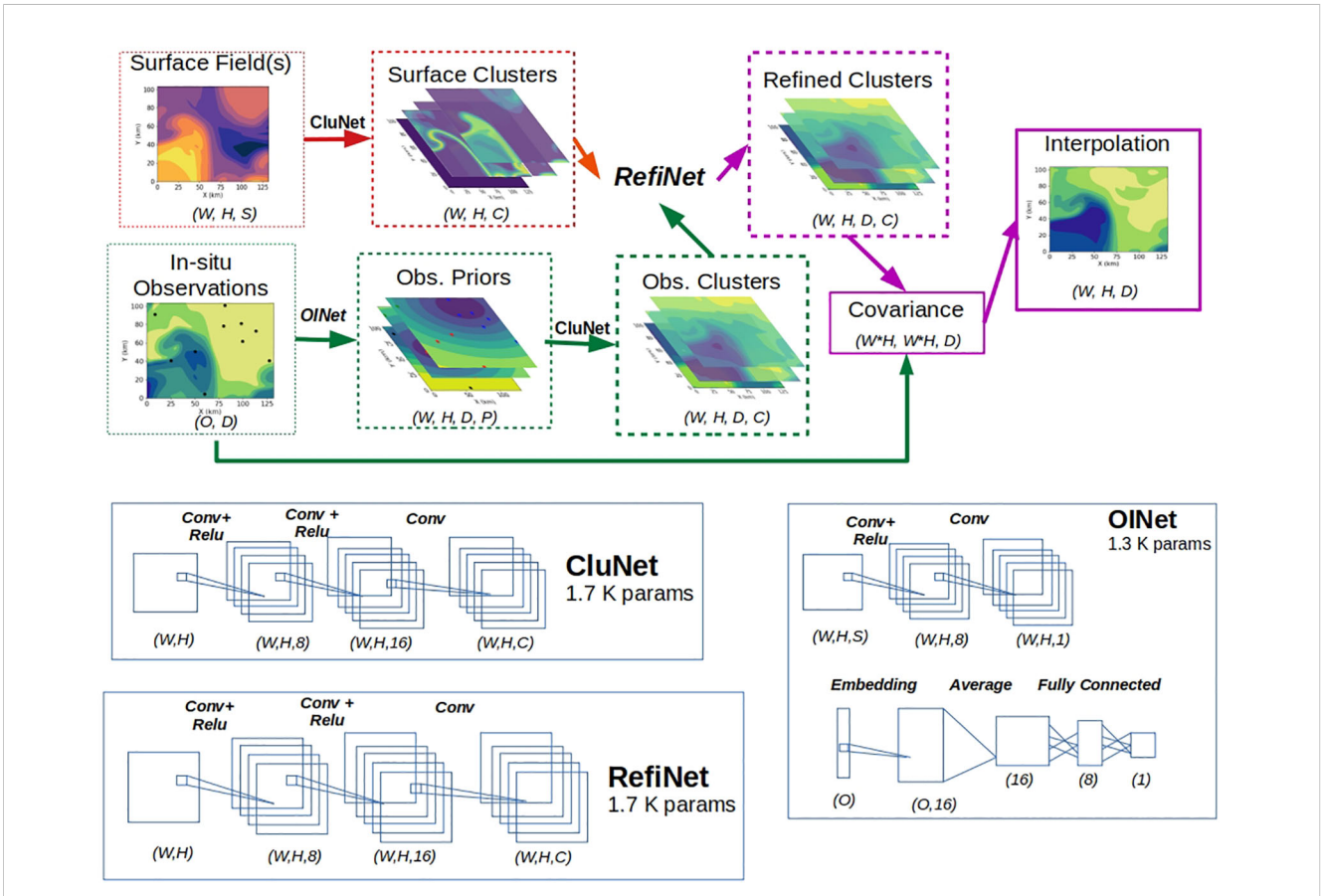


FIGURE 2
Flow chart of CLOINet information processing: Red and green elements (boxes and arrows) represent the processing paths for the SST input surface field and the *in-situ* salinity observations, respectively. Purple elements indicate the combined use of both inputs. The bold text along the arrows specifies the network module in operation. The line style and width of each box vary to represent different processing stages, ranging from thin and dashed for inputs to solid and thick for outputs. Inside each box, capital letters denote the corresponding tensor dimensions: W for field width, H for field height, S for the number of input surface fields, P for the number of OINet priors, C for the number of clusters (consistent across all modules in our tests), D for the number of depths, and O for the number of *in-situ* observations. Colorbars are omitted for clarity. The lower part of the image illustrates the CNN architecture of the three modules, along with the number of parameters used.

between water masses with distinct physical and bio-optical properties. Conversely, in dynamic ocean features like meanders and eddies, water masses can remain similar over vast distances.

Here, we aim to benefit from the wealth of information from remote sensing regarding the shape of the ocean features, whether they belong to the mesoscale or the submesoscale. The key idea is grouping a set of objects in such a way that each object is more similar to the objects belonging to its same group (called a cluster) than the rest. This procedure in statistics is called clustering. Applying this concept to reconstructing the ocean state, our approach is to reveal non-local correlations by clustering grid points that are part of the same oceanic features. This led us to develop CLOINet (Cluster-enhanced Optimal Interpolation Net), an end-to-end system designed to optimally interpolate sparse *in-situ* observations using available remote-sensing images. CLOINet is able to process any kind of surface fields (2D images) and *in-situ* observations (2D masks and observation values). Its main submodule is CLuNet, which transforms 2D fields into fuzzy clusters. While satellite images could directly been passed to this module *in-situ* observation profiles are initially processed by OINet, which serves as a prior, converting them into images. Finally a further submodule, RefiNet, module merges the fuzzy clusters from both surface fields and observation priors into a final cluster set. Within this latent cluster space an alternative distance could replace the euclidean distance allowing a better estimation of $\mathbf{B}$ and consequently obtains the reconstructed field $\mathbf{x}^s$.

Our network structure allows a joint training of all modules, minimizing their specific loss function terms summed up in a global loss function. Convolutional Neural Networks (CNN) layers. Here following, we describe the details of the network submodules and how we obtained the interpolation in an end-to-end scheme also summarized in Figure 2).

3.2.1 Clusters space transformation: CluNet

The first module of our scheme, called CluNet is in charge of transform any images into a set of clusters. Piratically speaking it segments the input 2D images (like multivariate remote-sensing fields or the observations priors) into C clusters of similar points. In this context, we consider two points similar according to their positions (as in Equation 4) but also their values in the input 2D fields. In particular, we worked within the so-called "fuzzy logic", where the membership function m_{jk}, which expresses how much the j point belongs to the k cluster, could assume every value between 0 and 1. Considering this continuous range means that each grid point could be part of more than one cluster as long as the following normalization holds:

$$\sum_{k=1}^{C} m_{jk} = 1 \quad \forall j \tag{6}$$

For its non-binary logic, this clustering technique is called "Fuzzy Clustering" or soft k-means. being this last the simpler binary case in which m_{jk} could be just 1 or 0. CluNet takes the remote-sensing images as input and gives the tensor composed by all the m_{jk} through various CNN and finally a softmax layer to guarantee Equation 6. The associated training loss, referred to as a Robust fuzzy C-means (Chen et al., 2021) loss, is composed of two terms:

$$\mathcal{L}_{RFCM}(\mathbf{y};\ \theta) = \sum_{j\in\Omega}\sum_{k=1}^{C} m_{jk}^{q}(\mathbf{y};\theta)\left\|y_j - v_k\right\|^2 + \beta \sum_{j\in\Omega}\sum_{k=1}^{C} m_{jk}^{q}(\mathbf{y};\theta) \sum_{l\in N_j m\in M_k} \sum_{lm} m_{lm}^{q}(\mathbf{y};\theta) \tag{7}$$

$\mathbf{y}$ is the vector containing the surface field that we want to cluster with y_j its value at point j in our domain Ω. q is a parameter that satisfies $q \geq 1$ and controls the amount of fuzzy overlap between clusters. Minimizing the first term achieves that points with high membership function for the k cluster should be similar to its center v_k defined as follows Equation 8:

$$v_k = \frac{\sum_{j\in\Omega} m_{jk}^{q}(\mathbf{y};\theta) y_j}{\sum_{j\in\Omega} m_{jk}^{q}(\mathbf{y};\theta)} \tag{8}$$

The second term guarantees the membership function's spatial smoothness, forcing the j point to have a similar value to its neighborhood N_j. The parameter β controls the intensity of this constraint.

In summary, to obtain the clustering, we minimize Equation 7 with respect to the parameters of the CNN layers included in CluNet, which stand in the θ vector. Since in this loss term, we do not directly provide any ground truth (i.e., the best way of clustering the inputs), this part of the network could be considered self-supervised since it learns indirectly from the rest of the loss term. As it show in Figure 2 we used this module twice, firstly for clustering surface input fields and secondly for clustering the 2D fields coming from the observations priors described hereafter. Consequently in the global loss there are two terms like Equation 7.

3.2.2 Observations priors

We have outlined the process by which CluNet segments any set of 2D fields into distinct clusters. To handle *in-situ* observations, which are essentially vectors of observations at different depths, we utilize OINet to convert them into a series of images that can then be clustered. As previously mentioned, OINet has the capability to autonomously determine the appropriate correlation lengths for a given set of observations and then perform a canonical Optimal Interpolation (OI). In our approach, we generate four different versions of these interpolations, each initiated with correlation lengths that are submultiples of the domain sizes. These parameters, among others, are then fine-tuned during the learning phase. This process results in four fields that CluNet subsequently clusters into areas exhibiting similar values, despite being derived using different correlation lengths. The clusters formed from these observations provide insights into the certainty we have about specific regions and the extent to which a particular depth is influenced by surface conditions. Essentially, this method allows us to address potential anisotropy in the uncertainties without having to rely on fixed length scales.

3.2.3 Data fusion in the clusters space: RefiNet

We now have a set of clusters derived from the surface fields, and an additional set for each depth of the *in-situ* observations. For each depth, the corresponding sets of surface and observation clusters are processed through RefiNet. The resulting clusters, along with their membership vectors, are used to compute the covariance matrix as follows Equation 9:

$$\mathbf{B}_{i,j} = cov(r_i, r_j) = 1 - \sum_{k=1}^{C} (m'_{ik} - m'_{jk})^2 \tag{9}$$

In this equation, we sum the differences in the membership functions of points ii and jj across all clusters. This process, while bearing similarities to Equation 4 deviates by using subtraction instead of an exponential function since m_{ik} and m_{jk} are already bounded within the 0-1 range. This summation represents a non-local distance in the cluster space, replacing the classical Euclidean distance. Consequently, two points within the same cluster (i.e., with similar membership vectors) will be correlated, regardless of their spatial distance.

Using parametrization (9), we then compute the associated optimal interpolation as Equation 3 and then Equation 2. This forms an end-to-end architecture that uses remote sensing images and *in-situ* data to output regularly-gridded vertical profiles (see Figure 2 for the data flow).

The training loss combines three components: two clustering-based losses Equation 7 (one for the surface fields and one for the observations priors) and a supervised reconstruction term. So globally we minimize Equation 10:

$$\mathcal{L} = \alpha \mathcal{L}_{srf_R FCM} + \beta \mathcal{L}_{obs\ _R FCM} + \gamma \mathcal{L}_{MSE} \tag{10}$$

where α, β and γ are the weights of the three loss terms, and $L_{MSE}$is given by Equation 11:

$$\mathcal{L}_{MSE} = (\mathbf{x}^s - \mathbf{x})^2 \tag{11}$$

This last term is just the mean squared error with respect to the ground truth **x**. Within the considered supervised training strategy, self-supervised losses Equation 7 act as regularization terms to improve generalization performance and explainability. We maintain equal weights of α, β, and γ at 1, as no significant differences were observed with other values. Our network also shows relative insensitivity to other hyperparameters, such as the number of clusters. However, our cross-validation tests indicated that setting this number to 20 yielded the best results.

3.3 Performance metrics

To understand how the clusters sets were changing according to the input data we computed the associated entropy fields. In fact, given that the membership vector is normalized and thus it can be seen as a distribution, its entropy definition is Equation 12:

$$Si = -\sum_{k=1}^{C} m'_{ik} \log m'_{ik} \tag{12}$$

To assess the performance of the proposed approach, we first define the error between the ground truth and the estimated field value Equation 13:

$$\mathbf{x}_{err} = \mathbf{x} - \mathbf{x}_s \tag{13}$$

then we easily obtain our first performance metric: the Root Mean Squared Error (RMSE) Equation 14:

$$RMSE = \sqrt{x_{err}^2} \tag{14}$$

We will present this metric in percentage of the standard deviation of the ground truth fields. Now considering the standard deviation of the error over the whole N snapshot Equation 15:

$$\sigma_{err} = \frac{\sum_{t=1}^{N} \left(\mathbf{x}_{err}(t) - \overline{\mathbf{x}_{err}(t)}\right)^2}{N} \tag{15}$$

we can compute the explained variance score dividing by the standard deviation of the ground truth Equation 16.

$$\sigma_S(x, y) = 1 - \frac{\sigma_{err}}{\sigma_{true}} \tag{16}$$

To highlight the effective resolution of the different reconstruction methods we use the noise-to-signal ratio NSR (Ballarotta et al., 2019) Equation 17:

$$NSR(\lambda) = \frac{PSD(\mathbf{x}_{err}, \lambda)}{PSD(\mathbf{x}, \lambda)} \tag{17}$$

the effective resolution is in fact given by the wavelength λ_s where the NSR λ_s is 0.5.

4 Results and discussion

This section first reports numerical experiments using NATL60 OSSEs to evaluate the proposed approach quantitatively. The concluding subsection presents an application to real observations.

4.1 Clusters entropy

The initial part of our analysis focuses on understanding how CLOINet, via CluNet and subsequently RefiNet, organizes clusters based on different data inputs: SST, SSH, and various sets of randomly located *in-situ* salinity observations. To illustrate this, we plotted some example entropy fields in Figure 3 along with statistics on how entropy changes with an increasing number of observations N. In the four panels on the left side of Figure 3 we display two clusters' entropy fields (panels a and e) and their corresponding input fields for SSH (panel b) and SST (panel f) from a selected snapshot. In the four panels on the right side, we present the entropy associated with the *in-situ* observations' clusters at two different depths $z = 5$ (panel c) and $z = 150m$ (panel g) together with the correspondent refined clusters entropy (panel d and h) along with the refined clusters' entropy (panels d and h).

This particular snapshot was chosen for its submesoscale features. The differences between SSH and SST-based entropy are

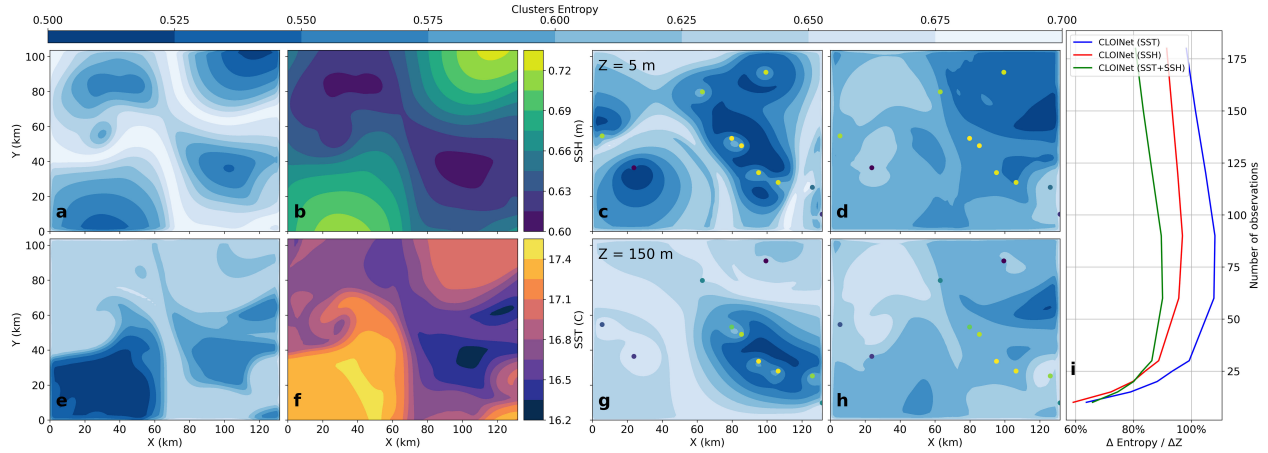


FIGURE 3
The entropy of the cluster sets, resulting from the input SSH and SST, is depicted in panels a and e, respectively, while the corresponding fields themselves are shown in panels b and f. Panels c and d (and g and h) display the entropy of the observation and refined cluster sets at a depth of Z = 5 m and Z = 150 m, respectively. The dots in these panels represent salinity observations at these depths, with their colors indicating the magnitude of salinity (scale not shown). Panel i illustrates the variation in the entropy of the refined cluster sets along the vertical axis, corresponding to different numbers of observations. The varying colors in this panel represent different networks.

noticeable; the SSH clusters highlight more prominent features, while SST forms smaller clusters that extend to deeper depths. The correspondence between the surface fields and their cluster entropy is relatively straightforward, but differences in other sets are more subtle. For observation clusters' entropy, we observe lower entropy (blue regions) near points with similar observations. Areas of higher entropy occur between two observation points with differing values. This behavior varies at different depths, explaining the differences between panels c and d. The refined clusters, influenced by both observations and surface fields, exhibit subtler changes, but we can still see an increase in entropy with depth, particularly noticeable in the northeast region of panels d and h.

Beyond this specific snapshot, panel i shows the percentage change in entropy between the two depths, averaged across the entire test dataset as a function of the number of observations. When only SST data is available, the changes in clusters are more pronounced, as SST information is less directly related to the ocean's interior compared to SSH or combined SST and SSH data. As expected, all deltas increase with the number of observations, eventually reaching a saturation point where they decrease. This occurs because the clusters' information becomes less critical, and the field can be reconstructed relying primarily on *in-situ* observations.

4.2 RMSE and correlation

WWe present the outcomes of the random sampling OSSE in Figure 4. The first two rows illustrate a ground truth salinity example at two different depths, alongside the reconstructions by the baseline OINet and CLOINet with various surface input fields. Again, we chose the same snapshot from Figure 3 for its distinct submesoscale features.

OINet can effectively use surface information for reconstructing the surface layer, but it struggles to propagate this information to deeper layers. We also experimented forcing a bigger correlation length in the z axis but we ended up with a reversed scenario: a well-reconstructed bottom layer but a poorly reconstructed (not shown). This limitation arises because the simple network cannot determine which surface fields to prioritize based on the *in-situ* observations.

In the case of CLOINet, we observe different results based on the input fields provided. SST leads to better surface interpolations, while SSH is more effective for deeper fields. This outcome aligns with our expectations, as SSH data is depth-integrated and thus more informative than SST for understanding the shape of water masses at depth. Notably, when both SST and SSH are used as inputs, the network effectively leverages their shape information to enhance both surface and interior reconstructions, leading to a reduction in RMSE by about 40% at both depths.

The results across the entire testing set show similar patterns. On panel l (m) we show how for all methods, the RMSE (correlation) decreases (increases) in proportion to the number of observations. In these plots, solid lines represent surface salinity fields, while dashed lines indicate interior fields at $z = 150m$. Interestingly, on average, OINet's performance is comparable to CLOINet's for surface reconstructions but falls short for interior reconstructions. This fact is mostly related with presence or not of submesoscale features as the next subsection analysis will show. The variation in CLOINet's surface inputs shows minimal impact on surface results, with only slight improvements observed in the SST +SSH case. However, the introduction of the SSH field significantly enhances the interior field reconstructions. Once again, this confirms that SSH provides more comprehensive information about the entire water column compared to SST.

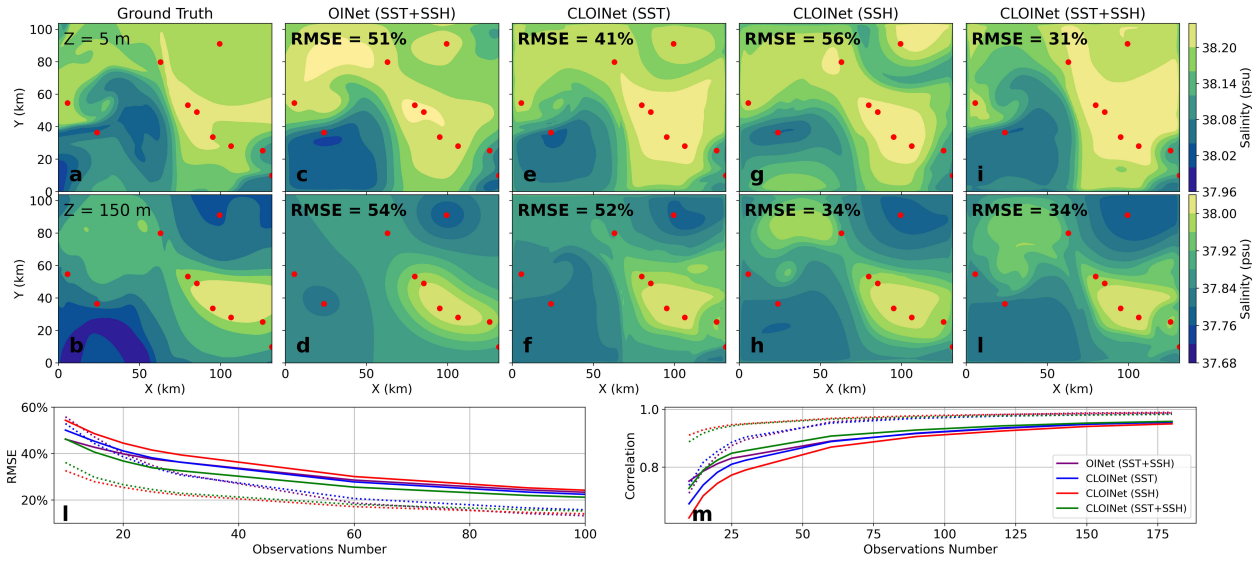


FIGURE 4
Example of salinity field interpolation at Z = 5m (first row, panels **A, C, E, G, I**) and Z = 150m (second row, panels **B, D, F, H, I**) with ten random observations. The first column shows the ground truth, and the subsequent columns represent various interpolation methods. The two bottom plots display the RMSE (panel **L**) and correlation coefficient (panel **M**) as functions of the number of observations in a random sampling scenario, averaged across the entire test dataset. In these plots, the solid line corresponds to Z = 5m, while the dashed line represents Z = 150m.

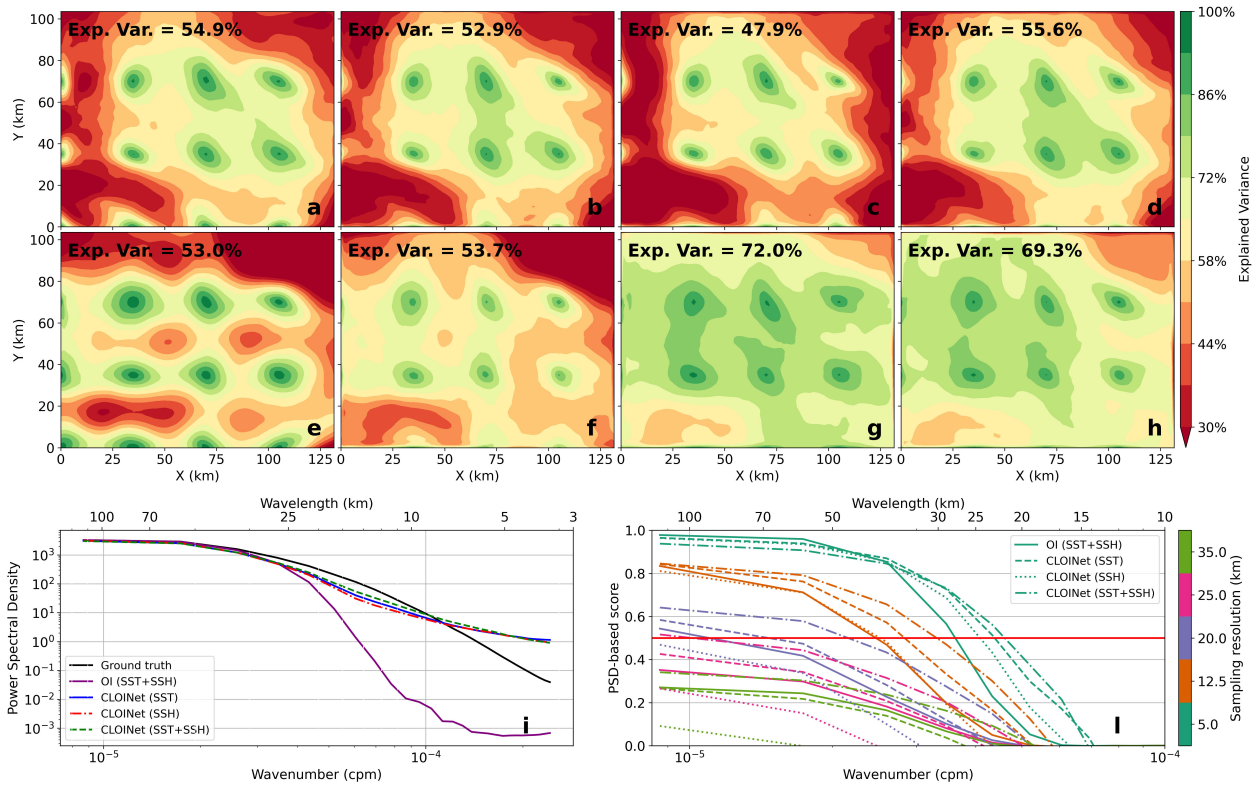


FIGURE 5
This figure shows the explained variance for various interpolation methods averaged across the entire test dataset, in a scenario with regular sampling at 45 km intervals. Panels **(A-D)** display the results at Z = 5m for OI, CLOINet-SST, CLOINet-SSH, and CLOINet-SST+SSH, respectively, panels **(E-H)** are instead relative to Z = 150m. Black dots mark the locations of *in-situ* observations while the spatial average value for each method is indicated on the corresponding subplot. Panel **(I)** compares the Power Spectral Density (PSD) of the different reconstruction methods with the ground truth, with each color representing a different method. Panel **(L)** shows the corresponding score, where the colors denote different sampling resolutions. In both panels **(I, L)**, solid lines represent surface fields, while dashed lines correspond to fields at a depth of z = 150m. The red line across panel **(L)** marks the 0.5 threshold value, indicating the effective resolution of the interpolation methods.

4.3 Resolved scales

In Figure 5, we present the results of the OSSE conducted with regular grid sampling, varying the spacing between observations to understand how different methods resolve various spatial scales. Specifically, we examined the impact of sampling resolution on the explained variance and the Power Spectral Density (PSD)-based score. The explained variance for the different reconstruction methods at a sampling resolution of 20 km is shown in panels a, c, e, and g (and panels b, d, f, and h for the interior field). When provided with the same inputs as OINet (SST and SSH), CLOINet slightly surpasses it on the surface and by about 20% in the interior. Again, we observe superior performance from CLOINet-SSH in the interior, while the inferior performance of the network relying solely on SST suggests that, on average, this field does not significantly account for salinity variability.

The PSD-based score, shown in panel l, indicates the effective resolution of the reconstruction (the point at which the score falls below 0.5) demonstrating how CLOINet generally resolves smaller scales than OINet across various sampling resolutions. For higher resolutions, such as 5 and 12 km, CLOINet resolves scales approximately 1.5 times larger than OINet. The training set's averaged spectra, depicted in panel i, reveal that OINet is typically limited to reconstructing larger scales. Indirectly, this suggests that the variability explained in the test region is predominantly due to larger scales, which even OINet can adequately account for.

4.4 Real ocean data preliminary tests

In line with many deep learning studies, our research focuses on applying neural networks, initially trained on synthetic data, to real-world observations. We evaluated CLOINet's effectiveness in improving Sea Surface Temperature (SST) estimates using glider surface temperature observations, enhanced with shape information from a Chlorophyll (CHL) snapshot (refer to Figure 6). Both OINet and CLOINet were able to reconstruct the general SST pattern observed in reality. CLOINet demonstrated a slightly superior performance, as evidenced by higher correlation values. This improvement aligns with our qualitative observations, suggesting that CLOINet more accurately preserves submesoscale features. Notably, this achievement was realized without the networks being specifically trained on CHL data. In these preliminary tests, the CHL data was provided as if it were the SST and SSH fields, demonstrating the networks' versatility in utilizing shape information from various types of variables. Achieving similar levels of accuracy with traditional Optimal

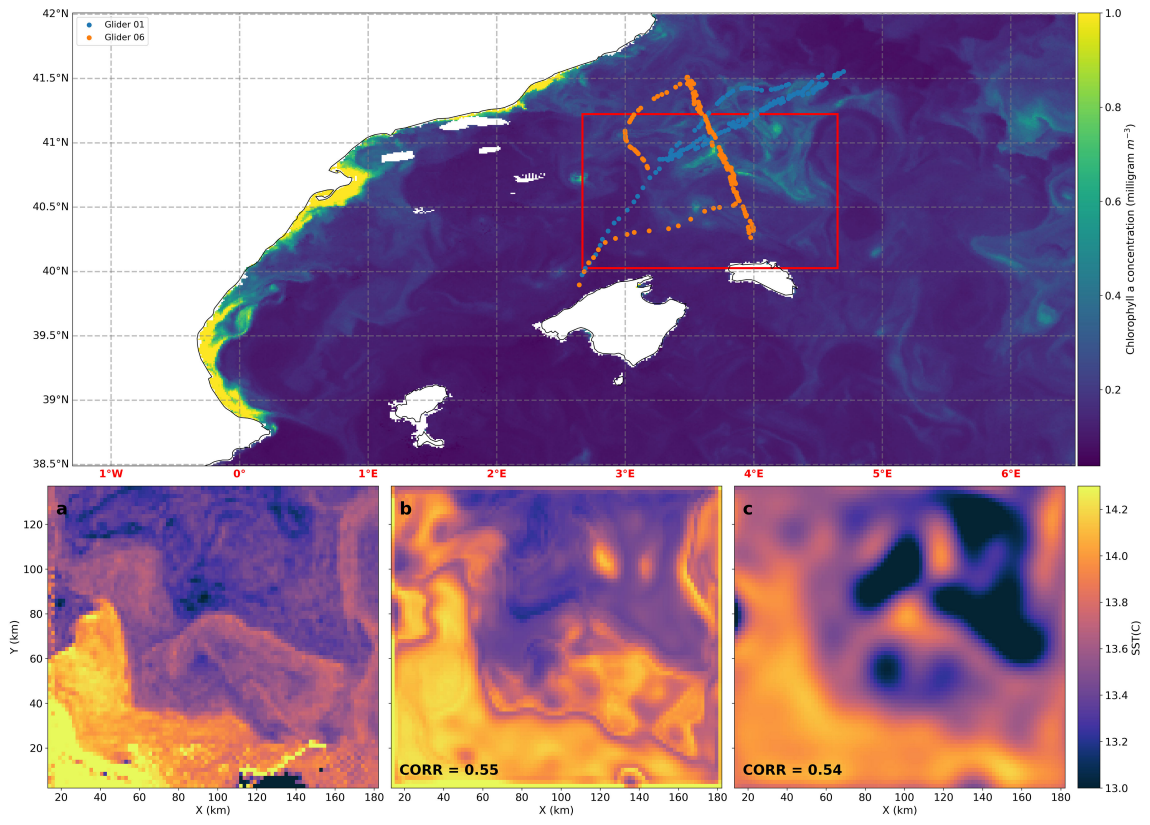


FIGURE 6
The top panel displays the Chlorophyll-a (CHL-a) concentration in the Western Mediterranean Sea on February 18, 2022. The dots represent temperature observations from two gliders over the preceding 48 hours, while the red box outlines the area where interpolation was performed. The bottom panels, labeled a, b, and c, show the actual Sea Surface Temperature (SST) this same day and the reconstructed SST using CLOINet and OINet, respectively. The correlation values for each reconstruction method are indicated in the corresponding plots.

Interpolation (OI) methods would be more complex, likely necessitating intricate, predefined multi-variate correlation functions and extensive parameter tuning.

5 Conclusion

In this, we presented CLOINet, a comprehensive end-to-end neural network designed to combine sparse *in-situ* observations into a full 3D field leveraging shape information from kind of ocean remote sensing images. We conducted end-to-end training of CLOINet within a supervised framework, using Observing System Simulation Experiments (OSSEs) based on the NEMO-derived NATL60 simulation. Our study focuses on comparing the reconstruction of 3D salinity capabilities of CLOINet with those of a data-driven version of classical Optimal Interpolation, which we have named OINet. This comparison also extends to applications involving real observational data.

Our research covered various scenarios, including both randomly and regularly spaced *in-situ* salinity observations, paired with different remote sensing inputs such as Sea Surface Temperature (SST), Sea Surface Height (SSH), or a combination of both. Upon creating a 3D salinity field, we thoroughly analyzed how our performance metrics responded to variations in the number and density of *in-situ* observations.

In dense regular sampling we showed how CLOINet was able to resolve scales 1.5 smaller scales compared to OINet while in random sampling contexts, CLOINet showed enhanced performance in terms of both RMSE and correlation, especially notable when limited observations were available. This improvement was significant in scenarios involving in-depth fields and areas rich in submesoscale features, where RMSE improvements reached as high as 40%.

Despite not incorporating simulated errors to mimic actual sampling instruments, the promising results with real data highlight the potential of our approach in operational contexts. In fact, CLOINet adeptly handled noisy CHL fields and gliders *in-situ* temperature and successfully reconstructed the general pattern of an unseen SST field, without specific training for this task. These outcomes also demonstrate that, apart from reconstructing salinity, the process of transforming input data into a latent space composed of clusters enables comprehensive multi-variate analysis.

Our training approach, which combined two self-supervised losses with a supervised reconstruction loss, enabled the network to generalize effectively. This was evident as it performed accurately in the Western Mediterranean test area, distinctly different from the North Atlantic training region. This suggests that our method is not limited by specific regional climatology and could potentially be scaled for global application.

Overall, the modular design of CLOINet not only enhances our understanding of its internal processes but also positions it for future enhancements. One promising direction for subsequent research is extending the model to incorporate space-time dynamics. Another intriguing possibility is employing this neural network approach for guiding an adaptive sampling multi-platform ocean campaign. Given the significant role of SSH data in assessing the reconstruction of the deeper water layers, the upcoming high-resolution SSH observations from SWOT present an exciting opportunity for further refining and applying CLOINet.

Additional requirements

For additional requirements for specific article types and further information please refer to AuthorGuidelines.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

EC, AP, SR, and RF contributed to conception and design of the study. EC developed the software CLOINet, performed the statistical analysis and wrote the first draft of the manuscript. NZ helped for writing the glider section. All authors contributed to the article and approved the submitted version.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. EC acknowledges support from the Spanish Ministerio de Ciencia e Innovacion (grant BES-2017-080763). This article is also a contribution to the PRE-SWOT and FAST-SWOT projects funded by the Spanish Research Agency and the European Regional Development Fund (AEI/FEDER, UE) under grant agreements (CTM2016-78607-P) and (PID2021-122417NB-I00) respectively. Furthermore AP and SR were funded by the CALYPSO project Office of Naval Research grant N0014-21-1-2702.

Acknowledgments

EC acknowledges the MEOM Research Group group for kindly providing the eNATL60 model outputs and all the members of the CALYPSO project for their comments and suggestions. EC acknowledges Prof. Carlos Granero Belinchon for engaging in

valuable discussions that significantly enhanced the quality of the manuscript.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ajayi, A., Le Sommer, J., Chassignet, E., Molines, J. M., Xu, X., Albert, A., et al. (2020). Spatial and temporal variability of the North Atlantic eddy field from two kilometric-resolution ocean models. *J. Geophys. Res.: Oceans.* 125, e2019JC015827. doi: 10.1029/2019JC015827

Alvarez, A., Garau, B., and Caiti, A. (2007). "Combining networks of drifting profiling floats and gliders for adaptive sampling of the Ocean," in *Proceedings - IEEE International Conference on Robotics and Automation*, 157–162. doi: 10.1109/ROBOT.2007.363780

Amores, A., Jordà, G., Arsouze, T., and Le Sommer, J. (2018). Up to what extent can we characterize ocean eddies using present-day gridded altimetric products? *J. Geophys. Res.: Oceans.* 123, 7220–7236. doi: 10.1029/2018JC014140

Arcucci, R., Zhu, J., Hu, S., and Guo, Y. K. (2021). Deep data assimilation: integrating deep learning with data assimilation. *Appl. Sci.* 11, 1114. doi: 10.3390/APP11031114

Arnold, C. P., and Dey, C. H. (1986). Observing-systems simulation experiments: Past, present, and future. *Bull. Am. Meteorol. Soc.* 67, 687–695. doi: 10.1175/1520-0477(1986)067<0687:OSSEPP>2.0.CO;2

Ballarotta, M., Ubelmann, C., Pujol, M. I., Taburet, G., Fournier, F., Legeais, J. F., et al. (2019). On the resolutions of ocean altimetry maps. *Ocean. Sci.* 15, 1091–1109. doi: 10.5194/OS-15-1091-2019

Barth, A., Alvera-Azcárate, A., Licer, M., and Beckers, J. M. (2020). DINCAE 1.0: A convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations. *Geosci. Model. Dev.* 13, 1609–1622. doi: 10.5194/GMD-13-1609-2020

Carrassi, A., Bocquet, M., Bertino, L., and Evensen, G. (2018). Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *Wiley. Interdiscip. Rev.: Climate Change* 9, e535. doi: 10.1002/WCC.535

Charantonis, A. A., Testor, P., Mortier, L., D'Ortenzio, F., and Thiria, S. (2015). Completion of a sparse GLIDER database using multi-iterative self-organizing maps (ITCOMP SOM). *Proc. Comput. Sci.* 51, 2198–2206. doi: 10.1016/J.PROCS.2015.05.496

Chen, J., Li, Y., Luna, L. P., Chung, H. W., Rowe, S. P., Du, Y., et al. (2021). Learning fuzzy clustering for SPECT/CT segmentation via convolutional neural networks. *Med. Phys.* 48, 3860–3877. doi: 10.1002/MP.14903

Contractor, S., and Roughan, M. (2021). Efficacy of feedforward and LSTM neural networks at predicting and gap filling coastal ocean timeseries: oxygen, nutrients, and temperature. *Front. Mar. Sci.* 8. doi: 10.3389/FMARS.2021.637759/BIBTEX

Cutolo, E., Pascual, A., Ruiz, S., Johnston, T. S., Freilich, M., Mahadevan, A., et al. (2022). Diagnosing frontal dynamics from observations using a variational approach. *J. Geophys. Res.: Oceans.*, e2021JC018336. doi: 10.1029/2021JC018336

Durack, P. J., Gleckler, P. J., Landerer, F. W., and Taylor, K. E. (2014). Quantifying underestimates of long-term upper-ocean warming. *Nat. Climate Change* 4, 999–1005. doi: 10.1038/NCLIMATE2389

Fablet, R., Chapron, B., Drumetz, L., Mémin, E., Pannekoucke, O., and Rousseau, F. (2021). Learning variational data assimilation models and solvers. *J. Adv. Modeling. Earth Syst.* 13, e2021MS002572. doi: 10.1029/2021MS002572

Fablet, R., Drumetz, L., and Rousseau, F. (2020). Joint learning of variational representations and solvers for inverse problems with partially-observed data. doi: 10.48550/arxiv.2006.03653

Fresnay, S., Ponte, A. L., Le Gentil, S., and Le Sommer, J. (2018). Reconstruction of the 3-D dynamics from surface variables in a high-resolution simulation of North Atlantic. *J. Geophys. Res.: Oceans.* 123, 1612–1630. doi: 10.1002/2017JC013400

Gandin, L. S. (1966). Objective analysis of meteorological fields. Translated from the Russian. Jerusalem (Israel Program for Scientific Translations), 1965. Pp. vi, 242: 53 Figures; 28 Tables. £4 1s. 0d. *Q. J. R. Meteorol. Soc.* 92, 447–447. doi: 10.1002/QJ.49709239320

Gaspari, G., and Cohn, S. E. (1999). Construction of correlation functions in two and three dimensions. *Q. J. R. Meteorol. Soc.* 125, 723–757. doi: 10.1002/qj.49712555417

Gomis, D., Ruiz, S., and Pedder, M. A. (2001). Diagnostic analysis of the 3D ageostrophic circulation from a multivariate spatial interpolation of CTD and ADCP data. *Deep. Sea. Res. Part I.: Oceanogr. Res. Papers.* 48, 269–295. doi: 10.1016/S0967-0637(00)00060-1

Gueye, M. B., Niang, A., Arnault, S., Thiria, S., and Crépon, M. (2014). Neural approach to inverting complex system: Application to ocean salinity profile estimation from surface parameters. *Comput. Geosci.* 72, 201–209. doi: 10.1016/J.CAGEO.2014.07.012

Gurvan, M., Bourdallé-Badie, R., Chanut, J., Clementi, E., Coward, A., Ethé, C., et al. (2022). NEMO ocean engine. *Tech. Rep*. doi: 10.5281/ZENODO.6334656

Hernandez-Lasheras, J., and Mourre, B. (2018). Dense ctd survey versus glider fleet sampling: comparing data assimilation performance in a regional ocean model west of sardinia. *Ocean. Sci.* 14, 1069–1084. doi: 10.5194/os-14-1069-2018

Heslop, E. E., Ruiz, S., Allen, J., López-Jurado, J. L., Renault, L., and Tintoré, J. (2012). Autonomous underwater gliders monitoring variability at "choke points" in our ocean system: A case study in the Western Mediterranean Sea. *Geophys. Res. Lett.* 39. doi: 10.1029/2012GL053717

Jiang, F., Ma, J., Wang, B., Shen, F., and Yuan, L. (2021). Ocean observation data prediction for argo data quality control using deep bidirectional LSTM network. *Secur. Commun. Networks* 2021. doi: 10.1155/2021/5665386

Manucharyan, G. E., Siegelman, L., and Klein, P. (2021). A deep learning approach to spatiotemporal sea surface height interpolation and estimation of deep currents in geostrophic ocean turbulence. *J. Adv. Modeling. Earth Syst.* 13, e2019MS001965. doi: 10.1029/2019MS001965

Metref, S., Cosme, E., Le Guillou, F., Le Sommer, J., Brankart, J. M., and Verron, J. (2020). Wide-swath altimetric satellite data assimilation with correlated-error reduction. *Front. Mar. Sci.* 6. doi: 10.3389/FMARS.2019.00822/BIBTEX

Metref, S., Cosme, E., Le Sommer, J., Poel, N., Brankart, J. M., Verron, J., et al. (2019). Reduction of spatially structured errors in wide-swath altimetric satellite data using data assimilation. *Remote Sens.* 11, 1336. doi: 10.3390/RS11111336

Miyoshi, T., and Kondo, K. (2013). A multi-scale localization approach to an ensemble kalman filter. *SOLA* 9, 170–173. doi: 10.2151/SOLA.2013-038

Morrow, R., Fu, L. L., Ardhuin, F., Benkiran, M., Chapron, B., Cosme, E., et al. (2019). Global observations of fine-scale ocean surface topography with the Surface Water and Ocean Topography (SWOT) Mission. *Front. Mar. Sci.* 6. doi: 10.3389/FMARS.2019.00232/BIBTEX

Mourre, B., De Mey, P., Lyard, F., and Le Provost, C. (2004). Assimilation of sea level data over continental shelves: an ensemble method for the exploration of model errors due to uncertainties in bathymetry. *Dynamics. Atmospheres. Oceans.* 38, 93–121. doi: 10.1016/J.DYNATMOCE.2004.09.001

Pascual, A., Macías, D., Tintoré, J., Turiel, A., Ballabrera-Poy, J., Castro, C. G., et al. (2021). *White Paper 13: Ocean science challenges for 2030* (Consejo Superior de Investigaciones Científicas (España).

Pascual, A., Ruiz, S., Olita, A., Troupin, C., Claret, M., Casas, B., et al. (2017). A multiplatform experiment to unravel meso- and submesoscale processes in an intense front (AlborEx). *Front. Mar. Sci.* 4. doi: 10.3389/FMARS.2017.00039/BIBTEX

Pauthenet, E., Bachelot, L., Balem, K., Maze, G., Tréguier, A.-M., Roquet, F., et al. (2022). Fourdimensional temperature, salinity and mixed-layer depth in the Gulf Stream, reconstructed from remotesensing and in *situ* observations with neural networks. *Ocean. Sci.* 18, 1221–1244. doi: 10.5194/OS-18-1221-2022

Ruiz, S., Claret, M., Pascual, A., Olita, A., Troupin, C., Capet, A., et al. (2019). Effects of oceanic mesoscale and submesoscale frontal processes on the vertical transport of phytoplankton. *J. Geophys. Res.: Oceans.* 124, 5999–6014. doi: 10.1029/2019JC015034

Ruiz, S., Pascual, A., Garau, B., Faugère, Y., Alvarez, A., and Tintoré, J. (2009). Mesoscale dynamics of the Balearic Front, integrating glider, ship and satellite data. *J. Mar. Syst.* 78, S3–S16. doi: 10.1016/J.JMARSYS.2009.01.007

Ryabinin, V., Barbière, J., Haugan, P., Kullenberg, G., Smith, N., McLean, C., et al. (2019). The UN decade of ocean science for sustainable development. *Front. Mar. Sci.* 6, 470. doi: 10.3389/fmars.2019.00470

Sammartino, M., Nardelli, B. B., Marullo, S., and Santoleri, R. (2020). An artificial neural network to infer the mediterranean 3D chlorophyll-a and temperature fields from remote sensing observations. *Remote Sens.* 12, 4123. doi: 10.3390/RS12244123

Siegelman, L., Klein, P., Rivière, P., Thompson, A. F., Torres, H. S., Flexas, M., et al. (2019). Enhanced upward heat transport at deep submesoscale ocean fronts. *Nat. Geosci.* 13, 50–55. doi: 10.1038/s41561-019-0489-1

Troupin, C., Beltran, J. P., Heslop, E., Torner, M., Garau, B., Allen, J., et al. (2015). A toolbox for glider data processing and management. *Methods Oceanogr.* 13-14, 13–23. doi: 10.1016/J.MIO.2016.01.001

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst.* 2017-December, 5999–6009. doi: 10.48550/arxiv.1706.03762

Volpe, G., Colella, S., Brando, V. E., Forneris, V., La Padula, F., Di Cicco, A., et al. (2019). Mediterranean ocean colour Level 3 operational multi-sensor processing. *Ocean. Sci.* 15, 127–146. doi: 10.5194/OS-15-127-2019

Wang, G., Cheng, L., Abraham, J., and Li, C. (2018). Consensuses and discrepancies of basin-scale ocean heat content changes in different ocean analyses. *Climate Dynamics.* 50, 2471–2487. doi: 10.1007/S00382-017-3751-5/FIGURES/13

Welch, G., and Bishop, G. (1995). *An introduction to the Kalman Filter. Tech. Rep* (Chapel Hill, NC, USA: University of North Carolina at Chapel Hill), 95–041.

Zarokanellos, N. D., Rudnick, D. L., Garcia-Jove, M., Mourre, B., Ruiz, S., Pascual, A., et al. (2022). Frontal dynamics in the alboran sea: 1. Coherent 3D pathways at the almeria-oran front using underwater glider observations. *J. Geophys. Res.: Oceans.* 127, e2021JC017405. doi: 10.1029/2021JC017405

Zhu, X. X., Tuia, D., Mou, L., Xia, G. S., Zhang, L., Xu, F., et al. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Magazine.* 5, 8–36. doi: 10.1109/MGRS.2017.2762307

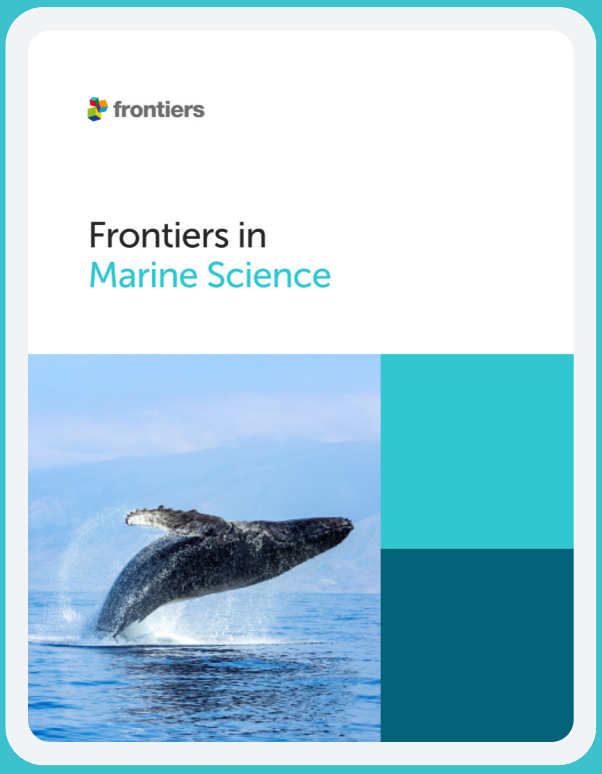
Frontiers in Marine Science

Explores ocean-based solutions for emerging global challenges

The third most-cited marine and freshwater biology journal, advancing our understanding of marine systems and addressing global challenges including overfishing, pollution, and climate change.

Discover the latest Research Topics

See more →



Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact