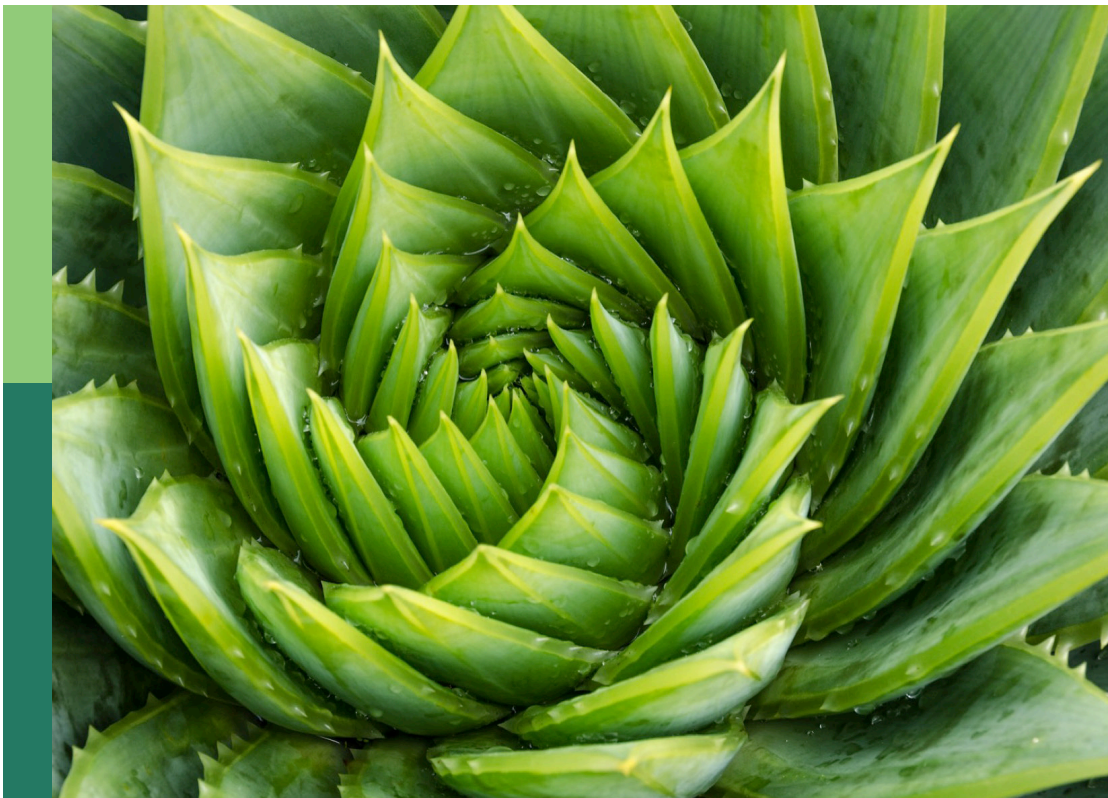
frontiers | Research Topics

Vision, learning, and robotics: AI for plants in the 2020s

Edited by
Zhenghong Yu, Huabing Zhou, Luca Iocchi, Jeffrey Too Chuan Tan, Changcai Yang and Hao Lu

Published in
Frontiers in Plant 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 978-2-8325-5891-1
DOI 10.3389/978-2-8325-5891-1

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

Vision, learning, and robotics: AI for plants in the 2020s

Topic editors

Zhenghong Yu — Guangdong Polytechnic of Science and Technology, China
Huabing Zhou — Wuhan Institute of Technology, China
Luca Iocchi — Sapienza University of Rome, Italy
Jeffrey Too Chuan Tan — Genovasi University College, Malaysia
Changcai Yang — Fujian Agriculture and Forestry University, China
Hao Lu — Huazhong University of Science and Technology, China

Citation

Yu, Z., Zhou, H., Iocchi, L., Tan, J. T. C., Yang, C., Lu, H., eds. (2025). *Vision, learning, and robotics: AI for plants in the 2020s*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-5891-1

Table of contents

05 Editorial: Vision, learning, and robotics: AI for plants in the 2020s
Zhenghong Yu, Luca Iocchi, Jeffrey Too Chuan Tan, Huabing Zhou, Changcai Yang and Hao Lu

08 CropQuant-Air: an AI-powered system to enable phenotypic analysis of yield- and performance-related traits using wheat canopy imagery collected by low-cost drones
Jiawei Chen, Jie Zhou, Qing Li, Hanghang Li, Yunpeng Xia, Robert Jackson, Gang Sun, Guodong Zhou, Greg Deakin, Dong Jiang and Ji Zhou

23 Wood identification of *Cyclobalanopsis* (Endl.) Oerst based on microscopic features and CTGAN-enhanced explainable machine learning models
Weihui Zhan, Bowen Chen, Xiaolian Wu, Zhen Yang, Che Lin, Jinguo Lin and Xin Guan

42 Real-time dense small object detection algorithm based on multi-modal tea shoots
Luyu Shuai, Ziao Chen, Zhiyong Li, Hongdan Li, Boda Zhang, Yuchao Wang and Jiong Mu

62 WeedNet-R: a sugar beet field weed detection algorithm based on enhanced RetinaNet and context semantic fusion
Zhiqiang Guo, Hui Hwang Goh, Xiuhua Li, Muqing Zhang and Yong Li

76 Overcoming field variability: unsupervised domain adaptation for enhanced crop-weed recognition in diverse farmlands
Talha Ilyas, Jonghoon Lee, Okjae Won, Yongchae Jeong and Hyongsuk Kim

90 Multi-index fuzzy comprehensive evaluation model with information entropy of alfalfa salt tolerance based on LiDAR data and hyperspectral image data
Jiaxin Zhang, Aiwu Zhang, Zixuan Liu, Wanting He and Shengyuan Yang

108 Deep learning for Chilean native flora classification: a comparative analysis
Carola Figueroa-Flores and Pablo San-Martin

121 High-throughput and separating-free phenotyping method for on-panicle rice grains based on deep learning
Yuwei Lu, Jinhu Wang, Ling Fu, Lejun Yu and Qian Liu

133 AC-UNet: an improved UNet-based method for stem and leaf segmentation in Betula luminifera
Xiaomei Yi, Jiaoping Wang, Peng Wu, Guoying Wang, Lufeng Mo, Xiongwei Lou, Hao Liang, Huahong Huang, Erpei Lin, Brian Tapiwanashe Maponde and Chaihui Lv

149 An improved YOLOv5s model for assessing apple graspability in automated harvesting scene
Huibin Li, Peng Yang, Huaiyang Liu, Xiang Liu, Jianping Qian, Qiangyi Yu, Changxing Geng and Yun Shi

164 Predicting resprouting of Platanus × hispanica following branch pruning by means of machine learning
Qiguan Shu, Hadi Yazdi, Thomas Rötzer and Ferdinand Ludwig

175 Identification of leaf diseases in field crops based on improved ShuffleNetV2
Hanmi Zhou, Jiageng Chen, Xiaoli Niu, Zhiguang Dai, Long Qin, Linshuang Ma, Jichen Li, Yumin Su and Qi Wu

192 Image recognition-based petal arrangement estimation
Tomoya Nakatani, Yuzuko Utsumi, Koichi Fujimoto, Masakazu Iwamura and Koichi Kise

206 Efficient residual network using hyperspectral images for corn variety identification
Xueyong Li, Mingjia Zhai, Liyuan Zheng, Ling Zhou, Xiwang Xie, Wenyi Zhao and Weidong Zhang

219 Cauli-Det: enhancing cauliflower disease detection with modified YOLOv8
Md. Sazid Uddin, Md. Khairul Alam Mazumder, Afrina Jannat Prity, M. F. Mridha, Sultan Alfarhood, Mejdl Safran and Dunren Che

235 Robust deep learning method for fruit decay detection and plant identification: enhancing food security and quality control
Pariya Afsharpour, Toktam Zoughi, Mahmood Deypir and Mohamad Javad Zoqi

251 Research on autonomous navigation system of greenhouse electric crawler tractor based on LiDAR
Huiping Guo, Yi Li, Hao Wang, Tingwei Wang, Linrui Rong, Haoyu Wang, Zihao Wang, Chensi Wang, Jiao Zhang, Yaobin Huo and Shaomeng Guo

266 Development of a machine vision-based weight prediction system of butterhead lettuce (*Lactuca sativa* L.) using deep learning models for industrial plant factory
Jung-Sun Gloria Kim, Seongje Moon, Junyoung Park, Taehyeong Kim and Soo Chung

281 How to discriminate wood of CITES-listed tree species from their look-alikes: using an attention mechanism with the ResNet model on an enhanced macroscopic image dataset
Shoujia Liu, Chang Zheng, Jiajun Wang, Yang Lu, Jie Yao, Zhiyuan Zou, Yafang Yin and Tuo He

295 BRA-YOLOv7: improvements on large leaf disease object detection using FasterNet and dual-level routing attention in YOLOv7
Rong Ye, Quan Gao and Tong Li

frontiers | Frontiers in Plant Science

TYPE Editorial
PUBLISHED 23 December 2024
DOI 10.3389/fpls.2024.1539626

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Ruslan Kalendar,
University of Helsinki, Finland

*CORRESPONDENCE
Zhenghong Yu
honger1983@gmail.com

RECEIVED 04 December 2024
ACCEPTED 12 December 2024
PUBLISHED 23 December 2024

CITATION
Yu Z, Iocchi L, Tan JTC, Zhou H, Yang C and Lu H (2024) Editorial: Vision, learning, and robotics: AI for plants in the 2020s. *Front. Plant Sci.* 15:1539626. doi: 10.3389/fpls.2024.1539626

COPYRIGHT
© 2024 Yu, Iocchi, Tan, Zhou, Yang and Lu. 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: Vision, learning, and robotics: AI for plants in the 2020s

Zhenghong Yu[1]*, Luca Iocchi[2], Jeffrey Too Chuan Tan[3], Huabing Zhou[4], Changcai Yang[5] and Hao Lu[6]

[1]Guangdong Polytechnic of Science and Technology, Zhuhai, China, [2]Sapienza University of Rome, Rome, Italy, [3]Genovasi University College, Petaling Jaya, Selangor, Malaysia, [4]Wuhan Institute of Technology, Wuhan, China, [5]Fujian Agriculture and Forestry University, Fuzhou, China, [6]Huazhong University of Science and Technology, Wuhan, China

KEYWORDS

artificial intelligence, machine learning, robotics, precision agriculture, disease detection, sustainable farming

Editorial on the Research Topic
Vision, learning, and robotics: AI for plants in the 2020s

Introduction

With the growth of the global population and increasing demand for food, agricultural production is under significant pressure. At the same time, climate change and resource constraints exacerbate these challenges, further heightening the need for sustainable agricultural practices. To address these complex issues, the field of plant science is undergoing a technological revolution. The rapid advancement of artificial intelligence (AI), computer vision, and robotics is redefining how plants are studied and agricultural practices are managed. From high-throughput phenotyping to precision agriculture and real-time monitoring, these technologies are dramatically improving efficiency and accuracy, laying a foundation for more resilient and sustainable agricultural systems. This Research Topic brings together pioneering studies to demonstrate how AI is advancing plant science and providing innovative solutions for modern agriculture.

Research contributions

The articles in this Research Topic showcase innovations across multiple fields. These contributions can be summarized into five key areas, each highlighting significant advancements in the study and application of plant science.

High-throughput phenotyping and crop monitoring

High-throughput phenotyping is a critical component of precision agriculture. By incorporating advanced deep learning models, researchers have significantly enhanced the efficiency and accuracy of crop phenotyping. For instance, Li et al. proposed a residual

network approach based on hyperspectral imaging, enabling rapid identification of corn varieties while adapting to varying growth conditions. This method not only improves prediction accuracy but also demonstrates the potential of hyperspectral data in agriculture. Additionally, the integration of RGB imaging with environmental variables broadens the scope of crop monitoring, driving the adoption of multimodal data fusion in agricultural applications.

Applications of robotics and automation in agriculture

Agricultural automation is transforming traditional farming practices. Guo et al. developed an autonomous navigation system for a greenhouse electric crawler tractor based on LiDAR, demonstrating its ability to navigate complex environments accurately. This system combines high-precision sensors with AI algorithms, reducing dependence on manual operation and significantly improving operational efficiency. Furthermore, solutions that integrate ground-based robots with drones have been applied to canopy imaging, weed detection, and disease monitoring, opening new avenues for smart farming.

Plant disease detection and management

Plant disease detection remains a critical area of agricultural research. Zhou et al. developed an improved ShuffleNetV2 model for rapid identification of field crop leaf diseases. This lightweight model maintains high accuracy while reducing computational requirements, making it well-suited for deployment in resource-constrained agricultural environments. Additionally, Ye et al. proposed enhancements to the YOLOv7 model for large-scale tea leaf disease detection in complex environments. The dual-level routing dynamic sparse attention mechanism employed significantly improves detection accuracy, offering robust support for precision agriculture.

Predicting plant growth and pruning behavior

Using machine learning to predict plant growth and pruning behavior provides new tools for agricultural decision-making. Shu et al. employed machine learning algorithms to predict the resprouting of Platanus × hispanica after pruning. This study not only reveals the relationship between pruning and plant growth but also offers practical guidance for forestry and horticulture. Moreover, multimodal modeling that integrates plant phenotypic data with environmental variables further enhances the accuracy of growth pattern predictions.

Food safety and quality monitoring of agricultural products

Improving food safety and quality is a primary goal of modern agricultural research. Afsharpour et al. proposed a robust deep learning method for detecting fruit decay and identifying plants. By integrating advanced image processing and classification algorithms, this method enables rapid identification of decayed fruit, improving efficiency and safety in food processing. Similarly, Kim et al. developed a machine vision-based weight prediction system for butterhead lettuce, providing an effective quality control tool for industrial agriculture.

Key trends and challenges

This Research Topic highlights several important trends while reflecting on persistent challenges in the field. Firstly, lightweight AI models for on-site deployment are becoming increasingly mainstream. These models maintain high accuracy despite limited computational resources, as demonstrated by Ye et al. Secondly, the rise of multimodal data fusion offers more comprehensive insights for phenotyping and health analysis, exemplified by the integration of RGB imaging and hyperspectral data by Li et al. However, the field continues to face challenges such as data scarcity, high equipment costs, and the complexity of model deployment. Addressing these challenges will require interdisciplinary collaboration, open-access datasets, and innovative engineering solutions.

Future directions

To advance plant science and achieve sustainable agriculture, future research should focus on the following directions: (1) Developing open-access datasets and affordable hardware to lower the barriers to AI adoption; (2) Optimizing lightweight models to enhance their robustness and scalability for smallholder farms and diverse agricultural environments; (3) Integrating satellite imagery, drones, and ground-based sensors to create a multi-layered crop monitoring system; (4) Exploring the long-term impacts of robotics and AI on agricultural ecosystems, particularly in terms of environmental sustainability and economic equity. These directions will provide new momentum for achieving precision agriculture.

Conclusion

This Research Topic demonstrates how artificial intelligence, machine learning, and robotics can address critical challenges in modern agriculture by enhancing efficiency and sustainability. The studies not only reveal diverse applications of these technologies in plant science but also lay a foundation for future agricultural innovations. As technology continues to evolve, these breakthroughs will offer new solutions for global food security and ecological conservation.

Author contributions

ZY: Writing – original draft, Writing – review & editing. LI: Conceptualization, Writing – review & editing. JT: Investigation, Writing – review & editing. HZ: Supervision, Writing – review & editing. CY: Validation, Writing – review & editing. HL: Supervision, 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.

The author(s) declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

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 Plant Science

TYPE Methods
PUBLISHED 19 June 2023
DOI 10.3389/fpls.2023.1219983

Check for updates

OPEN ACCESS

EDITED BY
Changcai Yang,
Fujian Agriculture and Forestry University, China

REVIEWED BY
Huabing Zhou,
Wuhan Institute of Technology, China
Zejun Zhang,
Zhejiang Normal University, China

*CORRESPONDENCE
Ji Zhou
Ji.Zhou@NJAU.edu.cn;
Ji.Zhou@NIAB.com

†These authors have contributed equally to this work

RECEIVED 09 May 2023
ACCEPTED 26 May 2023
PUBLISHED 19 June 2023

CITATION
Chen J, Zhou J, Li Q, Li H, Xia Y, Jackson R, Sun G, Zhou G, Deakin G, Jiang D and Zhou J (2023) CropQuant-Air: an AI-powered system to enable phenotypic analysis of yield- and performance-related traits using wheat canopy imagery collected by low-cost drones.
Front. Plant Sci. 14:1219983.
doi: 10.3389/fpls.2023.1219983

COPYRIGHT
© 2023 Chen, Zhou, Li, Li, Xia, Jackson, Sun, Zhou, Deakin, Jiang and Zhou. 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.

CropQuant-Air: an AI-powered system to enable phenotypic analysis of yield- and performance-related traits using wheat canopy imagery collected by low-cost drones

Jiawei Chen [1,2†], Jie Zhou [1,2†], Qing Li [3†], Hanghang Li [1], Yunpeng Xia [1], Robert Jackson [4], Gang Sun [1], Guodong Zhou [1], Greg Deakin [4], Dong Jiang [3] and Ji Zhou [1,4*]

[1]State Key Laboratory of Crop Genetics & Germplasm Enhancement, Academy for Advanced Interdisciplinary Studies, Nanjing Agricultural University, Nanjing, China, [2]College of Engineering, Nanjing Agricultural University, Nanjing, China, [3]Regional Technique Innovation Center for Wheat Production, Key Laboratory of Crop Physiology and Ecology in Southern China, Ministry of Agriculture, Nanjing Agricultural University, Nanjing, China, [4]Cambridge Crop Research, National Institute of Agricultural Botany (NIAB), Cambridge, United Kingdom

As one of the most consumed stable foods around the world, wheat plays a crucial role in ensuring global food security. The ability to quantify key yield components under complex field conditions can help breeders and researchers assess wheat's yield performance effectively. Nevertheless, it is still challenging to conduct large-scale phenotyping to analyse canopy-level wheat spikes and relevant performance traits, in the field and in an automated manner. Here, we present CropQuant-Air, an AI-powered software system that combines state-of-the-art deep learning (DL) models and image processing algorithms to enable the detection of wheat spikes and phenotypic analysis using wheat canopy images acquired by low-cost drones. The system includes the YOLACT-Plot model for plot segmentation, an optimised YOLOv7 model for quantifying the spike number per m^2 ($SNpM^2$) trait, and performance-related trait analysis using spectral and texture features at the canopy level. Besides using our labelled dataset for model training, we also employed the Global Wheat Head Detection dataset to incorporate varietal features into the DL models, facilitating us to perform reliable yield-based analysis from hundreds of varieties selected from main wheat production regions in China. Finally, we employed the $SNpM^2$ and performance traits to develop a yield classification model using the Extreme Gradient Boosting (XGBoost) ensemble and obtained significant positive correlations between the computational analysis results and manual scoring, indicating the reliability of CropQuant-Air. To ensure that our work could reach wider researchers, we created a graphical user interface for CropQuant-Air, so that non-expert users could readily use our work. We believe that our work

represents valuable advances in yield-based field phenotyping and phenotypic analysis, providing useful and reliable toolkits to enable breeders, researchers, growers, and farmers to assess crop-yield performance in a cost-effective approach.

KEYWORDS

wheat spike detection, drone phenotyping, key yield component, yield classification, open AI software

Introduction

Yield performance is in the heart of breeding, crop research and agricultural practices (Ferrante et al., 2017). The ability of reliably classifying and predicting yield production was key for plant researchers and breeders to understand crop yield performance under complex field conditions (Jin et al., 2017). Moreover, to be able to estimate yield production during the season could facilitate growers and farmers to make reliable decisions of agronomic management such as crop rotations, fertilisation, and irrigation, so that growing conditions for crops could be optimised to facilitate more accurate and sustainable agricultural practices (Reynolds et al., 2020).

In this study, we used wheat (*Triticum aestivum*) as our model plant, a key staple food in China and many countries around the world. Global wheat consumption reached 793 million tons in 2021/22 (Nduku et al., 2023), demonstrating the great significance to ensure its supply. Nevertheless, wheat yield production could be affected by many factors in the field, ranging from environmental factors to agronomic inputs (Yang et al., 2021). Hence, it is important to equip breeders and plant researchers with suitable toolkits, so that they could assess yield performance during the reproductive phase. To characterise wheat grain yield, key components such as spike number per unit area ($SNpM^2$), grain number per spike (GNpS), and thousand grain weight (TGW) were often utilised (Griffiths et al., 2015). The $SNpM^2$ trait was regarded as a key indicator to evaluate yield potential (Bastos et al., 2020). Breeders, crop researchers, growers and farmers often manually scored or statistically estimated this trait during field surveillance (Marza et al., 2006). However, traditional methods to quantify $SNpM^2$ in the field were not only laborious, but also prone to error (Qiu et al., 2019), leading to new approaches developed to address this challenge (Furbank et al., 2019). More importantly, due to the rapidly changing climates, breeding strategies were reformed towards the improvement of crops' climate resilience and sustainability, requiring more effective data collection and analytic tools to accelerate the process of characterising yield components (Bevan et al., 2017).

Unmanned aerial vehicles (UAVs) based plant phenotyping has been developed rapidly in the past decade (Jang et al., 2020). Due to the decreasing costs of drones and image sensors, the improvement of flight control software, and more powerful UAV-based analytic software introduced to the research field, many research groups integrated drone phenotyping into their field-based research activities (Yang et al., 2017). In order to study yield performance, a range of image sensors such as red-green-blue (RGB) cameras, multi- and hyper-spectral devices, Light Detection and Ranging (LiDAR), and thermal and infrared sensors (Kachamba et al., 2016; Gracia-Romero et al., 2017; ten Harkel et al., 2020) were utilised in drone phenotyping to acquire plant's morphological and spectral features, from which yield-related traits and proxies could be derived (Jiang et al., 2021). For example, AirSurf applied convolutional neural networks (CNNs) to analyse millions of lettuce heads collected by manned light aircrafts, so that marketable yield of lettuce production could be estimated (Bauer et al., 2019); multi-temporal vegetation indices derived from drone-collected multi-spectral and RGB imagery were employed to predict rice grain production, showing the drone-based phenotyping could be used to identify the optimal stage for carrying out yield prediction in rice (Zhou X. et al., 2017); deep CNNs were employed to estimate rice yield performance during ripening based on aerial imagery (Yang et al., 2019); multimodal data fusion and deep learning were integrated into the classification of yield production in soybean through drone-based field phenotyping (Maimaitijiang et al., 2017); CropQuant-3D utilised open-source 3D point clouds analysis algorithms to extract canopy-level yield-related traits (e.g. 3DCI) collected by light detection and ranging (LiDAR) or drones to identify resource use efficiency wheat varieties and their yield performance (Zhu et al., 2021); AirMeasurer combined computer vision and supervised machine learning (ML) to build dynamic phenotyping algorithms to analyse yield-related traits in rice (e.g. early establishment and heading date) based on 2D/3D aerial imagery, resulting in reliable loci identified to enable the exploration of new candidate genes (Sun et al., 2022).

Abbreviations: ASM, Angular Second Moment; DL, Deep Learning; ExG, Excess Green; GLCM, Gray-level Co-occurrence Matrix; $GPpM^2$, Grain Production per m^2; GNpS, grain number per spike; GUI, graphical user interface; GWHD, Global Wheat Head Detection; ML, Machine Learning; NDYI, Normalized Difference Yellowness Index; R-CNN, Region-based Convolutional Neural Network; ROI, Region of Interest; RMSE, Root Mean Square Error; RGB, red-green-blue; $SNpM^2$, spike number per unit area; SOLO, Segmenting Objects by Locations; TGW, thousand grain weight; UAV, unmanned aerial vehicle; VARI, Visible Atmospheric Resistance Index; XGBoost, eXtreme Gradient Boosting; YOLACT, You Only Look At CoefficienTs; YOLO, You Only Look Once.

The above research made valuable progresses in yield-based aerial phenotyping. Still, much research aimed to establish relationships between physiological parameters (e.g. vegetation indices and canopy structural features) with yield production, which were useful proxies but did not provide a direct yield-based measure (Gizaw et al., 2018). Due to the rapid advances in vision-based artificial intelligence (AI) and deep learning (DL), AI-powered techniques such as object detection, classification, semantic segmentation, and pattern recognition opened a new door for yield-based trait analysis (Wang et al., 2020). For example, SpikeSegNet (Misra et al., 2020) used an encoder-decoder with hourglass architecture to detect wheat spike signals for indoor experiments; DeepCount (Sadeghi-Tehran et al., 2019) combined simple linear iterative clustering and deep CNNs to identify wheat spikes in the field, indicating the feasibility of applying DL to detect spike-like objects but was limited in scale and varieties; by tilting camera angles, DL models were trained to count wheat spikes and estimate yield when spike density was low (Hasan et al., 2018); fully convolutional network (FCN) and transfer learning were employed to perform semantic segmentation of wheat spike regions using time series collected by CropQuant workstations (Zhou J. et al., 2017; Alkhudaydi et al., 2019), demonstrating the usefulness of AI-powered trait analysis of key yield components.

The above methods verified that DL-based approaches could bring unique values to the detection of key yield components such as wheat spikes under field conditions. Nevertheless, due to the complex field conditions and the large-scale nature of field trials, ground-based stationed phenotyping devices were rather limited if hundreds of plots needed to be examined. Consequently, drone-based field phenotyping was likely to bridge the gap between accuracy and scalability in yield-based studies (Yang et al., 2017). While the above ML/DL methods advanced our capability in detecting wheat spikes, the generality and scalability of them needed to be improved due to diverse wheat spike morphologies (e.g. awned, awnless, long and short spikes) and recent advances in vision-based AI research (Patrício and Rieder, 2018). In fact, besides morphological and spectral features, semantic information should also be considered in the detection of wheat spikes from the canopy (Sadeghi-Tehran et al., 2019). Hence, domain knowledge such as wheat spike developmental (e.g. key growth stages) features and plot- and organ-level morphological features should be taken into consideration when building DL models (Zhou et al., 2022), so that wheat spikes could be identified reliably under field conditions, including colour changes caused by changing natural illuminance, clustered or sparse spikes due to dissimilar growth paces, or canopy-level spike occlusion during the reproductive phase.

Our work, CropQuant-Air, presents an open and AI-powered software system that combined state-of-the-art DL techniques into the detection of wheat spikes under complex field conditions. The CropQuant-Air system first integrated the YOLACT-Plot model, a DL model based on the YOLACT++ model (Angeles Ceron et al., 2021), to enable the automated plot segmentation based on drone-collected wheat canopy image series. Within the segmented plots, an optimised YOLOv7 model (Wang et al., 2022) was trained using our labelled spikes together with the Global Wheat Head Detection (GWHD) dataset (David et al., 2020), which was employed to perform spike detection and quantify the $SNpM^2$ trait. Moreover, we have included a range of image processing algorithms in CropQuant-Air to conduct canopy-level trait analysis using spectral and textural features possessed by the aerial images. To verify the CropQuant-Air system, we applied it to a field experiment studying 210 wheat varieties (two replicates; 420 plots) selected from main wheat production regions in China. Besides the analysis of the $SNpM^2$ and canopy-level traits, we also developed the Extreme Gradient Boosting (XGBoost) ensemble (Chen and Guestrin, 2016) to classify yield groups using the quantified yield- and performance-related traits. After that, we performed correlation analysis between the computational analysis results and manual scoring on target traits and obtained significant positive correlations, indicating the reliability of the CropQuant-Air system in AI-powered phenotypic analysis. Finally, to ensure that our work could reach the broader plant research community, we created a graphical user interface (GUI) for CropQuant-Air, so that non-expert users could readily use our work for their yield- and performance-related trait analysis.

Materials and methods

Plant materials and field experiments

In order to verify trait analysis results generated by CropQuant-Air, we selected 210 winter wheat varieties cultivated for several main wheat production regions in China including Jiangsu (East China), Shandong (North China), and Henan (Central China). These lines were known for their dissimilar yield performance and different spike morphologies (Betts et al., 2014), which were suitable for building generalised DL models to incorporate dissimilar phenotypic variation into the AI-powered plot and spike detection based on canopy-level wheat images (see Table S1 in Supplementary Material for the list).

The field experiment was conducted at the Nanjing Agricultural University's Baimai field trial center (Nanjing China; 31°36'57.8" N, 119°10'46.1" E; red coloured 5-point star, Figure 1A), just below 0.6 hectares (ha) in size. During the 2021/22 growing season, the 210 wheat varieties (two replications; dark blue and dark red shading areas; Figure 1B) were sown in 1.5 × 1.5 m plots, with 20 cm spacing, 5 rows per plot, and around 450 plants per plot. At the end of the season, we threshed and weighted the dry grains to measure grain production per plot (in kg), grain production per m^2 ($GPpM^2$), and thousand grain weight (TGW, in kg). The yield data was manually classified into three groups (i.e. high, medium, and low) according to protocols previously published (Leilah and Al-Khateeb, 2005), which were also used as groundtruthing when verifying the yield classification model.

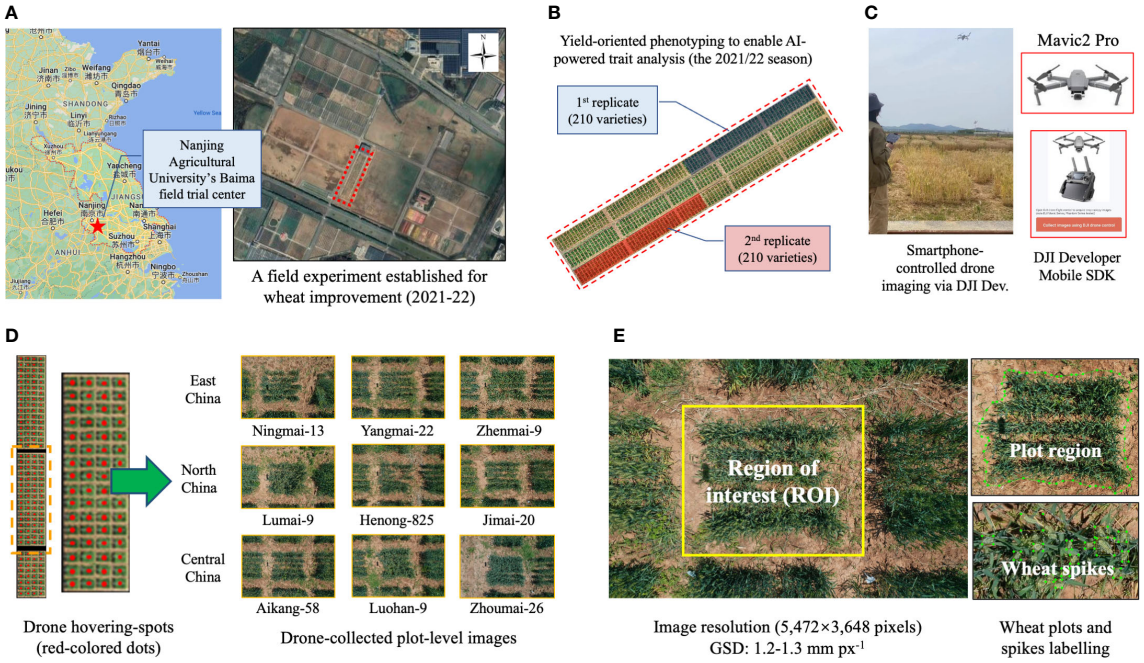


FIGURE 1
The field experiment and plot-based images acquired by low-cost drones. **(A, B)** Geo-location of the trial centre, where 210 wheat varieties were studied; the varieties were selected from three main wheat production regions in China. **(C, D)** The aerial phenotyping using a low-cost drone and wheat canopy imagery acquired from an overhead perspective, with some representative varieties listed. **(E)** The image resolution of the collected aerial images together with plot- and spike-level image annotation.

Drone-based phenotyping

As we focused on collecting canopy-level features using low-cost drone phenotyping, we therefore developed a smartphone-based flight control function based on the DJI developer Mobile SDK (DJI, Shenzhen, China), which could control small drones (i.e. Mavic2 Pro, equipped with an high-definition RGB camera, with a maximum image resolution of 5,472 × 3,648 pixels). After verifying the image resolution for AI-powered detection as well as the speed of drone phenotyping, we chose to fly the drone to image wheat canopy from an overhead perspective at a 4-m altitude during late flowering and early grain filling stages, so that we could accomplish the aerial imaging within 30 minutes and with AI-compatible wheat imagery (Figure 1C). Also, due to the aviation regulation in China, we manually flew the drone via an Android smartphone to hover at fixed spots directly above target plots and imaging was conducted using an auto-ISO mode with a fast fixed shutter speed (Figure 1D, left). Two series of images were generated during the phenotyping, consisted of 210 aerial images, representing the 210 winter wheat varieties' morphological and spectral properties at the canopy level (Figure 1D, right). After each flight, the acquired images were transferred to a cloud server (Baidu Netdisk, Beijing, China) to enable different project partners to review and pre-process. Some testing files were uploaded to our GitHub repository for academic research and development (R&D) activities.

The training dataset for AI-powered trait analysis

AI-powered plant phenomics research heavily relied on high-quality labelled data. To extract wheat plots (i.e. regions of interest, ROIs; Figure 1E, left) from acquired aerial images effectively, we first employed the Labelme toolkit (Torralba et al., 2010) to label plot outlines (480 labelled plots in total, in the COCO2017 format); then, we applied the Labelimg tool (Yu et al., 2019) to annotate wheat spikes within the ROIs (212,596 spikes in total, stored in the PASCAL VOC format). Some of the labelled plot- and spike-level datasets (Figure 1E, right) was also uploaded to the GitHub repository.

The wheat-plot training dataset was used to enable the AI-powered plot detection, so that the central plot region in a given aerial image could be reliably identified (Figure 2A). To improve the wheat-plot training dataset in terms of unevenly distributed samples (i.e. dissimilar variety numbers from the three wheat production regions), we applied image augmentation techniques to enhance the dataset, including techniques such as luminance enhancement, random rotation, pretzel noise, and mosaics (Figure 2B; middle), resulting in a total of 1,920 annotated plots, which were divided into training (1,351 images; 70%) and testing (579 images, 30%) sets.

The wheat spike annotation was conducted by three specialists, whose labels were combined as the wheat-spike training dataset.

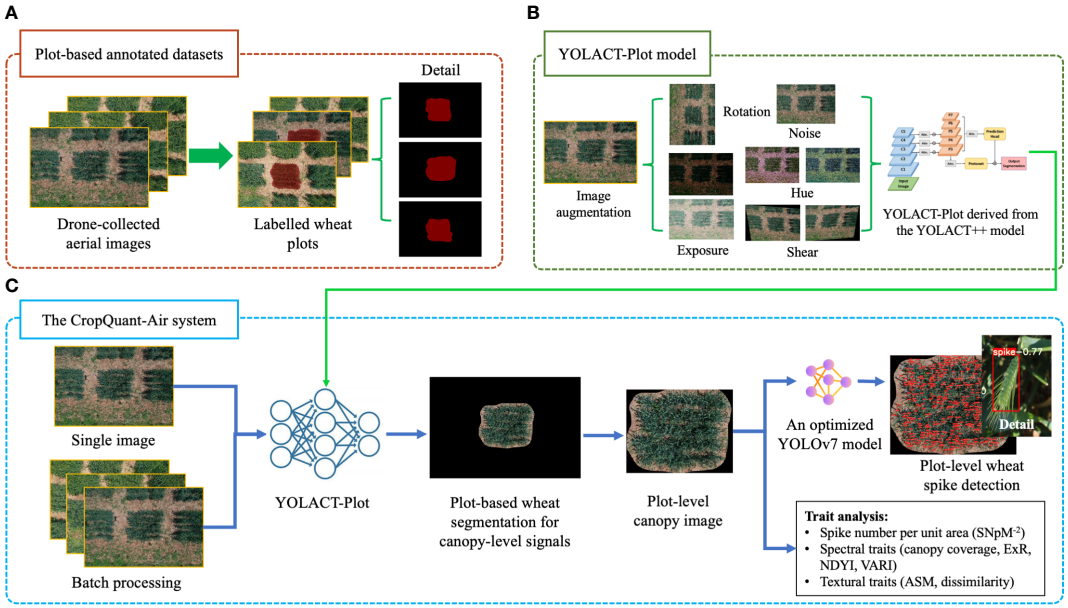


FIGURE 2
The analysis workflow of the CropQuant-Air system established for extracting plot-level phenotypic traits from drone-collected wheat canopy images. **(A)** The establishment of wheat-plot training data to identify wheat plot region using the Labelme toolkit. **(B)** Image augmentation applied to enhance the wheat-plot training dataset. **(C)** The CropQuant-Air system developed for processing drone-collected wheat images and quantifying phenotypic traits based on morphological and spectral signals.

Besides our own imagery, we utilised training datasets previously published, i.e. the GWHD dataset (collected by nine organisations, covering genotypes from Western Europe, North America, Australia and East Asia), where an open and diverse dataset of wheat spikes were labelled from RGB images for developing and benchmarking ML/DL models. In order to improve the generalisation and accuracy of our AI-powered wheat spike detection algorithm, we combined GWHD and our wheat-spike training data with image augmentation techniques to train the YOLOv7 (Wang et al., 2022) based wheat spike detection model.

The algorithmic workflow of phenotypic analysis

We developed a three-step algorithmic workflow to incorporate computer vision and DL algorithms into the plot-based trait analysis: (1) a YOLACT-Plot segmentation algorithm (Figure 2B; right) was trained to enable the detection of wheat plots in aerial images; (2) then, canopy-level signals within the segmented plot were analysed by computer vision and DL-based object detection algorithms, resulting in the measurement of a range of phenotypic features, including plot-level wheat spikes, spectral (i.e. excess red vegetation index, ExR; normalised difference yellowness index, NDYI; visible atmospherically resistant index, VARI) and textural traits (i.e. canopy coverage; angular second moment, ASM; greyscale co-occurrence matrices, GLCM dissimilarity); (3) finally, plot-based trait analysis results (in CSV) and processed images (i.e. plot region segmentation and plot-level spike detection; in JPG) were produced and downloadable via the CropQuant-Air system (Figure 2C). When calculating spectral traits, we followed the approach that was developed for RGB-sensor-based trait analysis without radiation calibration (Svensgaard et al., 2021).

The YOLACT-Plot segmentation model

The YOLACT++ model was an enhanced fully-convolutional model built for real-time instance segmentation. We adopted its learning architecture and built the YOLACT-Plot model to identify wheat plots within aerial images. The YOLACT++ network was composed of a ResNet101 backbone network to extract features from input images, generating five feature maps (i.e. from C1 to C5; Figure 3E). Following the standard architecture, we utilised the C3-C5 feature maps (red coloured numbered circles 1-3) as input layers of the feature pyramid, which were fused to produce five sub-feature maps (P3 to P7) at different scales (Figure 3B). The ResNet-101-based backbone network in the YOLACT++ employed the Bottleneck Residual structure as the fundamental module to enhance feature extraction and address the gradient vanishing problem (Wu et al., 2019). In our case, different sizes and shapes of plots needed to be detected under complex field conditions. Hence, we optimised the model by replacing the Bottleneck module with a modified Res2Net module (Gao et al., 2021), which facilitated the extraction of deeper and high-level features contained in a single layer. The Res2Net module consisted of four feature sub-graphs that had the same spatial size and channels, whose output was convolved 3×3 with the previous feature sub-graph as its input. Finally, the outputs of the four feature sub-graphs were combined via a 1×1 convolution, enabling the reuse of features to help us expand the

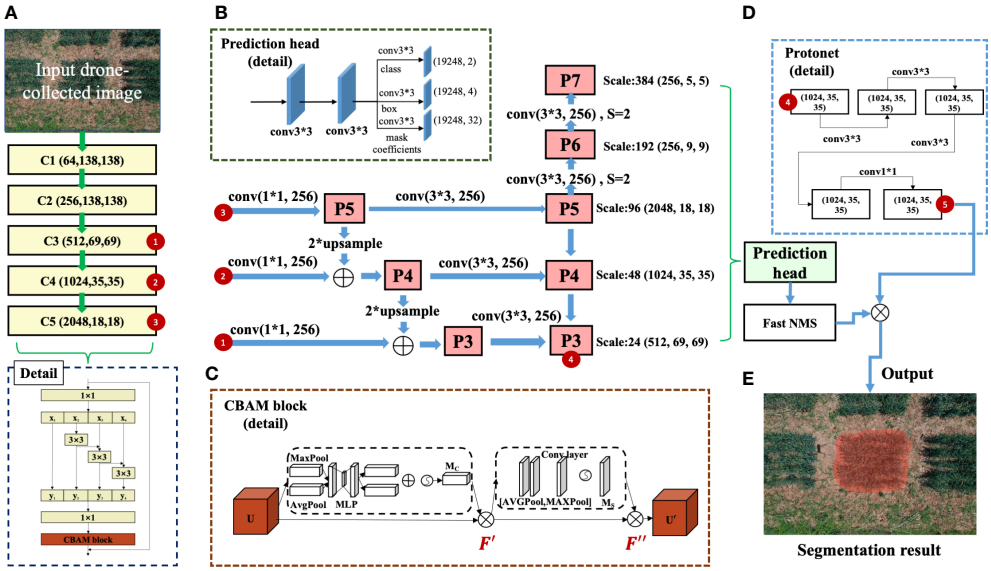


FIGURE 3
The learning architecture of the YOLACT-Plot model, which was built on the YOLACT++ model together with wheat plot training data. **(A)** The backbone architecture of the improved YOLACT-Plot model. **(B, C)** The architecture with improved Res2Net together with the prediction head as well as the CBAM block with attention mechanism. **(D, E)** The Protonet block and the plot segmentation result produced by the learning model.

perceptual domain to facilitate the extraction of both global and local information (Figure 3A; below).

Additionally, to further the enhancement of feature extraction in the YOLACT-Plot model so that invalid or irrelevant features could be suppressed, we added a Convolutional Block Attention Module (CBAM) to the Res2Net module (Woo et al., 2018). The CBAM block (Figure 3C) had two components: (1) channel attention and (2) spatial attention, which could generate weights of one-dimensional channel attention from the feature map as well as two-dimensional spatial attention. The channel attention component obtained the input feature map U (in H × W × C format, C represents the number of feature map channels) through global maximum pooling and then average pooling, producing two 1 × 1 × C feature maps, both of which were fed into a two-layer perceptron with shared weights. The output features were summed and activated to produce channel attention weights M_c (Eqns. 1-3). Finally, the input feature map and the channel attention weights were multiplied to generate the channel attention feature map F' (Eqn. 4), enabling the model to focus both high- and low-level features of target plots and hence the improvement of YOLACT-Plot's feature extraction.

$$Z_{avg} = \text{AVG}Pool(\text{U}) \quad \text{(Eqn. 1)}$$

$$Z_{max} = \text{MAX}Pool(\text{U}) \quad \text{(Eqn. 2)}$$

$$M_c = \sigma[f_2(\delta(f_1(Z_{avg}))) + f_2(\delta(f_1(Z_{max})))] \quad \text{(Eqn. 3)}$$

$$F' = M_c \times U \quad \text{(Eqn. 4)}$$

where Avg*Pool* denotes the global average pooling, averaging the pixel-based intensity values of each channel; MaxPool denotes the global maximum pooling, preserving the maximum intensity value of each channel's feature map; f_1 represents the fully connected layer of an input channel C for the CBAM and output channel C/16; f_2 represents the fully connected layer of input channel C/16 and output channel; δ signifies a rectified linear unit (ReLU) function; σ signifies the Sigmoid function.

The spatial attention component used the input feature map F' through the average pooling and maximum pooling of channel dimensions to obtain two H × W × 1 feature maps, which were combined and then passed through a 7 × 7 convolution layer, followed by the Sigmoid function to obtain the spatial attention weights M_c (Eqns. 5-7) from a single channel in two dimensions. Finally, multiplying the spatial attention weights with the channel attention feature map resulted in a new feature map F'', with both spatial and the channel attention features (Eqn. 8).

$$Z_{AVG} = \text{AVGPool}(\text{U}) \quad \text{(Eqn. 5)}$$

$$Z_{MAX} = \text{MAXPool}(\text{U}) \quad \text{(Eqn. 6)}$$

$$M_c = \sigma[f_3(Z_{AVG}, Z_{MAX})]] \quad \text{(Eqn. 7)}$$

$$F'' = M_s \times F' \quad \text{(Eqn. 8)}$$

where AVGPool denotes channel dimensional averaging pooling, where pixel values corresponding to each channel's feature map are summed and averaged; MAXPool denotes maximum pooling of channel dimensions, where the maximum pixel intensity value of teach channel's feature map are retained; f_3 represents a convolution of size 7 × 7, with an output channel of 1; σ signifies the Sigmoid function.

Building on the refined feature maps (C1-C5) and the above attention mechanism, P3-P7 sub-feature maps were fed into two

parallel branches to perform plot detection: (1) the ProtoNet branch (square-dotted rectangle; Figure 3C, upper), which generated *k*-prototype masks with varying regional responses from the P3 feature map (red coloured numbered circle 4 in the square-dotted rectangle; Figure 3D upper); (2) the prediction head (square-dotted rectangle; Figure 3B upper left), which produced anchor frames with varied aspect ratios (i.e. 1, 1/2, 2, 1/3, 3) that employed pixel-based points of the output feature map as anchor points to detect instances, followed by an anchor-frame classification and the coefficient prediction. Both Fast Non-maximum Suppression (NMS) filters and Bounding Box Regression were applied to screen all the candidate detection boxes, resulting in instance prediction that was linearly combined with the prototype mask coefficients and hence the final mask obtained after the auto-thresholding (Figure 3E).

YOLOv7 for wheat spike detection

We combined the annotated wheat spikes in GWHD and our annotated wheat spikes to train a detection model based on YOLOv7 as the baseline model. YOLOv7 was an efficient and accurate object detector that was suitable for detecting small objects in regions with dense target objects. Particularly, we chose YOLOv7 to detect canopy-level wheat spikes when they were densely clustered, occluded, or under varied nature illuminance because the YOLO-based model was suitable for recognising objects in crowded scenes in a high-throughput and high-accuracy manner (Chen et al., 2021). As a result, we chose the Standard version of YOLOv7 due to the easiness of the software deployment and reasonable computational cost. The detailed implementation of other versions of YOLOv7 (e.g. Tiny and W6) can be found via https://github.com/WongKinYiu/yolov7.

Model training strategies

We built a workstation (Intel Core i7-8700 CPU, Nvidia 3060 GPU, and 64 GB RAM) to conduct the DL training. Both PyTorch (v1.11) framework (Paszke et al., 2019) and Python (v3.8) were utilised to implement and train the above two DL models, one for plot segmentation and one for wheat spike detection. During the training, an input images were first resized (550 × 550 pixels) and then trained with optimised parameters (e.g. batch size = 8; stochastic gradient descent (SGD) momentum = 0.9; learning rate = 0.001; epochs = 500). The loss value was used as an evaluation metric to quantify the deviation between the predicted results and the labelled data (i.e. the lower the loss value, the higher the prediction accuracy) through each iteration during training. We applied the binary cross-entropy as the loss function. When the number of iterations increased, the loss value decreased, indicating an improved performance. The loss value of the YOLACT-Plot model stabilised at approximately 0.52 after 800 iterations, whereas the optimised YOLOv7 model stabilised at around 0.22 after 1,000 iterations.

Model evaluation

We selected average precision (AP) as the performance metric to evaluate the segmentation result. AP was calculated using Eqns. 9-11. To ensure a comprehensive evaluation of the performance of the DL models, AP50 (intersection over union, IoU = 0.5), AP75 (IoU = 0.75), and the mean Average Precision (mAP) were employed since the selection of IoU could influence the precision and recall scores.

$$P = \frac{TP}{TP + FP} \quad \text{(Eqn. 9)}$$

$$R = \frac{TP}{TP + FN} \quad \text{(Eqn. 10)}$$

$$\mathrm{AP} = \int_0^1 P(R)dR \quad \text{(Eqn. 11)}$$

where *TP* denotes the number of samples where the predicted category of the model matches the true labelled category; *FP* indicates the number of samples where the predicted category does not match the true labelled category; *FN* denotes the number of samples where the predicted category is the background, but the true labelled category are other categories.

The complexity of the learning model was evaluated using the number of parameters and floating-point calculations (*FLOPs*). Since the activation functions and biases affect the calculation of *FLOPs*, different calculations were performed. To ensure consistency of our analysis, we used PyTorch's third-party library, Thop (Jian et al., 2022), to calculate model parameters and *FLOPs*. For a single convolution operation, the model parameters and *FLOPs* were calculated using the equations below:

$$Parameter = K^2 \times C_{in} \times C_{out} + C_{out} \quad \text{(Eqn. 12)}$$

$$FLOPs = C_{in} \times K^2 \times C_{out} \times H \times W \quad \text{(Eqn. 13)}$$

where C_{in} and C_{out} are the number of input and output feature map channels; K denotes the size of the convolution kernel; $H \times W$ stands for the size of the output feature map.

GUI design and software implementation

Finally, to ensure that our AI-powered phenotypic analysis system could reach the broader research community, we created a graphical user interface (GUI) for nonexperts. The CropQuant-Air GUI followed a modular architecture and implemented using the Python programming language. The cross-platform GUI (in EXE) integrated the above trained AI models and trait analysis algorithms into a stepwise software system and was developed using the Tkinter library (Shipman, 2013). To implement phenotypic analysis and ML/DL libraries, we employed open-source libraries such as SciPy (Virtanen et al., 2020) for scientific data processing, OpenCV (Bradski and Kaehler, 2008) for image analysis, Scikit-Learn (Pedregosa et al., 2011) for yield classification modelling, and the AirMeasurer libraries (Sun et al., 2022) for developing phenotypic trait analysis.

Yield classification model and statical analysis

In order to train a ML-based model to classify yield production, we used the seven traits produced by the CropQuant-Air (i.e. $SNpM^2$, NDYI, ExR, VARI, canopy coverage, ASM and GLCM dissimilarity) as input parameters and the manually measured yield production (i.e. $GPpM^2$ and TGW) as targets. After testing a range of supervised ML models, we chose a stochastic gradient boosting algorithm, extreme gradient boosting (XGBoost) ensemble, as it provided the best performance compared with other supervised ML classifiers. When training the model, we fine-tuned the hyperparameters of the XGBoost model, including the number of trees, tree depth, learning rate, the number of samples, and the number of features. A grid-based search was employed to fine-tune the model to yield optimised hyper-parameters, followed by a combination of parameters with reasonable ranges of parametric values to simplify the procedure. Finally, *k*-fold cross-validation was adopted to evaluate the model performance, based on which the model with the best accuracy was selected. The above algorithmic steps and software implementation were performed using the Scikit-Learn library and saved in a separate executable Jupyter notebook, which can be downloaded from our GitHub repository.

Results

Datasets collected from the study

We collected two series of aerial images from the field experiment between booting and early grain filling using the low-cost drone, over 15.3 GB in total. The plot- and spike-training datasets were built using these images, including 420 annotated plots, 212,596 labelled wheat spikes, and augmented sub-images (2,940 for plots and 1,488,172 for spikes), covering the 210 varieties. During post-harvest handling, plot-based yield production, $GPpM^2$ and TGW were manually quantified from the 420 plots, which were randomly divided into 70% and 30% datasets for training, testing and validation to build the yield classification model.

Plot segmentation using the YOLACT-Plot model

We applied the YOLACT-Plot model to identify the central plot in a given aerial image (Figure 4A; first column). To evaluate the impacts of different components or hyper-parameters on the plot detection, we conducted an ablation study (Meyes et al., 2019), which compared different components in the model and identified the essential factors for the plot detection. Three sets of experiments were accomplished (Table 1). The 1st and 2nd experiments suggested that the introduction of the Res2Net module in the backbone network improved the mAP of the prediction frame (2.35%) and mask (1.76%); whereas the 2nd and 3rd experiments indicated that the Res2Net-CBAM module improved the mAP of the prediction further (1.89% and 0.85%, respectively), with slightly decrease in speed (i.e. 1 frame per second, FPS).

With an improved backbone network, features from different channels of the same feature layer were combined for multiple times, facilitating the extraction of semantic information while leading to an increase in the size and parameters of the model.

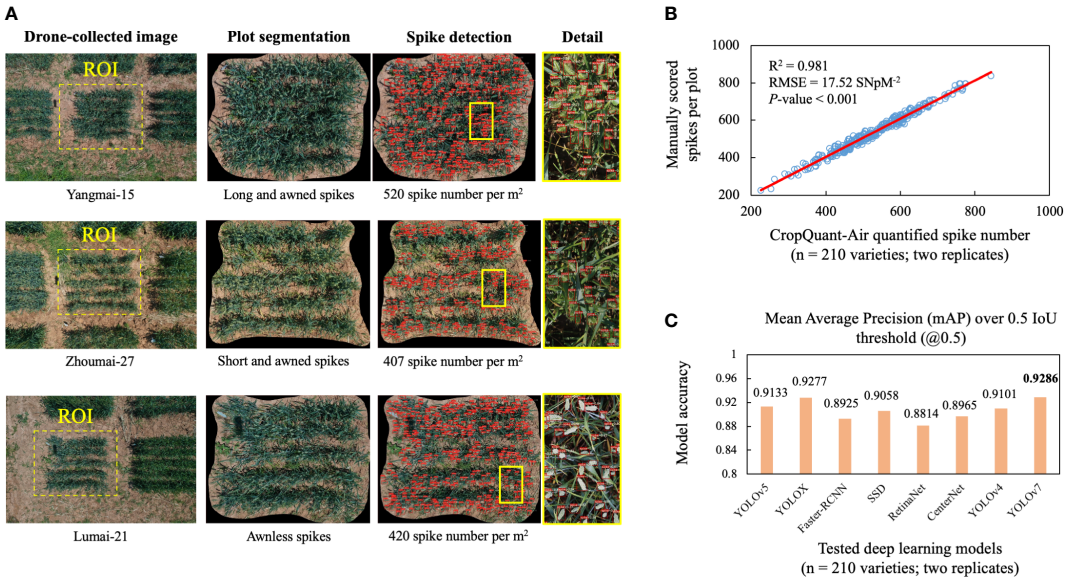


FIGURE 4
Plot- and spike-based detection using the trained YOLCAT-Plot and YOLOv7 models together with correlation analysis between AI-derived and manually scored spike number per m^2 ($SNpM^2$), followed by comparisons of eight AI models for wheat spike detection. **(A)** Plot regions detected from aerial images using YOLACT-Plot model, followed by the optimised YOLOv7 model established for detecting wheat spikes. **(B)** Correlation analysis performed between AI-derived and manually scored wheat spikes using 420 plot-based wheat canopy images, showing a significant positive correlation. **(C)** Performance comparisons of eight state-of-the-art AI-powered object detection methods, indicating that the optimised YOLOv7 model performed the best in terms of the mean average precision at intersection over union (IoU) thresholds of over 0.5 (mAP@0.5).

TABLE 1 Ablation experiments conducted to identify key components essential for better detection performance in the YOLACT-Plot model.

Ablation experiment ID	Backbone network	FPS	Frame mAP%	Frame AP75%	Mask mAP%	Mask AP75%
1	ResNet-101	28.9	62.34	73.11	61.69	68.91
2	Res2Net	26.7	64.69	76.01	63.45	70.75
3	Res2Net-CBAM	27.7	66.38	77.13	64.30	72.36

Table 2 listed the number of parameters and the total *FLOPs* before and after the optimisation, suggesting that the YOLACT-Plot model improved in the network parameters (9.8%), model size (1.06%), and *FLOPs* (6.7%) compared with a standard YOLACT++ model.

To further evaluate the YOLACT-Plot model in instance segmentation, we compared the model with several state-of-the-art instance segmentation DL algorithms such as Mask R-CNN, SOLOv2, and YOLACT++ (Zhao et al., 2021). Our wheat-plot training data was also used when training and testing these DL models. Table 3 listed the results, indicating that the YOLACT-Plot model outperformed Mask R-CNN, SOLOv2, and YOLACT++ in the mAP by 4.15%, 5.33% and 4.1%, respectively. Hence, the feature extraction capability in the YOLACT-Plot model was clearly enhanced due to the optimisation of the learning architecture.

Wheat spike detection using the optimise YOLOv7-based model

In order to evaluate the YOLOv7-based model for wheat spike detection within an identified plot, we have performed correlation analysis based on the 210 varieties possessing varied spike morphologies such as long and awned, short and awned, and awnless varieties (Figure 4A; second column). The detected wheat spikes, red-coloured binding boxes and confidence levels were also generated by the model (Figure 4A; third and fourth columns). We arranged three technicians to manually score the number of wheat spikes per plot using the same plot images segmented by the YOLCAT-Plot model. Coefficient of determination (R^2) was computed to evaluate correlations between the CropQuant-Air-derived and manually scored spike numbers together with root-mean-square error (RMSE), resulting in $R^2 = 0.981$ ($P < 0.001$, RMSE = 17.52; Figure 4B). The result suggested that the AI-powered spike detection was significantly correlated with the manual scoring, indicating the reliability of the AI-powered detection under field conditions.

Additionally, we compared the spike detection results generated by the optimised YOLOv7 model and seven state-of-the-art object detection models, including YOLOv4, YOLOv5, YOLOX, Faster-RCNN, SSD, RetinaNet, and CenterNet (Zhang et al., 2023), all of which were carefully fine-tuned to yield an optimal wheat spike detection. The detection results produced by the DL models (Figure 4C) suggested that the YOLOv7-based model achieved the best accuracy in terms of the mAP@0.5 (i.e. 0.9286), slightly higher than YOLOX (0.9277), YOLOv5 (0.9133), SSD (0.9058), and YOLOv4 (0.9101), demonstrating the biological relevance of applying DL techniques to study spike-like yield components under field conditions.

The GUI of CropQuant-Air software

The CropQuant-Air software system provides a graphical user interface (GUI) that enables non-expert users to perform plot-based trait analysis using an aerial image selected through a unified workspace. A user selects the image selection section (Figure 5A) to choose an aerial image. After that, the CropQuant-Air will initiate the display function to visualise the selected image in the workspace for the user to verify the selection. By clicking the 'Next' button, the software invokes the plot segmentation module that applied the YOLACT-Plot model to process the selected image, resulting in the central plot in the input image segmented from its surrounding pixels (Figure 5B). Depending on the GPU and the size of the selected image, the segmentation process could take up to 10-15 seconds. The final step of the analysis is to carry out wheat spike detection (using the optimised YOLOv7) and canopy-level trait analysis (using the AirMeasurer library), which generates red-coloured binding boxes and confidence levels of all the detected spikes (Figure 5C, right), as well as trait analysis results (including $SNpM^2$, canopy coverage, ExR, NDYI, VARI, ASM, and GLCM dissimilarity) in a table at the bottom of the workspace (red dotted rectangle; Figure 5C, left). The software also supports batch-processing, which can analyse a series of input images and export associated trait analysis (in CSV). Users can download the analysis results and processed images (i.e. segmented plots and detected spikes) via the software. Using an NVIDIA 3060 graphics card, we could achieve a shorter running time (27-30% fasters than CPU-based computation during batch processing) on the CropQuant-Air system compared with an integrated graphics (e.g. Intel's Iris graphics series) as both plot segmentation and spike detection models were accelerated by GPU through parallel computing.

TABLE 2 Comparison of parameters and FLOPS before and after improving the model.

Model	Parameter size/MB	Model size/MB	FLOPS/GFLOPS
YOLACT++	31.62	126	15.36
YOLACT-Plot	33.79	129	16.77

TABLE 3 Result comparison between DL models using the wheat-plot training dataset.

DL models	Backbone network	FPS	Mask mAP%	Mask AP75%	Mask AP50%
Mask R-CNN	ResNet-101	25.8	61.21	68.21	84.10
SOLOv2	ResNet-101	29.6	60.03	68.31	85.87
YOLACT++	ResNet-101	25.1	61.26	69.43	86.33
YOLACT-Plot	CBAM-ResNet-101	21.2	65.36	72.60	87.39

Trait analysis using varieties from different production regions

We aimed to apply the CropQuant-Air software to quantify differences between varieties selected from the three wheat production regions (53 varieties from Central China, 83 from East China, and 74 from North China; also see Supplementary Material). After processing the 210 varieties, we produced six phenotypic traits relevant to yield components. Comparing the SNpM^2 trait, while the average value of spike density was slightly different across the three variety groups, increasing between Central (350-500 per m^2, mean = 410), Eastern (400-550 per m^2, mean = 500) and Northern wheat varieties (500-575 per m^2, mean = 530), the distribution for Northern varieties was much more diverse (Figure 5D), indicating the large variation of spike density in the variety group.

The same elongation was apparent when comparing performance-related traits, where the spectral traits (i.e. ExR and NDYI) also had a much broader distribution in the selected Northern varieties (Figure 5E; right), suggesting varied colour features (i.e. ExG) and developmental paces (i.e. NDYI) among the Northern varieties. The three other measured traits such as canopy coverage, VARI, and ASM were generally similar between the three variety groups according to the violin diagrams. It seemed that the canopy-level phenotypic variation between all the Eastern and Central varieties were relatively small, whereas the Northern varieties possessed much bigger differences (Figure 5E). The above observation was applied to the following yield-based analysis and was utilised when building the yield classification model.

The yield classification in wheat

To classify wheat yield production for agronomic management reasons (Leilah and Al-Khateeb, 2005), we chose the XGBoost model to perform yield-based classification. We used the trait

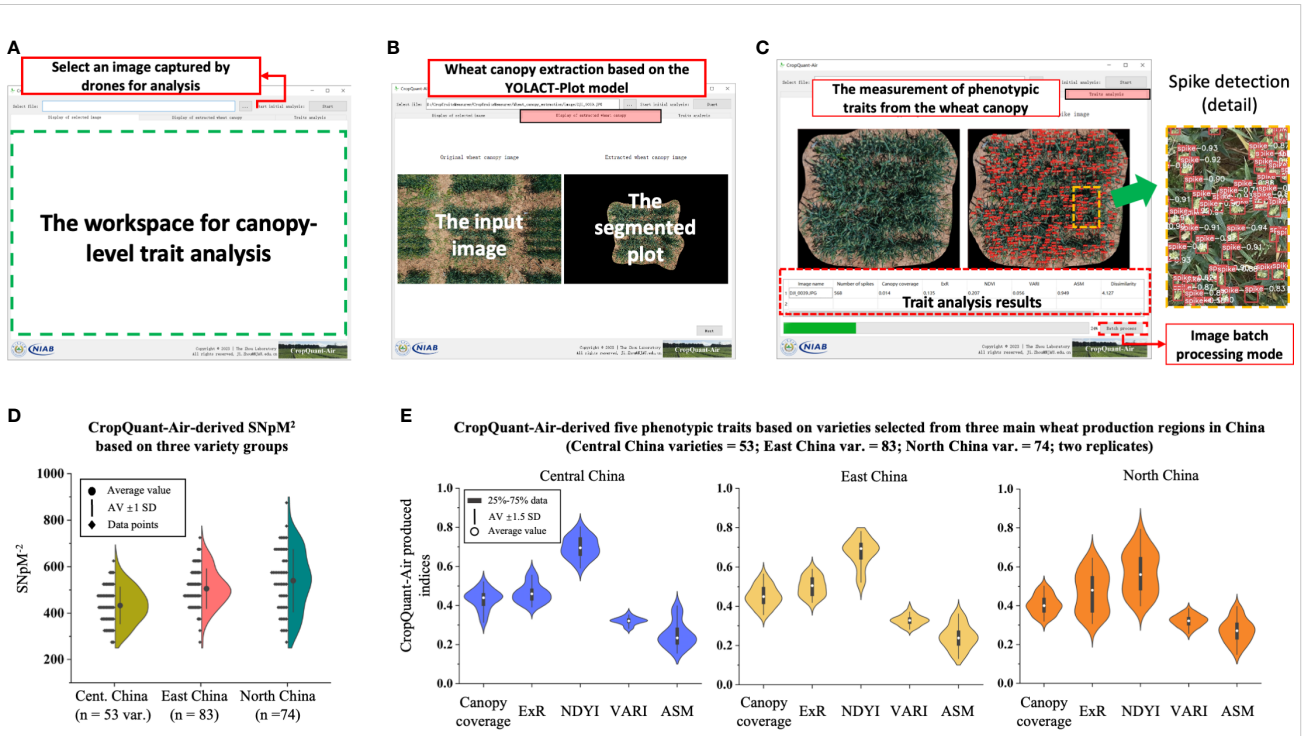


FIGURE 5
Graphic user interface (GUI) of the CropQuant-Air software system developed for non-expert users together with trait analysis results produced by the system. **(A–C)** The GUI window of the CropQuant-Air system, consisting of input and analysis sections, which could process a single or a series of drone-collected aerial image for plot segmentation and phenotypic analysis, quantifying traits such as SNpM^2, morphological and spectral traits. **(D, E)** CropQuant-Air-derived traits divided by wheat varieties selected from three main wheat production regions in China, i.e. Central, East and North China.

analysis results (seven parameters; n = 210 records, which was averaged using the two replicates) generated by the CropQuant-Air system as inputs (Figure 6A), including SNpM^2, canopy-level spectral (i.e. ExR, NDYI, VARI) and textural traits (canopy coverage, ASM, dissimilarity). The dataset together with the variety-based yield production data (210 records, derived from plot-based grain production) was then divided into 7:3 ratio, with 70% (147 lines) for training and 30% (63 lines) for testing. When applying the XGBoost learning model to classify yield production, we performed cross-validation in each round of Boosting iteration, enabling the optimal iteration number (Figures 6B, C). The yield production was divided into three categories, i.e. high-, medium-, and low-yielding groups, following a published approach for wheat breeding and cultivation (Pantazi et al., 2016).

The trained XGBoost model identified 101 high-yield, 90 medium-yield and 19 low-yield wheat varieties (Figure 6D; also see Table S2 in the Supplementary Material). We used confusion matrices to verify the accuracy of the model with manually scored yield production and concluded that: (1) for the high-yielding varieties, the model achieved an accuracy of 97.0%; (2) the medium-yielding group, 96.4% accuracy; (3) the low-yielding group, 94.7% accuracy (Figure 6E; left). Moreover, we studied the weights of all the traits in the model and identified that SNpM^2, NDYI, and ASM jointly contributed 82.7% of the prediction power (coloured red in Figure 6E; right), indicating their relevance in wheat yield production. To gain an in-depth understanding of the three traits, we then plotted the distribution of the SNpM^2, NDYI, and ASM traits using violin diagrams. We could observe that: (1) for the SNpM^2 trait, spike density of high-yield varieties largely located in the 375-500 region, whereas medium- and low-yield groups had a more diverse spike density; also, low-yield varieties on average had a lower SNpM^2 value with many low values absent from the other two groups (Figure 6F; left); (2) for the NDYI trait, both high- and low-yield varieties had diverse distributions with peaks at 0.68 and 0.45, respectively; the medium-yield varieties followed a double normal distribution with the two peaks corresponding to those in the high- and low-yield groups (Figure 6F; middle); (3) for the ASM trait, broadly similar distributions across the 3 yield groups could be observed, with descending means from high to low (Figure 6F; right).

Discussion and conclusion

The ability to identify key phenotypic traits that could be utilised to classify yield production was key for breeders, crop researchers, growers and farmers, and even policymakers as reliable decisions could be rendered to facilitate agronomic management, the selection of crop varieties, and even planning food supply for the market (Chen et al., 2021). For example, understanding the yield potential at key growth stages was essential for breeders to make decisions regarding their crop improvement strategies, helping an efficient identification of genotypes with desired yield- and performance-related traits (Cobb et al., 2013). From cultivation and agronomy's perspective, yield-based analysis could also lead to the development of more precise agronomic management activities to optimise crop growing conditions and thus improved yield performance (Reynolds et al., 2019). For growers and farmers, to be able to classify yield enabled

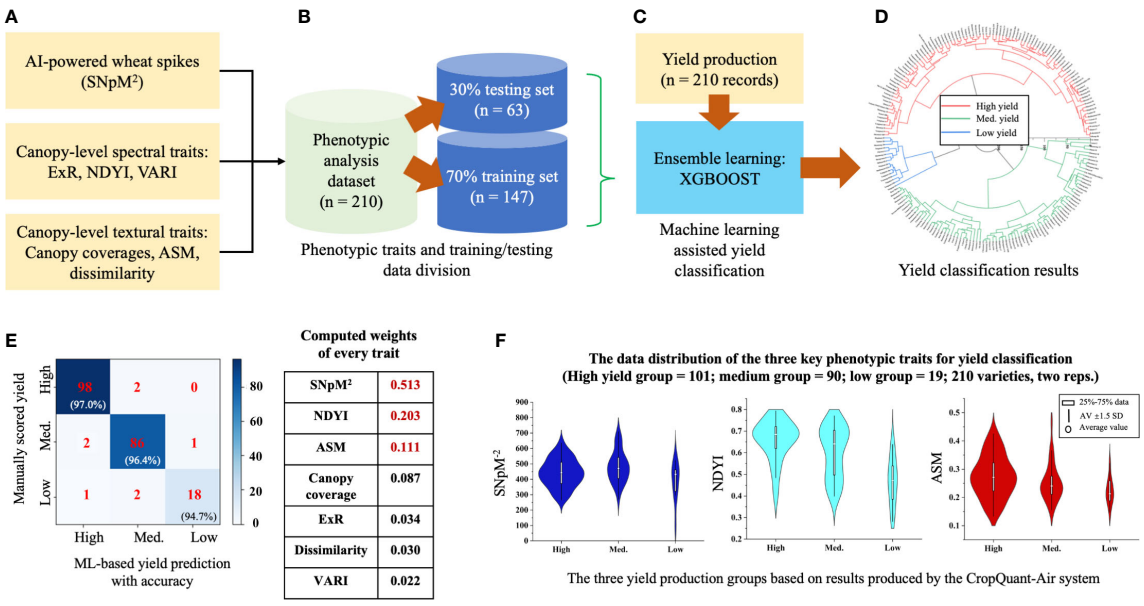


FIGURE 6
The establishment of the yield classification model and identified key contributing phenotypic traits. **(A–C)** The XGBoost model was used to train the yield classification model with 147 lines and 63 lines for evaluation. **(D)** The model was then applied to classify yield production, resulting in three yield groups. **(E)** Confusion matrix and weights of every trait were computed to verify the classification result and identify key contributing traits. **(F)** Violin diagrams used to represent distributions of the three key traits according to the three yield groups.

efficient crop management, providing a baseline to plan agricultural activities (Kremen et al., 2012).

Standard drone-based phenotyping and AI-powered trait analysis

In our study, we demonstrated that low-cost drones could be utilised to perform standardised aerial imaging to collect canopy-level wheat spikes at key growth stages, whose quality was sufficient for AI-powered plot and spike detection, as well as yield-based classification. To quantify the $SNpM^2$ trait, we combined the plot- and spike-level object detection, which was empowered by the AI-based semantic segmentation and vision-based object detection to identify ROIs (i.e. wheat plots) from an aerial image, within which spikes were detected. This progressive algorithmic approach helped us establish an effective workflow to batch-process aerial image series, improving the productivity of the analysis solution presented here. More interestingly, we incorporated spectral and textural features and semantic information of wheat spikes into the model training, which achieved the best detection accuracy compared with seven state-of-the-art DL models, demonstrating a valuable attempt that combined plot- (i.e. the YOLACT-Plot model for instance segmentation) and spike-level (i.e. the optimised YOLOv7 model for semantic segmentation) DL techniques with traditional image processing algorithms to quantify key yield components. To verify the AI-powered trait analysis, we further evaluated the computational results with both manually scored spike number and yield production, resulting in highly significant correlations and thus the reliability of our phenotypic analysis pipeline.

The open-source platform and yield classification

To enable non-experts to use our solution, we developed an open-source software system called CropQuant-Air, which integrated DL models and image processing algorithms to perform plot-based spike detection, as well as spectral and textural trait analysis with a batch-processing mode. Due to limited toolkits available for nonexperts to examine multigenic traits and develop markers (Sun et al., 2022), we developed the CropQuant-Air system using open scientific libraries, demonstrating the value of open scientific solutions for plant researchers when carrying out phenotypic analysis. The modular design also indicated that all the functions or modules in the CropQuant-Air could be utilised independently, accelerating other academic researchers or developers to build upon our work. Furthermore, we are maintaining the software via our GitHub repository, so that new developments of CropQuant-Air could be promptly shared with the broader plant research community to support other phenotyping research.

To facilitate yield classification in wheat, we produced a separate XGBoost ensemble, through which we identified that the $SNpM^2$ trait contributed the most in yield classification and hence the most important factor for yield-related prediction in wheat. Also, subsequent improvements to the ensemble model could include spikelet density, historic yield records, growth stages, and key environmental factors such as ambient temperature and accumulated temperature (Yang et al., 2018), which could improve the generalisation of the yield classification to be applicable during the entire reproductive phase and across different environments.

Limitation of the study

With the rapid development of multi- and hyper-spectral imaging technologies in recent years, the quality of visible and invisible spectrum imaging has been greatly improved, providing new approaches to image wheat spikes and their development at the canopy level. We used sRGB images to capture wheat spikes' spectral and morphological features; however, it is likely that multi- and hyper-spectral imagery could obtain more unique spectral signatures of wheat spikes and thus potentially easier to analyse the trait. Additionally, it is worth noting that the RGB images were very limited in detecting plant abiotic or biotic stresses at the spike level, for which hyperspectral sensors could be valuable in studying plant-disease interactions such as the early growth of *Fusarium* within infected wheat spikes (Ninomiya, 2022). Also, our study focused on detecting wheat spikes within breeding plots and did not perform trait analysis under agricultural conditions. Hence, more R&D activities are still required if the CropQuant-Air system needs to be utilised for cultivation and agronomic services.

Another limit of the open scientific platform that could prevent easy-to-access of open scientific work was the Python dependencies. Due to computer vision and DL/ML based software implementation, when sharing, extending, and upgrading our modules in the CropQuant-Air system, it was important to ensure that the correct versions of ML/DL and open scientific libraries were installed. We mitigated the version risk by releasing an executive file (.EXE) of the system, which required us to publish new versions of the executive file if new functions or dependencies were updated. As a result, a community-driven solution might be valuable to develop and improve CropQuant-Air, promoting open and easy-to-use software solutions jointly via the GitHub platform, which could also maximise the impacts of open scientific software R&D in a collaborative manner.

Future work

Besides the desktop implementation of CropQuant-Air, we could consider deploying the analysis pipeline onto the cloud-based and/or edge computing, so that the software solution could be utilised for different breeding and crop research scenarios. New hardware is also likely to support near real-time analysis based upon our phenotypic analysis solution, providing more economic and powerful tools for agricultural practitioners and researchers. So, key yield-related analysis could be obtained to benchmark yield potential, comparing the performance of different crop varieties and identifying varieties with higher yields under field conditions.

This could also be valuable when identifying crop varieties that were better adapted to local environmental conditions, leading to less water, fertilisers, and other agronomic inputs, which will help growers and farmers reduce the environmental footprint while still maintaining yields. Finally, through big data analytics and cost-effective hardware, much labour-intensive crop surveillance activities could be greatly benefited, facilitating Agri-Food and Agri-Tech companies, policymakers to determine the economic viability of recommended varieties in changing environments, which could also help assess the potential commercial value of the selected varieties so that sound and affordable agricultural production could be promoted.

Data availability statement

The testing datasets, Jupyter notebook, and CropQuant-Air software used in this paper are available at the Zhou lab's GitHub repository: https://github.com/The-Zhou-Lab/CropQuant-Air/releases/tag/v1.0. The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation. Other data and user guides are openly available on request.

Author contributions

JiZ and JC wrote the manuscript with inputs from all the authors; QL, HL, YX, and RJ conducted field experiments and aerial phenotyping under JiZ, GZ, and DJ's supervision; JiZ, GD, RJ, QL, and JC performed data analysis and statistical analysis; JieZ, JC, GS, and QL built and tested the deep learning models, trait analysis, and the yield classification model under JiZ and DJ's supervision. JieZ and JC developed the GUI of the CropQuant-Air system with help from GS. JC, JieZ, and QL contributed equally to this work. All authors contributed to the article and approved the submitted version.

Funding

This work was partially supported by the National Natural Science Foundation of China (32070400 to JiZ). The drone-based phenotyping and yield prediction were supported by the Key Project of Modern Agriculture of Jiangsu Province (BE2019383). JiZ, RJ, and GD were partially supported by the Allan & Gill Gray Philanthropies' sustainable productivity for crops programme (G118688 to the University of Cambridge and Q-20-0370 to NIAB). JiZ and RJ were also supported by the One CGIAR's Seed Equal Research Initiative (5507-CGIA-07 to JiZ), as well as the United Kingdom Research and Innovation's (UKRI) Biotechnology and Biological Sciences Research Council's (BBSRC) International Partnership Grant (BB/X511882/1).

Acknowledgments

The authors would like to thank all members of the Zhou laboratory at the Nanjing Agricultural University (NAU, China) and Cambridge Crop Research, the National Institute of Agricultural Botany (NIAB, UK). In particular, the authors would like to thank Daniel Reynolds, Jeremy Tang, and Alex Swanepoel for their technical supports in aerial phenotyping and establishing training datasets.

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/fpls.2023.1219983/full#supplementary-material

SUPPLEMENTARY TABLE 1
The trial design and wheat varieties studied in the 2021/22 season.

SUPPLEMENTARY TABLE 2
Yield classification classified by the XGBoost model, including 101 high-yielding, 90 medium-yielding, and 19 low-yielding wheat varieties, with manually scored grain production per unit area ($GPpM^2$).

References

Alkhudaydi, T., Reynolds, D., Griffiths, S., Zhou, J., and de la Iglesia, B. (2019). An exploration of deep-learning based phenotypic analysis to detect spike regions in field conditions for UK bread wheat. *Plant Phenomics* 2019, 1–17. doi: 10.34133/2019/7368761

Angeles Ceron, J. C., Chang, L., Ruiz, G. O., and Ali, S. (2021). Assessing YOLACT++ for real time and robust instance segmentation of medical instruments in endoscopic procedures. *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc EMBS* 1824–1827. doi: 10.1109/EMBC46164.2021.9629914

Bastos, L. M., Carciochi, W., Lollato, R. P., Jaenisch, B. R., Rezende, C. R., Schwalbert, R., et al. (2020). Winter wheat yield response to plant density as a function of yield environment and tillering potential: a review and field studies. *Front. Plant Sci.* 11. doi: 10.3389/fpls.2020.00054

Bauer, A., Bostrom, A. G., Ball, J., Applegate, C., Cheng, T., Laycock, S., et al. (2019). Combining computer vision and deep learning to enable ultra-scale aerial phenotyping and precision agriculture: a case study of lettuce production. *Hortic. Res.* 6. doi: 10.1038/s41438-019-0151-5

Betts, A., Jia, P. W., and Dodson, J. (2014). The origins of wheat in China and potential pathways for its introduction: a review. *Quat. Int.* 348, 158–168. doi: 10.1016/j.quaint.2013.07.044

Bevan, M. W., Uauy, C., Wulff, B. B. H., Zhou, J., Krasileva, K., and Clark, M. D. (2017). Genomic innovation for crop improvement. *Nature* 543, 346–354. doi: 10.1038/nature22011

Bradski, G., and Kaehler, A. (2008). Histograms and matching. *Learn. OpenCV* 193.

Chen, T., and Guestrin, C. (2016). XGBoost: a scalable tree boosting system. *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Min.* 785–794. doi: 10.1145/2939672.2939785

Chen, J., Li, Q., Tan, Q., Gui, S., Wang, X., Yi, F., et al. (2021) Combining lightweight wheat spikes detecting model and offline android software development for in-field wheat yield prediction. *Trans. Chin. Soc. Agric. Eng.* 37 (19), 156–164. doi: 10.11975/j.issn.1002-6819.2021.19.018

Cobb, J. N., DeClerck, G., Greenberg, A., Clark, R., and McCouch, S. (2013). Next-generation phenotyping: requirements and strategies for enhancing our understanding of genotype-phenotype relationships and its relevance to crop improvement. *Theor. Appl. Genet.* 126, 867–887. doi: 10.1007/s00122-013-2066-0

David, E., Madec, S., Sadeghi-Tehran, P., Aasen, H., Zheng, B., Liu, S., et al. (2020). Global wheat head detection (GWHD) dataset: a large and diverse dataset of high resolution RGB labelled images to develop and benchmark wheat head detection methods. *Plant Phenomics* 1–10. doi: 10.34133/2020/3521852

Ferrante, A., Cartelle, J., Savin, R., and Slafer, G. A. (2017). Yield determination, interplay between major components and yield stability in a traditional and a contemporary wheat across a wide range of environments. *F. Crop Res.* 203, 114–127. doi: 10.1016/j.fcr.2016.12.028

Furbank, R. T., Jimenez-Berni, J. A., George-Jaeggli, B., Potgieter, A. B., and Derry, D. M. (2019). Field crop phenomics: enabling breeding for radiation use efficiency and biomass in cereal crops. *New Phytol.* 223, 1714–1727. doi: 10.1111/nph.15817

Gao, S. H., Cheng, M. M., Zhao, K., Zhang, X. Y., Yang, M. H., and Torr, P. (2021). Res2Net: a new multi-scale backbone architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 652–662. doi: 10.1109/TPAMI.2019.2938758

Gizaw, S. A., Godoy, J. G. V., Garland-Campbell, K., and Carter, A. H. (2018). Using spectral reflectance indices as proxy phenotypes for genome-wide association studies of yield and yield stability in pacific northwest winter wheat. *Crop Sci.* 58, 1232–1241. doi: 10.2135/cropsci2017.11.0710

Gracia-Romero, A., Kefauver, S. C., Vergara-Díaz, O., Zaman-Allah, M. A., Prasanna, B. M., Cairns, J. E., et al. (2017). Comparative performance of ground vs. aerially assessed rgb and multispectral indices for early-growth evaluation of maize performance under phosphorus fertilization. *Front. Plant Sci.* 8. doi: 10.3389/fpls.2017.02004

Griffiths, S., Wingen, L., Pietragalla, J., Garcia, G., Hasan, A., Miralles, D., et al. (2015). Genetic dissection of grain size and grain number trade-offs in CIMMYT wheat germplasm. *PloS One* 10, 1–18. doi: 10.1371/journal.pone.0118847

Hasan, M. M., Chopin, J. P., Laga, H., and Miklavcic, S. J. (2018). Detection and analysis of wheat spikes using convolutional neural networks. *Plant Methods* 14, 1–13. doi: 10.1186/s13007-018-0366-8

Jang, G. J., Kim, J., Yu, J. K., Kim, H. J., Kim, Y., Kim, D. W., et al. (2020). Review: cost-effective unmanned aerial vehicle (UAV) platform for field plant breeding application. *Remote Sens.* 12, 1–20. doi: 10.3390/rs12060998

Jian, T., Gong, Y., Zhan, Z., Shi, R., Soltani, N., Wang, Z., et al. (2022). Radio frequency fingerprinting on the edge. *IEEE Trans. Mob. Comput.* 21, 4078–4093. doi: 10.1109/TMC.2021.3064466

Jiang, Z., Tu, H., Bai, B., Yang, C., Zhao, B., Guo, Z., et al. (2021). Combining UAV-RGB high-throughput field phenotyping and genome-wide association study to reveal genetic variation of rice germplasms in dynamic response to drought stress. *New Phytol.* 232, 440–455. doi: 10.1111/nph.17580

Jin, X., Liu, S., Baret, F., Hemerlé, M., and Comar, A. (2017). Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery. *Remote Sens. Environ.* 198, 105–114. doi: 10.1016/j.rse.2017.06.007

Kachamba, D. J., Ørka, H. O., Gobakken, T., Eid, T., and Mwase, W. (2016). Biomass estimation using 3D data from unmanned aerial vehicle imagery in a tropical woodland. *Remote Sens.* 8, 1–18. doi: 10.3390/rs8110968

Kremen, C., Iles, A., and Bacon, C. (2012). Diversified farming systems: an agroecological, systems-based alternative to modern industrial agriculture. *Ecol. Soc* 17. doi: 10.5751/ES-05103-170444

Leilah, A. A., and Al-Khateeb, S. A. (2005). Statistical analysis of wheat yield under drought conditions. *J. Arid Environ.* 61, 483–496. doi: 10.1016/j.jaridenv.2004.10.011

Maimaitijiang, M., Ghulam, A., Sidike, P., Hartling, S., Maimaitiyiming, M., Peterson, K., et al. (2017). Unmanned aerial system (UAS)-based phenotyping of soybean using multi-sensor data fusion and extreme learning machine. *ISPRS J. Photogramm. Remote Sens.* 134, 43–58. doi: 10.1016/j.isprsjprs.2017.10.011

Marza, F., Bai, G. H., Carver, B. F., and Zhou, W. C. (2006). Quantitative trait loci for yield and related traits in the wheat population Ning7840 x Clark. *Theor. Appl. Genet.* 112, 688–698. doi: 10.1007/s00122-005-0172-3

Meyes, R., Lu, M., de Puiseau, C. W., and Meisen, T. (2019) *Ablation studies in artificial neural networks*. Available at: http://arxiv.org/abs/1901.08644.

Misra, T., Arora, A., Marwaha, S., Chinnusamy, V., Rao, A. R., Jain, R., et al. (2020). SpikeSegNet-a deep learning approach utilizing encoder-decoder network with hourglass for spike segmentation and counting in wheat plant from visual imaging. *Plant Methods* 16, 1–20. doi: 10.1186/s13007-020-00582-9

Nduku, L., Munghemezulu, C., Mashaba-Munghemezulu, Z., Kalumba, A. M., Chirima, G. J., Masiza, W., et al. (2023). Global research trends for unmanned aerial vehicle remote sensing application in wheat crop monitoring. *Geomatics* 3, 115–136. doi: 10.3390/geomatics3010006

Ninomiya, S. (2022). High-throughput field crop phenotyping: current status and challenges. *Breed. Sci.* 72, 3–18. doi: 10.1270/jsbbs.21069

Pantazi, X. E., Moshou, D., Alexandridis, T., Whetton, R. L., and Mouazen, A. M. (2016). Wheat yield prediction using machine learning and advanced sensing techniques. *Comput. Electron. Agric.* 121, 57–65. doi: 10.1016/j.compag.2015.11.018

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.

Patrício, D. I., and Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: a systematic review. *Comput. Electron. Agric.* 153, 69–81. doi: 10.1016/j.compag.2018.08.001

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830. doi: 10.1289/EHP4713

Qiu, R., Yang, C., Moghimi, A., Zhang, M., Steffenson, B. J., and Hirsch, C. D. (2019). Detection of fusarium head blight in wheat using a deep neural network and color imaging. *Remote Sens.* 11. doi: 10.3390/rs11222658

Reynolds, D., Ball, J., Bauer, A., Davey, R., Griffiths, S., and Zhou, J. (2019). CropSight: a scalable and open-source information management system for distributed plant phenotyping and IoT-based crop management. *Gigascience* 8, 1–11. doi: 10.1093/gigascience/giz009

Reynolds, M., Chapman, S., Crespo-Herrera, L., Molero, G., Mondal, S., Pequeno, D. N. L., et al. (2020). Breeder friendly phenotyping. *Plant Sci.* 295, 110396. doi: 10.1016/j.plantsci.2019.110396

Sadeghi-Tehran, P., Virlet, N., Ampe, E. M., Reyns, P., and Hawkesford, M. J. (2019). DeepCount: in-field automatic quantification of wheat spikes using simple linear iterative clustering and deep convolutional neural networks. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.01176

Shipman, J. W. (2013). Tkinter 8.5 reference: a GUI for Python. *Comput. (Long. Beach. Calif).* 1–118.

Sun, G., Lu, H., Zhao, Y., Zhou, J., Jackson, R., Wang, Y., et al. (2022). AirMeasurer: open-source software to quantify static and dynamic traits derived from multiseason aerial phenotyping to empower genetic mapping studies in rice. *New Phytol.* 236, 1584–1604. doi: 10.1111/nph.18314

Svensgaard, J., Jensen, S. M., Christensen, S., and Rasmussen, J. (2021). The importance of spectral correction of UAV-based phenotyping with RGB cameras. *F. Crop Res.* 269, 108177. doi: 10.1016/j.fcr.2021.108177

ten Harkel, J., Bartholomeus, H., and Kooistra, L. (2020). Biomass and crop height estimation of different crops using UAV-based LiDAR. *Remote Sens.* 12. doi: 10.3390/RS12010017

Torralba, A., Russell, B. C., and Yuen, J. (2010). LabelMe: online image annotation and applications. *Proc. IEEE* 98, 1467–1484. doi: 10.1109/JPROC.2010.2050290

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* 17, 261–272. doi: 10.1038/s41592-019-0686-2

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*. Available at: http://arxiv.org/abs/2207.02696.

Wang, H., Cimen, E., Singh, N., and Buckler, E. (2020). Deep learning for plant genomics and crop improvement. *Curr. Opin. Plant Biol.* 54, 34–41. doi: 10.1016/j.pbi.2019.12.010

Woo, S., Park, J., Lee, J., and Kweon, I. S. (2018). Convolutional_Block_Attention. *Eccv* 17.

Wu, Z., Shen, C., and van den Hengel, A. (2019). Wider or deeper: revisiting the ResNet model for visual recognition. *Pattern Recognit.* 90, 119–133. doi: 10.1016/j.patcog.2019.01.006

Yang, W., Doonan, J. H., Hawkesford, M. J., Pridmore, T., and Zhou, J. (2021). Editorial: state-of-the-Art technology and applications in crop phenomics. *Front. Plant Sci.* 12. doi: 10.3389/fpls.2021.767324

Yang, G., Liu, J., Zhao, C., Li, Z., Huang, Y., Yu, H., et al. (2017). Unmanned aerial vehicle remote sensing for field-based crop phenotyping: current status and perspectives. *Front. Plant Sci.* 8. doi: 10.3389/fpls.2017.01111

Yang, Q., Shi, L., Han, J., Zha, Y., and Zhu, P. (2019). Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images. *F. Crop Res.* 235, 142–153. doi: 10.1016/j.fcr.2019.02.022

Yang, L., Wen, K. S., Ruan, X., Zhao, Y. X., Wei, F., and Wang, Q. (2018). Response of plant secondary metabolites to environmental factors. *Molecules* 23, 1–26. doi: 10.3390/molecules23040762

Yu, C. W., Chen, Y. L., Lee, K. F., Chen, C. H., and Hsiao, C. Y. (2019). Efficient intelligent automatic image annotation method based on machine learning techniques. *2019 IEEE Int. Conf. Consum. Electron. - Taiwan ICCE-TW 2019* 4–5. doi: 10.1109/ICCE-TW46550.2019.8991727

Zhang, G., Wang, L., Wang, L., and Chen, Z. (2023). Hand-raising gesture detection in classroom with spatial context augmentation and dilated convolution. *Comput. Graph.* 110, 151–161. doi: 10.1016/j.cag.2022.11.009

Zhao, K., Zhang, R., and Ji, J. (2021). A cascaded model based on efficientdet and yolact++ for instance segmentation of cow collar id tag in an image. *Sensors* 21. doi: 10.3390/s21206734

Zhou, J., Chen, J., Shen, L., Dai, J., Wen, Z., Sun, G., et al. (2022). Artificial intelligence:advancing plant research beyong the state of the art. *J. Nanjing Agric. Univ.* 45 (5), 1060–1071. doi: 10.7685/jnau.202207021

Zhou, J., Reynolds, D., Le Cornu, T., Websdale, D., Lister, C., Gonzalez-navarro, O., et al. (2017). CropQuant: an automated and scalable field phenotyping platform for crop monitoring and trait measurements to facilitate breeding and digital agriculture 1–41. doi: 10.1101/161547

Zhou, X., Zheng, H. B., Xu, X. Q., He, J. Y., Ge, X. K., Yao, X., et al. (2017). Predicting grain yield in rice using multi-temporal vegetation indices from UAV-based multispectral and digital imagery. *ISPRS J. Photogramm. Remote Sens.* 130, 246–255. doi: 10.1016/j.isprsjprs.2017.05.003

Zhu, Y., Sun, G., Ding, G., Zhou, J., Wen, M., Jin, S., et al. (2021). Large-Scale field phenotyping using backpack LiDAR and CropQuant-3D to measure structural variation in wheat. *Plant Physiol.* 187, 716–738. doi: 10.1093/plphys/kiab324

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 07 July 2023
DOI 10.3389/fpls.2023.1203836

Check for updates

OPEN ACCESS

EDITED BY
Zhenghong Yu,
Guangdong Polytechnic of Science and Technology, China

REVIEWED BY
Xiaodong Bai,
Hainan University, China
Yong Li,
Guangxi University, China
Yanan Li,
Wuhan Institute of Technology, China

*CORRESPONDENCE
Xin Guan
guanxin1001@sina.com

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

RECEIVED 11 April 2023
ACCEPTED 06 June 2023
PUBLISHED 07 July 2023

CITATION
Zhan W, Chen B, Wu X, Yang Z, Lin C, Lin J and Guan X (2023) Wood identification of *Cyclobalanopsis* (Endl.) Oerst based on microscopic features and CTGAN-enhanced explainable machine learning models.
Front. Plant Sci. 14:1203836.
doi: 10.3389/fpls.2023.1203836

COPYRIGHT
© 2023 Zhan, Chen, Wu, Yang, Lin, Lin and Guan. 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.

Wood identification of *Cyclobalanopsis* (Endl.) Oerst based on microscopic features and CTGAN-enhanced explainable machine learning models

Weihui Zhan[1†], Bowen Chen[1†], Xiaolian Wu[1], Zhen Yang[2], Che Lin[1], Jinguo Lin[1,3] and Xin Guan[1,3*]

[1]College of Materials Engineering, Fujian Agriculture and Forestry University, Fuzhou, Fujian, China, [2]College of Transportation and Civil Engineering, Fujian Agriculture and Forestry University, Fuzhou, Fujian, China, [3]National Forestry and Grassland Administration Key Laboratory of Plant Fiber Functional Materials, Fuzhou, Fujian, China

Introduction: Accurate and fast identification of wood at the species level is critical for protecting and conserving tree species resources. The current identification methods are inefficient, costly, and complex

Methods: A wood species identification model based on wood anatomy and using the *Cyclobalanopsis* genus wood cell geometric dataset was proposed. The model was enhanced by the CTGAN deep learning algorithm and used a simulated cell geometric feature dataset. The machine learning models BPNN and SVM were trained respectively for recognition of three *Cyclobalanopsis* species with simulated vessel cells and simulated wood fiber cells.

Results: The SVM model and BPNN model achieved recognition accuracy of 96.4% and 99.6%, respectively, on the real dataset, using the CTGAN-generated vessel dataset. The BPNN model and SVM model achieved recognition accuracy of 75.5% and 77.9% on real dataset, respectively, using the CTGAN-generated wood fiber dataset.

Discussion: The machine learning model trained based on the enhanced cell geometric feature data by CTGAN achieved good recognition of *Cyclobalanopsis*, with the SVM model having a higher prediction accuracy than BPNN. The machine learning models were interpreted based on LIME to explore how they identify tree species based on wood cell geometric features. This proposed model can be used for efficient and cost-effective identification of wood species in industrial applications.

KEYWORDS

***Cyclobalanopsis* (Endl.) Oerst, wood identification, machine learning, CTGAN, LIME**

1 Introduction

Wood classification is a fundamental and essential task in wood science and technology. It enables the identification of various wood species and ensures their sustainable utilization. Wood can be classified into different levels: kingdom, division, class, order, family, genus, and species (Wheeler and Baas, 1998; Martins et al., 2013). In wood classification, identifying wood at the ßpecies" level is often regarded as the most arduous task. ßpecies" is the primary classification unit that refers to individuals of the same species with the same morphological characteristics, chemical composition, and tissue structure (Wheeler et al., 1989; Mai et al., 2022). The classification of wood at the ßpecies" level demands an in-depth comprehension of the differences among individuals of the same species, necessitating researchers to possess high levels of expertise and skills. According to the recent report "State of the World's Trees" by the Botanic Gardens Conservation International (BGCI), nearly 30% of tree species globally were in danger of extinction, with 27% under threat from the expanding wood trade. Consequently, wood regulation has become a significant challenge in safeguarding tree species. In this context, wood identification plays a crucial role. Therefore, research on wood classification at the ßpecies" level has tremendous importance in conserving forest resources and promoting the wood trade, thereby providing the wood industry with enhanced quality control and management (Gasson et al., 2010; Koch et al., 2015; Wiedenhoeft et al., 2019).

Wood identification is a complicated process. Experts use traditional wood identification techniques based on macroscopic and microscopic wood anatomy (Kuroda, 1987; Coday et al., 1997; Romagnolj et al., 2007). These techniques can identify tree species from raw wood, sawn wood, and finished products. Macroscopic identification serves as a supplemental reference to microscopic identification. The latter compares the morphological characteristics of tissues and cells in three sections of wood samples with accurately named wood specimens, enabling more precise identification (Jansen et al., 1998; Carlquist, 2013; Zhang et al., 2014). To attain accurate wood identification, experts and scholars have conducted research using quantitative identification methods for cell features. Experts use quantitative data to study patterns of cellular structural characteristic variation between tree species and subtle structural differences (Gasson et al., 2010). Nevertheless, manually identifying microscopic features is time-consuming, and some wood species have substantial inter-species variation. Personnel assigned with identification may encounter difficulty in mastering the variation rules of all wood species and struggle to discern nuanced structural differences between them without the help of a microscope. These factors add to the challenges of identifying wood at the species level, necessitating significant effort from researchers (Richter et al., 2004; Wheeler, 2011; Angyalossy et al., 2016).

In recent years, numerous methods have been developed and studied for wood identification, including molecular markers, spectral chemical analysis, stable isotopes, and other techniques (Ohyama et al., 2001; Grabner et al., 2009; Finkeldey et al., 2010; Kobayashi et al., 2019a; Sharma et al., 2020).While these methods have enabled identification at the ßpecies" level, significant investments in workforce and financial resources are required to build corresponding classification feature databases. However, advances in artificial intelligence (AI) technology have led to the emergence of new ideas to facilitate rapid and accurate identification at the ßpecies" level (Tou et al., 2007; Yuliastuti et al., 2013; Mohan et al., 2014). Machine learning forms the core of AI, withdeep learning constituting a large-scale machine learning approach often employing multilayer convolutional neural networks and deep, fully connected neural networks to construct models (Yadav et al., 2015a; Hwang and Sugiyama, 2021). These models rely on vast amounts of input data and significant computing power to gain a deeper understanding of knowledge. However, as the complexity of AI models continues to increase, the models themselves are becoming increasingly opaque, with input and output processing often complicated to comprehend (Sun et al., 2021). Within the field of wood identification, deep learning based on computer vision has been utilized for building models to classify wood species successfully (Hafemann et al., 2014). Researchers aim to capture the microscopic structural characteristics of wood by using a microscope, with image-based data being leveraged as input for the computer vision classification models. The characteristics of these models include complex calculations, large amounts of data, slow training, and relatively poor interpretability. To better facilitate rapid and accurate tree species recognition modeling, rigorous research in feature extraction, data preprocessing and enhancement, model selection, and evaluation is necessary.

The Fagaceae family comprises over 900 species distributed worldwide in Eurasia's temperate and subtropical forests. In China, seven Fagaceae genera are identified: Castanopsis, Quercus, *Cyclobalanopsis*, Lithocarpus, Fagus, Castanea, and Trigonobalanus, with more than 300 species. The *Cyclobalanopsis* genus is the most prevalent, with approximately 80 species predominantly found in the Qinling Mountains and south of the Yangtze River. One Fagaceae species is listed under Appendix II of the Convention on International Trade in Endangered Species of Wild Fauna and Flora (CITES). Six species are listed as Class II National Key Protected Wild Plants in China. Developing a precise and prompt identification method at the level of "species" is crucial to safeguard and sustainably use tree species resources (Lions, 2011; Bergesen et al., 2018). Therefore, this research aims to create a high-quality species identification model based on wood samples, focusing on the *Cyclobalanopsis*genus of Fagaceae.

In wood identification, research in artificial intelligence models has two main directions: popular deep learning models based on image e data (Kwon et al., 2017; Ravindran et al., 2018) and machine learning models based on specific quantitative values (Sugiarto et al., 2017; He et al., 2019; Lens et al., 2020; Liu et al., 2022). While both are valid methodologies, the latter commonly uses traditional machine learning models with relatively simple structures, making them faster to train and easier to comprehend. Consequently, such models have significant potential for application within the field of wood identification. Regarding data types, there are four types of data in wood identification models: microscopic images, stereograms images, CT images, and macroscopic images (Hwang and Sugiyama, 2021). Traditional

wood science suggests that microscopic images have the highest identification accuracy, with other images often used as auxiliary means of wood identification. Therefore, training a wood identification model based on microscopic images is the best choice. In the past decade, research on computer tree species recognition based on microscopic images has indicated that feature selection for microscopic image recognition models can generally be divided into image features and tabular numerical features (Table 1). Convolutional networks based on image features often extract convolutional features through convolutional layers, which are difficult to understand and confusing for wood science researchers. In contrast, machine learning models based on tabular numerical features are relatively easy to understand. However, the feature extraction method used in previous research was based on computer graphics rather than wood anatomy, which still needs to be more user-friendly for wood science researchers. As a result, we are approaching this problem from a wood anatomy perspective, and it is feasible to establish a wood identification model based on the geometric features of the wood anatomy dataset. Therefore, we extracted the geometric features of *Cyclobalanopsis* wood vessel cells and wood fiber cells as training features in this study. By comparing models trained on the geometric feature dataset of the two cell types, it could effectively study the specific impact of each cell type regarding the inter-species identification of *Cyclobalanopsis* wood.

However, due to the complexity of the internal structure of wood cells, collecting geometric feature data requires a substantial amount of time and labor (Von Arx et al., 2016; von Arx et al., 2021). Researchers must further explore data preprocessing and enhancement techniques. Using a deep learning network for feature modeling is a viable solution based on the similarity of geometric feature data within the same wood species. In a related study, Xu et al. (2019) proposed the conditional tabular generative adversarial network(CTGAN), which synthesized tabular dataset. CTGAN was compared with Bayesian and other deep learning methods on seven simulated and eight real dataset. Results revealed that CTGAN outperforms the alternative methods on most datasets, demonstrating a more extraordinary data generation ability that aligns closely with actual data distribution (Assefa et al., 2020; Bourou et al., 2021; Torfi et al., 2022). The robust data generation capabilities of CTGAN have earned the trust of many researchers. As a reliable generation model, CTGAN can help researchers generate data that would otherwise be costly. CTGAN has been applied in various fields, such as electroencephalogram, power system data generation, privacy medical data generation, mobile sensor data generation, and financial asset configuration data generation (Lee and Lee, 2021; Han et al., 2022; Fang et al., 2022; DeOliveira et al., 2022; Peña et al., 2023; Cifuentes et al., 2023). However, the application of generation models in the field of wood science is scarce, and the workload and cost of collecting data for training wood identification models or wood property prediction models are enormous. As a result, in this study, the CTGAN model was used to augment the geometric feature dataset of *Cyclobalanopsis* wood, effectively expanding the scale of the data and improving its diversity and reliability.

General wood identification models often face complex input data and network structure challenges. Typically, evaluations of these models often lack explanations of the models themselves, only providing accuracy on validation sets or confusion matrices as indicators of model quality (Ribeiro et al., 2016). While models perform well in wood identification, they may provide an overestimated recognition effect as real-world data differ significantly from simple validation sets. Therefore, model interpretability is necessary to provide insights into wood science and anatomy development. Expressly, model interpretability permits the analysis of the impact of different component factors on wood identification to create improved identification based on feature impact data (Vellido et al., 2012; Milli et al., 2019). Additionally, interpretable models provide essential explanations to non-data science professionals and the general public, establishing trust in the artificial intelligence model in wood science and the wood industry to enable people to make more informed decisions based on prior knowledge of wood science. For

TABLE 1 Research on CV-based microscopic image identification of wood in the last 10 years.

References	Model	Feature	Feature num	The best classification rate(%)
Ravindran and Wiedenhoeft (2020)	Deep learning	Convolution features		96.10%
Lens et al. (2020)	Deep learning	Convolution features		95.60%
Hwang et al. (2020)	Machine learning	SIFT	128	99.40%
Kobayashi et al. (2019b)	Machine learning	SIFT+Connected component labelling	17 + 17	95.30%
Rosa da Silva et al. (2017)	Machine learning	LPQ+LBP	256 + 4116	90.00%
Hwang et al. (2018)	Machine learning	SIFT	128	96.30%
Yadav et al. (2015b)	Machine learning	LBP	325	97.87%
Yadav et al. (2014)	Machine learning	Coiflet Discrete Wavelet Transform	48	92.20%
Hafemann et al. (2014)	Deep learning	Convolution features		97.32%
Martins et al. (2013)	Machine learning	Image Structural Features+GLCM+LBP	5 + 24 + 59	98.60%
Yadav et al. (2013)	Machine learning	GLCM	44	92.00%

these reasons, evaluations of the model should also include scientific explanations of the model in addition to traditional measures of accuracy and confusion matrix. From the observation of Table 1, machine learning models have an advantage over deep learning models when explaining wood species identification models based on microscopic images. Deep learning models require images as input, yielding convolutional features that are challenging to explain. On the other hand, machine learning models can establish tabular feature datasets based on wood anatomy, with a more intuitive feature selection range, making it easier for wood science researchers to understand the model. As a preliminary study of wood identification model interpretability, this paper will use Local Interpretable Model-Agnostic Explanations (Ribeiro et al., 2016) on machine learning models to interpret the wood identification model, and a simple linear model will approximate the prediction field of interest. The weight coefficients of geometric features will explain how the model identifies the wood species (Mishra et al., 2017; Peltola, 2018).

This study utilized the CTGAN-enhanced simulated *Cyclobalanopsis* wood cell geometric feature dataset to establish two machine learning models (SVM, BPNN) for wood identification. The models were tested and evaluated on an actual *Cyclobalanopsis* wood cell geometric feature dataset, and the interpretability of the wood recognition model was demonstrated using Local Interpretable Model-Agnostic Explanations (LIME). Thus, this study provides the following contributions:

- It proposed a novel wood recognition model based on microscopic images that leveraged wood anatomy principles and cell geometrical feature data to enhance accuracy. Compared with previous wood identification models based on microscopic images, the current work differs in that it extracts a small amount of anatomical feature data and combines it with prior knowledge of wood anatomy instead of extracting complex convolution features and local point-line-face features. The proposed model integrates more with previous knowledge of wood anatomy through artificial feature extraction and model training. Based on artificial neural network analysis, the study found that cell geometrical features, particularly those determined by vessels, significantly impacted wood identification accuracy more than features based on wood fibers.
- The CTGAN model was utilized based on deep learning to augment the wood's quantitative cell geometrical feature data, representing the first time such an approach has been used. The feasibility of this method was evaluated using a real wood cellgeometry dataset. CTGAN model dramatically reduces the cost of manually collecting anatomical features, enabling wood scientists to train their species identification models relatively quickly based on the wood anatomical features.
- Furthermore, interpretation of wood identification models was researched by examining the impact of various wood cell geometrical features of *Cyclobalanopsis* on models. This study presented the first of its kind. The findings of our study offered a quantitative and qualitative understanding of interpretable models for wood anatomy and also provided valuable insights for improving feature engineering of future artificial intelligence-based wood identification systems, thereby advancing the field of wood science.

2 Experiments and methods

2.1 Sample collection

Sample trees were collected from the middle subtropical region of Fujian province, China. The discs near ground with 5.0 cm thickness and logs above ground 1.3 m (breast height) with 2.0 m length were got from the straight and healthy average stem in the stand. Then discs and logs as wood samples were placed in a shaded and well-ventilated indoor environment until air drying. The general information of the tree samples was shown in Table 2.

Using a continuous zoom stereomicroscope (MZS0745, Guilin Mete Optical Instrument Co., Ltd.), stereograms of three types of Cyclobalanopsis wood were captured at 10X magnification, as shown in Figure 1. The macroscopic features of Cyclobalanopsis

TABLE 2 Tree species and collecting locations.

Latin name	Tree Age/ year	Genus Name	Collecting locations	Material colour	Scents
Cyclobalanopsis gilva (Blume) Oerst	35	*Cyclobalanopsis (Endl.) Oerst*	Jian'ou, Fujian	Sapwood is yellow-brown and the heartwood is red-brown to shades of red-brown	No distinct odor
Cyclobalanopsis chungii (Metc.) Y. C. Hsu et H. W. Jen ex Q. F. Zheng	33	*Cyclobalanopsis (Endl.) Oerst*	Minqing, Fujian	heartwood is red-brown and the sapwood is yellow-brown	No distinct odor
Cyclobalanopsis glauca(Thunb.) Oerst.	33	*Cyclobalanopsis (Endl.) Oerst*	Pingnan, Fujian	Heartwood color was similar to sapwood color with yellow-brown	No distinct odor

10.3389/fpls.2023.1203836

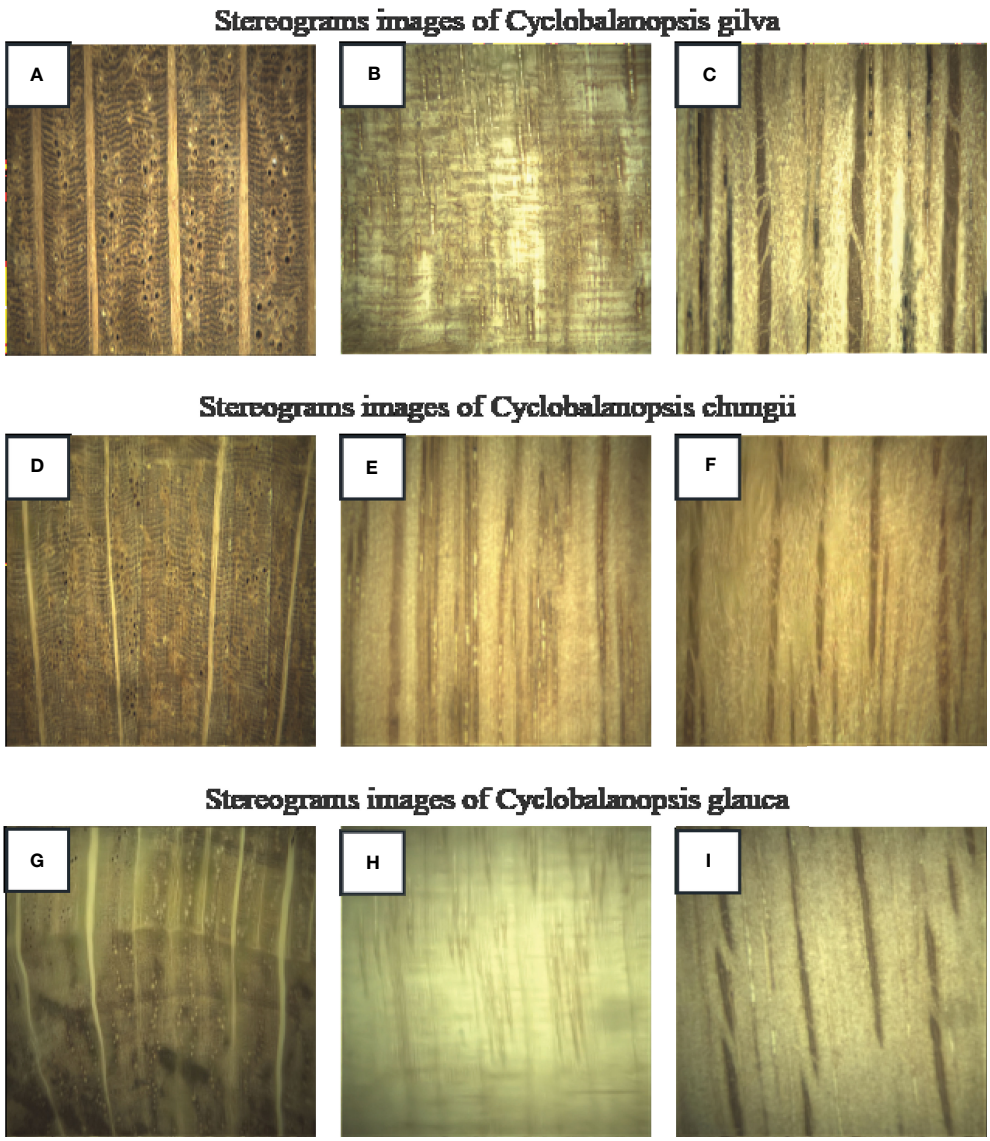


FIGURE 1
Images of the three species of Cyclobalanopsis under a stereomicroscope. **(A, D, G)** is cross section; **(B, E, H)** is radial section; **(C, F, I)** is tangential section.

gilva wood were shown in Figures 1A–C. Heartwood color was distinct from sapwood color, with the sapwood being yellow-brown and the heartwood being red-brown to shades of red-brown. There was no special odor. The growth ring boundaries were indistinct with a uniform width. The macroscopic features of Cyclobalanopsis chungii wood were shown in Figures 1D–F. Heartwood color was distinct from sapwood color, with the heartwood being red-brown and the sapwood being yellow-brown. There was no special odor. The growth ring boundaries were indistinct, which width was not uniform with an average value of 3.03mm. The macroscopic features of Cyclobalanopsis glauca wood were shown in Figures 1G–I. Heartwood color was similar to sapwood color with yellow-brown. There was no special odor. The growth ring boundaries were distinct with a uniform width.

2.2 Wood microstructure production

The air-dried wood was intercepted into a standard three-cut small test block (10mm×10mm×10mm), and to ensure that the cross-section had at least one complete annual ring. The blocks were boiled in distilled water for 8h, and then soaked in distilled water at 26°C for 12h until the specimen were soft. After that, using a sliding microtome (REM-710, Daiwa Optical Machinery Co., Ltd., Japan), sections with a thickness between 10 and 20 were prepared from the wood specimens. All sections were stained with 1% safranin for 3.5 h and subsequently dehydrated using a series of alcohol concentrations (30%, 50%, 75%, 85%, 95%, 100%) for 5 min each. The fully dehydrated sections were placed in 100% xylene for

clear treatment for 10 min and repeated once. Permanent slides were produced by sealing the slices with neutral resin. Slices were placed under a biological digital microscope (Leica DM2500, Leica Microsystems, Germany) to observe the microscopic characters of the wood. The Leica Application Suite software was used to extract data on microscopic features such as tangential diameter of vessel and vessel lumina, wall thickness of vessel, area of vessel and vessel lumina, perimeter of vessel and vessel lumina, substantive rate of vessel and fiber, area of fiber and fiber lumina.

2.3 Model building

2.3.1 Conditional tabular generative adversarial network

CTGAN is a deep learning model that employs conditional generative adversarial networks (Mirza and Osindero, 2014) to model the probability distribution of tabular data rows and synthesize data with features closely related to the input data. The model accomplishes this through a game between two neural networks: the generator and the discriminator. The generator learns the probability distribution of accurate data and generates high-quality synthetic data. The discriminator constantly judges the generated data and gives feedback to optimize the weight of the neural network. The generator network comprises three fully connected layers that employ batch normalization and LeakyReLU activation functions (Maas et al., 2013). In comparison, the discriminator network has three fully connected layers that use batch normalization, LeakyReLU activation functions, and the final layer with the sigmoid activation function. During the training phase, the generator and discriminator work together to ensure that the synthetic data generated by the generator are indistinguishable from the actual data. The input to the model consists of a noise vector and a conditional vector, while the output is the synthesized data.

The objective function of the Generative Adversarial Network is shown in Equation 1.

$$\min_G \max_D V(D,G) = E_{x\sim P_{\text{data}}(X)}[\log(D(x))] + E_{x\sim P_z(z)}[\log(1-D(G(z)))] \quad (1)$$

In the equation, G represents the generator, and D represents the discriminator. The generator and discriminator are trained iteratively until the discriminator cannot distinguish the authenticity of the generated data. Once this occurs, the Generative Adversarial Network (GAN) model is optimized (224 Courville and Bengio, 2014). Figure 2 shows the machine learning model based on CTGAN established in this study. The generator takes the *Cyclobalanopsis* wood cell geometrical feature data as a conditional input, adds random noise, and sends it to the discriminator for evaluation. The feedback from the discriminator is then used to adjust the network weights of the generator, which gradually optimizes the generated data features to mirror that of the actual dataset. CTGAN can significantly reduce the cost of collecting cell features, enabling this experiment to train a robust and reliable model with only a tiny amount of geometrical features collected from wood cells.

2.3.2 Backward propagation neural network

Artificial Neural Network (ANN) is a commonly used machine learning technique in various fields, including classification, prediction, optimization, and other tasks (Ingre and Yadav, 2015). The Back Propagation-Artificial Neural Network (BPNN) algorithm is widely used for quantitatively modeling numerical features. It is a simple multi-layer neural network with multiple layers of neurons trained using the backpropagation algorithm (Rumelhart et al., 1986). The BPNN's basic structure includes inputs, outputs, and multiple hidden layers of neurons. Each neuron's output is connected to the neurons in the previous layer, forming a multi-layer feedforward neural network structure (Agatonovic-Kustrin and Beresford, 2000).

As illustrated in Figure 2, the BP artificial neural network comprises three layers of neurons: the input layer, hidden layer, and output layer. The hidden layer consists of two layers and 20

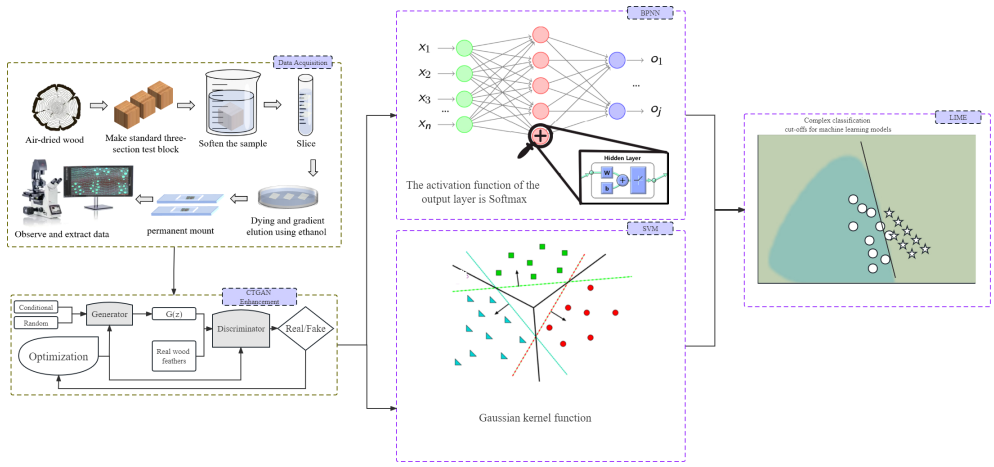


FIGURE 2
Machine learning model workflow based on CTGAN, wood cell geometry features figures and LIME.

nodes. The *Cyclobalanopsis* wood cell geometrical feature data was collected through software, enhanced by CTGAN, and standardized before passing through the input layer to the second layer, i.e., the hidden layer, where weight transfer occurs. In the hidden layer, each neuron was activated by weight W, threshold b, and the Relu activation function and transmitted to the output layer. The output layer generated the predicted value of the neural network and compared it with the expected value. Any errors detected were back propagated from the output layer, and the weights and thresholds were adjusted. The repeated training and adjustment process continued until the output error reached an acceptable level. The specific model calculation was shown below.

The model was propagated in the forward direction to generate the predicted output $\hat{y}_i$. The cross-entropy loss function was first used to evaluate the error between the predicted value $\hat{y}_i$ and true value y_i. The cross-entropy loss function determined the better prediction model by the maximum likelihood estimate of the correct prediction for each set of data. The smaller cross-entropy could get the lower model prediction error (2).

$$loss = -\sum_{i=1}^{n} y_i \log \hat{y}_i \tag{2}$$

In the first backpropagation step, the gradient value $\frac{\partial loss}{\partial w_o}$ of the backpropagation process with respect to the output layer weight w_o was obtained by taking the derivative of error with respect to the weight w_o. y represented the true value and $\hat{y}$ represented the predicted value which was operated by the Softmax activation function in the output layer(3).

$$\frac{\partial loss}{\partial w_o} = \frac{\partial loss}{\partial y}\frac{\partial \hat{y}}{\partial w_o} \tag{3}$$

The input values of the Softmax activation function were a set of vectors of tree species prediction scores, and the output consisted of a prediction vector of identification probabilities for the corresponding tree species(4).

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \tag{4}$$

In the second backpropagation step, the gradient value $\frac{\partial loss}{\partial w_h}$ in the backpropagation process with respect to the hidden layer weight w_h was obtained by taking the derivative of the error with respect to the weight w_h. h represented the Sigmoid activation function in the hidden layer(5).

$$\frac{\partial loss}{\partial w_h} = \frac{\partial loss}{\partial y}\frac{\partial \hat{y}}{\partial h}\frac{\partial h}{\partial w_h} \tag{5}$$

The activation function h for the j node in the hidden layer could be formulated as equation 6:

$$h = f(x_j) = \frac{1}{1 + exp\ (-x_j)} \tag{6}$$

4. Finally, the weights of output and hidden layers were updated by gradient descent(7).

$$w_{update} = w_{old} - \eta\frac{\partial loss}{\partial w_{old}} \tag{7}$$

Where w_{update} represented the weights of hidden and output layers after updating, w_{old} represented the weights of hidden and output layers before updating, η represented the model learning rate, and $\frac{\partial xtloss}{\partial w_{old}}$ was the gradient of corresponding weights.

5. The updated weights were substituted into the forward propagation for a new prediction, and the error was made to reach an acceptable range by repeating the above steps continuously. Finally, the predicted value was output.

The standard BP neural network algorithm's gradient descent algorithm (Ruder, 2016) adjusts the network weights and thresholds along the negative gradient direction of a network error. This eventually causes the error to reach a minimum value.

2.3.3 Support vector machine

Support Vector Machine (SVM) is a machine learning algorithm based on statistical VC dimension theory and structural risk minimization (Joachims, 1998). Initially applied to binary classification, it can also solve multi-classification problems and effectively address small sample, non-linear, high-dimensional, and local minimum problems. The core of SVM lies in finding the maximum hyperplane in high-dimensional space to separate sample data, making the classification reach the maximum interval, given a point x on the hyperplane, and ω as a vector perpendicular to the hyperplane as displacement interval, and b as the shift interval, the maximum interval can be represented as equation 8. If $\frac{2}{\|\omega\|}$ is maximized, it is equivalent to minimizing $\|\omega\|^2$. Equation 8 can be transformed into equation 9, the basic mathematical model of SVM.

$$\begin{cases} \max\limits_{\omega,b} \dfrac{2}{\|\omega\|} \\ y_i(\omega \cdot x_i \pm b) \geq 1, i = 1, 2, \cdots, m \end{cases} \tag{8}$$

$$\begin{cases} \min\limits_{\omega,b} \dfrac{1}{2}\|\omega\|^2 \\ y_i(\omega \cdot x_i \pm b) \geq 1, i = 1, 2, \cdots, m \end{cases} \tag{9}$$

$$y = \sum_{i=0}^{m} \omega_i^T \cdot x_i \pm b \tag{10}$$

In this research, the SVM classifier would be based on the Gaussian kernel function with a kernel scale of 3. The computation process of the SVM classifier was as follows: firstly, the geometric feature dataset of *Cyclobalanopsis* wood cells would be passed to the initial calculation module to obtain m-dimensional coefficients $\|\omega\|^T$ and a threshold value b. Then, equation 10 would multiply and sum the *Cyclobalanopsis* species wood cell geometric feature values x corresponding to each block with $\|\omega\|^T$. Finally, by comparing the relationship between the classification calculation result and the threshold value b, the classification result of *Cyclobalanopsis* species wood cell geometric feature data could be obtained (Chang et al., 2005; Friedrichs and Igel, 2005; Xiao et al., 2014).

2.4 Model evaluation

In machine learning, a model's classification capability is a crucial performance metric for evaluation purposes (McAvaney et al., 2001). During the process of evaluating model performance, various performance indicators are applied to obtain a comprehensive understanding of the model's performance. Generally, accuracy and confusion matrix are the two most commonly used evaluation indicators.

Accuracy, represented by formula 11, is defined as the proportion of samples correctly classified to the total number of samples. TP denotes the count of correctly identified positive samples, and TN represents the count of correctly identified negative samples by the classifier(11).

$$ACC = \frac{TP + TN}{TOTAL} \tag{11}$$

Despite being a widely used evaluation metric, accuracy needs to be improved for assessing the performance of classifiers in multi-class classification problems. Hence, the confusion matrix is a more suitable evaluation metric for comprehensively evaluating the model performance. The confusion matrix tabulates the misclassification frequency of each tree type by comparing the predicted and actual categories (Gauch et al., 2003; Raschka, 2018). The confusion matrix illustrates the predicted and true categories in columns and rows. Consequently, the matrix enables the identification of misclassified classes and their respective frequencies for each tree species.

2.5 Model interpretability

Interpretability refers to the degree to which humans can comprehend artificial intelligence algorithms, often called "black boxes," since their knowledge representation is often counterintuitive, making it complicated to understand their behavior. Interpretability techniques facilitate revealing the rationale behind predictions generated by black box machine learning models(Zhang and Zhu, 2018; Poursabzi-Sangdeh et al., 2021) By identifying how features affect, or do not affect, predictions, interpretability techniques assess whether the model utilizes appropriate professional knowledge, thereby detecting any biases that might arise during training.

Machine learning methods are substantially employed in identifying wood species. However, these models' complexity requires interpretable methods to unveil their decision-making procedures. This study suggested that LIME explain the model to address this issue. LIME is a technique for interpreting models, initially proposed in 2016 by Marco Ribeiro and colleagues, that assists in explaining the decision-making process of complex black-box models. LIME can enlighten us about the weight of variables, their contribution to specific predictions, or similar factors. In studies on identifying tree species based on wood geometry parameters, LIME can reveal the identification procedure of machine learning models based on distinct input cell features. Specifically, the LIME technique utilizes a local linear model to approximate a model's prediction on specific data. This method is simpler to comprehend since it only focuses on specific data on local linear structures (Swirszcz et al., 2009; Lozano et al., 2011; Ribeiro et al., 2016). As shown in Figure 2, machine learning models often have very complex and convoluted boundaries that are difficult to understand. Using the LIME technique, a simple linear model is locally fitted to the data for the boundary, and the interpretability of the model is achieved through the weight coefficients of the linear model.

The parameters of the local model in the LIME technique are calculated by the following formula(12).

$$\phi(x) = \underset{g \in G}{\operatorname{argmin}} L(f, g, \pi_x) + \Omega(g) \tag{12}$$

Among them, f is the original model, G is the hypothesis space of the locally interpretable model, π_x is the distribution of data points similar to instance x, L is the loss function, and $\Omega(g)$ is the regularization term. By minimizing this formula, the optimal local model ϕ can be obtained, thus explaining the output of the original model on a specific instance.

Using the LIME technique, the explanation results of specific instances could be obtained, including the weight of each feature, the contribution to the prediction result, and so on. For the BP neural network model and SVM support vector machine based on wood geometry features for tree species identification, the predictions for each wood species based on different models and cell data would be explained by LIME. These explanation results could help us better understand the working mode and importance of the features of the model, thus improving the interpretability and reliability of the model.

3 Results and analysis

3.1 Microstructure of three species of wood

Vessels in cross section of *Cyclobalanopsis gilva* were solitary in radial pattern (Figures 3A, B). The tangential diameter of vessel lumina was in the range 74.19 through 314.56 with a mean value of 198.20. Vessels/mm2 was 3.12. Perforation plates were simple (Figures 3C, G). Helical thickenings were unpresented. Tyloses presented in some vessel lumina. Intervessel pits were alternate with the shape of rounded or angular (Figures 3D, E). Vessel-ray pits were with much reduced borders to apparently simple: pits vertical mostly (Figure 3E). Vasicentric tracheids presented. Axial parenchyma presented diffuse-in-aggregates numerously, in narrow bands or lines up to six cells wide commonly and diffuse rarely (Figure 3A). Wood rays were exclusively uniseriate with an average number of 10.6/mm. Rays were homocellular with all ray cells procumbent. Two types of wood rays presented: (1) uniseriate and 2-seriate rays with a height of 2 22 ray cells (Figures 3F, H); (2) aggregate rays with a breadth >11 ray cells (Figure 3G).

Vessels in cross section of *Cyclobalanopsis chungii* were solitary in a radial or diagonal pattern (Figure 3I). The tangential diameter of vessel lumina was in the range 39.86 through 202.77 with an

10.3389/fpls.2023.1203836

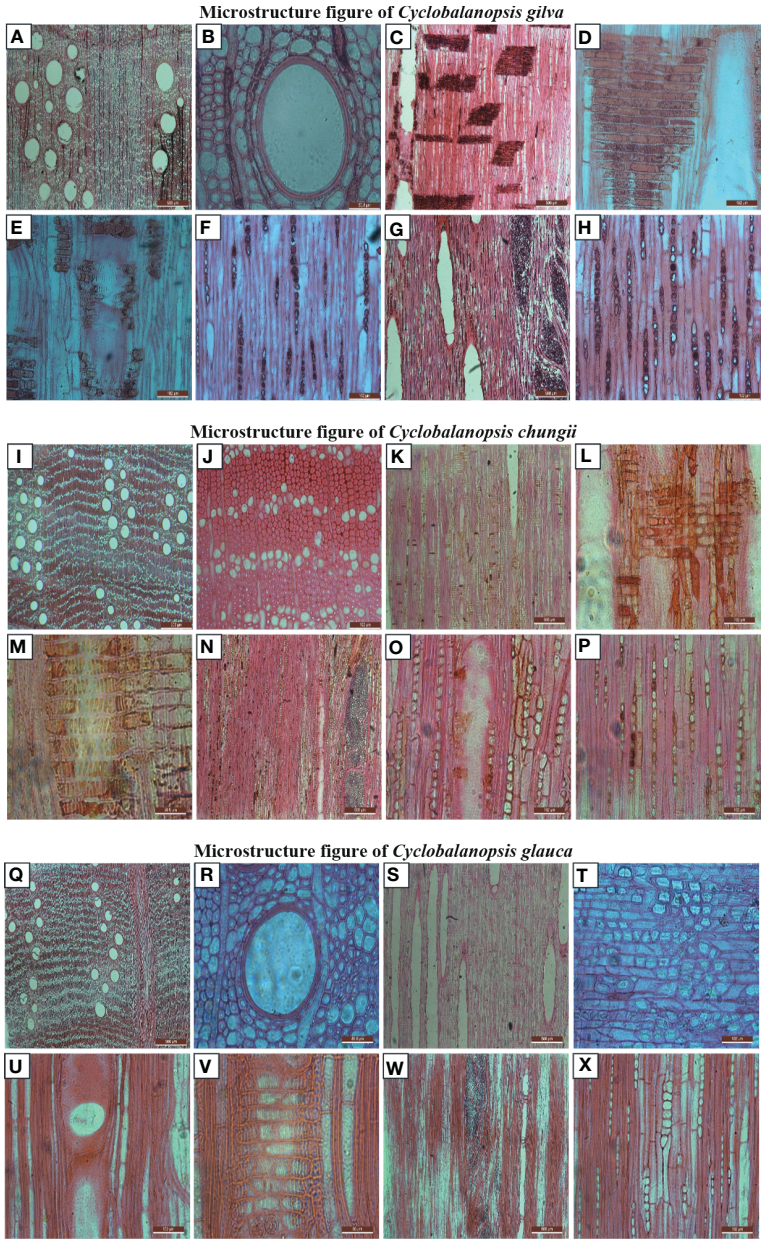


FIGURE 3
Microstructure Images of the three species of *Cyclobalanopsis* under a biological digital microscope. **(A–H)** is *Cyclobalanopsis gliva*; **(I–P)** is *Cyclobalanopsis chungii*; **(Q–X)** is *Cyclobalanopsis glauca*.

average value of 139.27. Vessels/mm2 was 3.91. Perforation plates were simple. Helical thickenings were unpresented. Tyloses presented in some vessel lumina. Intervessel pits were alternate with the shape of rounded or oval (Figure 3O). Vessel-ray pits were with much reduced borders to apparently simple: pits rounded mostly (Figures 3L, M). Vasicentric tracheids presented. Fibers with simple pits were infrequent (Figure 3P). Axial parenchyma presented in narrow bands of 1 3 cells wide (Figure 3I). Wood rays were homocellular with all ray cells procumbent. Two types of wood rays presented: (1) uniseriate and 2-seriate rays with a height of 3 21 ray cells (Figures 3O, P); (2) aggregate rays with a breadth >11 ray cells (Figure 3N).

Vessels in cross section of *Cyclobalanopsis glauca* were solitary in a radial or dendritic pattern (Figures 3Q, R). The tangential diameter of vessel lumina was in the range 44.33 through 184.50 with an average value of 116.62. Vessels/mm2 was 3.4. Perforation plates were simple (Figures 3S, U). Helical thickenings were unpresented. Tyloses presented in some vessel lumina. Intervessel pits were alternate with the shape of rounded or oval. Vessel-ray pits were with much reduced borders to apparently simple: pits rounded mostly. Vasicentric tracheids presented. Axial parenchyma presented in narrow bands of 1 5 cells wide (Figure 3Q). A majority of crystals presented in aggregate rays (Figure 3T). Wood rays were exclusively uniseriate with an average number of

7.38/mm. Wood rays were homocellular with all ray cells procumbent. Two types of wood rays presented: (1) uniseriate and 2-seriate rays with a height of 3 20 ray cells (Figure 3X); (2) aggregate rays with a breadth >11 ray cells (Figure 3W).

3.2 Microstructural data of three species of wood

54 sets of microscopic features data from vessel elements and 96 sets of microscopic features data from wood fibers were collected for *Cyclobalanopsis gilva*, while 60 sets of data from vessel elements and 106 sets of data from wood fibers were gathered for *Cyclobalanopsis chungii*. For *Cyclobalanopsis glauca*, 49 sets of data from vessel elements and 92 sets of data from wood fibers were acquired (Table 3).

In the genus *Cyclobalanopsis*, the anatomy of vessel cells is highly variable and can aid in species differentiation. As shown in Table 3, we compared the vessel elements and wood fiber characteristics of three *Cyclobalanopsis*species : *Cyclobalanopsis gilva*,*Cyclobalanopsis chungii*, and *Cyclobalanopsis glauca*.

The thickness of the vessel wall was significantly greater in *Cyclobalanopsis gilva* compared to the other two species. The tangential diameter and area of the vessel were also significantly more prominent in *Cyclobalanopsis gilva*. These features and the vessel's circumference could be used to distinguish *Cyclobalanopsis gilva* from the other two species. The luminal size of *Cyclobalanopsis gilva* vessels was also more extensive, but this difference was not as significant as the other features. The ratio of wall to lumina was higher in *Cyclobalanopsis chungii* than in the other two species. While this feature is less significant than the vessel diameter, it could still contribute to species identification. The substantial rate of the vessels, which measures their compactness, was significantly higher in *Cyclobalanopsis chungii*. This feature alone may not be sufficient to differentiate the species, but it may contribute to the overall analysis.

The area of the wood fibers was significantly larger in *Cyclobalanopsis gilva* —however, more than this feature is required to differentiate among the species. The size of the fiber lumina was larger in *Cyclobalanopsis chungii*, but this difference was not as significant as in the vessel lumina. The substantial rate of the fibers was significantly higher in *Cyclobalanopsis chungii*, which could be combined with other features to differentiate species.

In conclusion, the vessel elements and wood fiber characteristics of *Cyclobalanopsis* spp. can be helpful in species differentiation. The significant differences observed in vessel wall thickness, tangential diameter, area, circumference, and wood fiber substantial rate can aid in identifying *Cyclobalanopsis gilva*, *Cyclobalanopsis chungii*, and *Cyclobalanopsis glauca*. In the following work, the CTGAN model based on the anatomical characteristics of *Cyclobalanopsis* wood was trained to produce reliable simulated anatomical feature

TABLE 3 Extracted data of microscopic features from three species.

	Microscopic features	Cyclobalanopsis gilva			Cyclobalanopsis chungii			Cyclobalanopsis glauca		
		Mean value	standard deviation	Coefficient of variation	Mean value	standard deviation	Coefficient of variation	Mean value	standard deviation	Coefficient of variation
Vessel elements	Wall thickness/μm	12.136	2.998	0.143	5.720	1.268	0.222	4.129	1.284	0.311
	Tangential diameter/μm	198.199	59.125	0.298	139.272	35.051	0.252	113.654	37.029	0.326
	Tangential diameter of lumina/μm	186.024	58.061	0.312	127.832	33.894	0.265	105.436	35.309	0.335
	ratio of wall to lumina	0.071	0.025	0.348	0.097	0.041	0.422	0.087	0.042	0.490
	Area/μm^2	38603	21929	0.568	18529	8148	0.440	12481	6041	0.484
	Area of lumina/μm^2	34613	20380	0.589	15894	7240	0.456	10848	5444.	0.502
	Substantial rate	0.118	0.040	0.340	0.156	0.046	0.293	0.138	0.041	0.297
	Circumference/μm	668.663	211.527	0.316	471.070	117.040	0.248	377.391	119.029	0.315
	Circumference of lumina/μm	630.738	207.618	0.329	435.361	113.410	0.260	351.415	112.736	0.321
Wood fibers	Area/μm^2	206.634	53.536	0.259	143.274	41.509	0.290	106.891	40.613	0.380
	Area of lumina/μm^2	89.580	39.339	0.439	18.198	12.782	0.702	21.006	12.838	0.611
	Substantial rate	0.568	0.143	0.251	0.919	0.036	0.039	0.805	0.080	0.100

10.3389/fpls.2023.1203836

data for training a tree species identification model. Finally, the LIME model's explanation technique was used, compared, and discussed with the prior knowledge of the *Cyclobalanopsis* identification learned above.

3.3 Dataset construction and enhancement

Based on the geometric characteristics of real *Cyclobalanopsis* wood cells, we customized a *Cyclobalanopsis* cell simulation geometric data generation model using the CTGAN model through the application of Python's SDV library, and the model was reliably evaluated by Python's SDMetrics library. We used 163 sets of vessel cell data and 294 sets of wood fiber cell data to train the cell simulation geometric data generation model. Due to the relatively small amount of data in the training set, we used a relatively small batch size and a very long training period to improve the model's generalization and feature extraction abilities. In order to avoid the problem of model overfitting caused by small batch sizes and long training periods, we used relatively small neural network dimensions and relatively small learning rates. We used the Adam optimizer for optimization. After multiple rounds of parameter adjustment, our final model training parameters are shown in Table 4, and the model training loss is shown in Figure 4. The generator loss of the Vessel-CTGAN model gradually stabilizes after 6000 rounds, and the discriminator loss also slowly converges. Meanwhile, the WoodFiber-CTGAN model's generator loss stabilized at around -1.2 after 3000 rounds, and the discriminator also showed a stable trend, thus completing the model's training.

Synthetic data was generated using the CTGAN model based on a real geometric feature dataset of *Cyclobalanopsis*. The model learns from the *Cyclobalanopsis* wood cell geometrical feature dataset to generate synthetic data corresponding to actual *Cyclobalanopsis* wood geometric features' statistical regularities. This expansion of the dataset was one of the benefits of this approach. The training dataset for wood geometrical features consisted of 163 groups of tracheal geometric data from *Cyclobalanopsis gilva*, *Cyclobalanopsis chungii*, and *Cyclobalanopsis glauca*, as well as 294 groups of wood fiber geometric data, trained for 5000 rounds.

In addition, the KS test was used to compare the differences in probability distribution between actual wood geometric feature data and generated wood geometric feature data. The KS test could be used to compare the difference between two probability distributions, and its core was to calculate the empirical cumulative distribution function (ECDF) of the two distributions. The KS statistic D represented the maximum difference between the two ECDFs. The larger the D value, the more significant the difference between the two distributions. The Ks complement was a supplementary index of the KS test, and its value equalled 1-D. Therefore, the larger the D value, the smaller the Ks complement value. The range of Ks complement was between 0 and 1. When the Ks complement value was closer to 1, it indicated that the two probability distributions were more similar; conversely, when the Ks complement value was closer to 0, it indicated that the two

TABLE 4 Model hyperparameters configuration.

Model	epochs	batch size	discriminator dim	discriminator decay	discriminator lr	discriminator steps	Embedding_dim	generator decay	generator dim	generator lr	pac	Optimizers
Vessel-CTGAN	10000	30	(256, 256)	1e-6	1e-4	1	128	1e-6	(256,256)	1e-4	10	Adam
WoodFiber-CTGAN	10000	20	(256, 256)	1e-6	1e-4	1	128	1e-6	(256,256)	1e-4	10	Adam
Model	**epochs**	**number of fully-connected layers**	**first layer size**	**second layer size**	**activation function**	**standardize**						
BPNN	1000	2	10	10	relu	true						
Model	**epochs**	**kernel**	**nuclear constraint**	**kernel scale**	**multi-class approach**	**standardize**						
SVM	1000	gaussian	1	12	One-to-one	true						

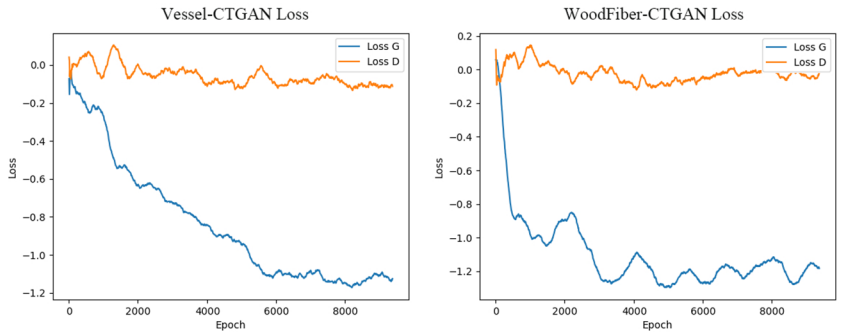


FIGURE 4
Variation in loss values for generators and discriminators.

probability distributions were less similar. When Cifuentes et al. (2023) studied optimizing asset allocation data generation using CTGAN, the Ks complement indicator reached 0.87. Alqarni and El-Alfy (2022) studied generating network traffic intrusion detection data, and the Ks complement score ranged from 0.77 to 0.82. Peng et al. (2021) studied on generating credit rating data, the Ks complement indicator reached 0.88.

In order to ensure accuracy and richness in data sampling, we divided the original real cell data into ten equal parts for each tree species. Then we carried out a single sampling of each of the equal parts to generate simulated data. For each equal part, we generated 100 simulation cell data samples, resulting in 1000 simulation cell data samples per tree species. The new data generated by CTGAN achieved good indicators on Ks complement (Figure 5). The average Ks complement of the vessel reached0.9, 0.88 and 0.87 corresponded to *Cyclobalanopsis gilva*, *Cyclobalanopsis chungii* and *Cyclobalanopsis glauca* in turn. The average Ks complement of the wood fibers reached 0.86, 0.89 and 0.92 corresponded to *Cyclobalanopsis gilva*, *Cyclobalanopsis chungii* and *Cyclobalanopsis glauca* in turn. Therefore, using the CTGAN model, 1000 groups of geometric data each for vessels and wood fibers were obtained, which could simulate the numerical distribution of natural wood geometric features of *Cyclobalanopsis gilva*, *Cyclobalanopsis chungii*, and *Cyclobalanopsis glauca*, respectively. SVM and BP neural networks would be trained based on 3000 groups of geometric feature data each for *Cyclobalanopsis*vessels and wood fibers.

3.4 Wood identification results

Using MATLAB, based on the hyperparameters configuration in Table 4, machine learning algorithm models were established to identify three species of *Cyclobalanopsis* by vessel and wood fiber geometry data through BP neural network and SVM support vector machine methods. The training of machine learning models is based on CTGAN-enhanced synthetic cell datasets. In order to verify the recognition effect of the model trained on CTGAN-enhanced data on real cell images, two machine learning models based on different synthetic cells were tested on both synthetic cell test sets and real cell datasets. The recognition results are shown in the Table 5, and the model trained on synthetic data can also achieve excellent recognition on real microscope features. The machine learning model based on Vessel-CTGAN-BPNN achieved high recognition rates of 99.2% in the test set and 96.4% in the real dataset, while 99.4% in the test set and maintained in the real dataset based on the Vessel-CTGAN-SVM. However, the recognition rates of the machine learning model based on the WoodFiber-CTGAN-BPNN was 76.6% in the test set and 75.7% in the real dataset, while 74.1% in the test set and 77.9% in the real dataset based on WoodFiber-CTGAN-SVM. These results indicated that both BPNN and SVM algorithms based on vessel geometry data could achieve high recognition levels, while the SVM algorithm performs better. However, machine learning models based on wood fiber geometry data could not achieve satisfactory recognition results. However, compared with the BPNN and SVM models without CTGAN-enhanced data, the recognition rates of the models have still improved. Compared with the Vessel-BPNN and Vessel-SVM models, the recognition rates have increased by 6.6 and 15.7 percentage points, respectively, after using the CTGAN data enhancement method. Compared with the WoodFiber-BPNN and WoodFiber-SVM models, the recognition rates have increased by 9.6 and 4.0 percentage points, respectively, after using the CTGAN data enhancement method.

The confusion matrices based on the CTGAN simulated vessel and wood fiber cell geometry data were plotted (Figure 6). Two machine learning models, BPNN and SVM, were trained based on CTGAN simulated anatomical data of two types of cells. The BPNN model trained on the CTGAN simulated anatomical data of vessel cells achieved high accuracy in the test set, with only three prediction errors in the identification of 299 *Cyclobalanopsis chungii* tree species, four errors in the identification of301 *Cyclobalanopsis gilva* tree species and no errors in the identification of *Cyclobalanopsis glauca*. Similarly, the SVM model trained on the CTGAN simulated anatomical data of vessel cells also achieved high accuracy in the test set, with only 2 and 3 prediction errors in the identification of 299 *Cyclobalanopsis chungii* and 301 *Cyclobalanopsis gilva* tree species, respectively, and no errors in the identification of *Cyclobalanopsis glauca*. It is worth noting that Vessel-CTGAN-BPNN and Vessel-CTGAN-SVM also achieved high recognition accuracy on the real cell dataset. In the real *Cyclobalanopsis chungii* cell dataset, Vessel-

10.3389/fpls.2023.1203836



FIGURE 5
The KS complement metrics of the real and synthetic *Cyclobalanopsis* wood cell geometry datasets are presented, with the following letter symbols representing: vessel's wall thickness (VEWT), vessel's tangential diameter (VETD), vessel's tangential diameter of lumina (VETDOL), vessel's ratio of wall to lumina (VEROWTL), vessel's area (VEA), vessel's area of lumina (VEAOL), vessel's substantial rate (VESR), vessel's circumference (VEC), vessel's circumference of lumina(VECOF),fiber's area (CA), fiber's substantial rate(SR), and fiber's area of lumina (AOL).

CTGAN-BPNN misclassified only 4 out of 65 wood cell data; in the real *Cyclobalanopsisgilva* cell dataset, Vessel-CTGAN-BPNN misclassified only 1 out of 51 wood cell data, and in the real *Cyclobalanopsis glauca* cell dataset, Vessel-CTGAN-BPNN misclassified only 1 out of 50 wood cell data. Similarly, Vessel-CTGAN-SVM achieved 100% accuracy in the real *Cyclobalanopsis chungii* cell dataset, misclassified only 1 out of 55 wood cell data in the real *Cyclobalanopsis gilva* tree species cell dataset, and achieved 100% accuracy in the real *Cyclobalanopsis glauca* tree species cell dataset. Except for vessel cells, the performance of the SVM and BPNN models trained on CTGAN-simulated anatomical data of wood fiber cells is relatively worse. Nevertheless, the performance of these models is very close in both the test set and the real dataset, which is enough to illustrate that CTGAN-simulated anatomical data of cells can train good wood identification models for real datasets.

TABLE 5 Recognition rates of models based on different cell geometry features and different machine algorithms.

Recognition algorithms	Recognition rate of test set	Recognition rate of real data
Vessel-BPNN	89.8%	
Vessel-SVM	83.7%	
WoodFiber-BPNN	65.9%	
WoodFiber-SVM	73.9%	
Vessel-CTGAN-BPNN	99.2%	96.4%
Vessel-CTGAN-SVM	99.4%	99.4%
WoodFiber-CTGAN-BPNN	76.6%	75.5%
WoodFiber-CTGAN-SVM	74.1%	77.9%

3.5 Model interpretability

3.5.1 Selection of local data for LIME

A LIME linear model was established using Matlab software to fit the identification of BPNN and SVM models on specific data and to demonstrate how the two machine-learning models identify specific wood species. The LIME modeling was conducted on vessel and wood fiber cells of three *Cyclobalanopsis* species. Six samples of two types of cells from the three species were randomly selected for model interpretation and analysis, as presented in Table 6. Here, the 10th, 89th, and 150th vessel cell data and the 14th, 114th, and 230th wood fiber cell data were analyzed. Considering that recognition rate of model was unrelated to model interpretation, BPNN model was randomly selected for model interpretation based on vessel features and SVM model was chosen for model interpretation based on wood fiber features.

Observing the vessel cell data interpreted by the LIME model through random selection, it can be seen that the difference in wall thickness feature is the greatest among the three tree species. Among other features, there are always two tree species with similar values. Theoretically, we hope that the LIME model's explanation technique is in line with our prior knowledge and that tree species identification is based on the wall thickness feature. Observing the wood fiber cell data that the LIME model interprets through random selection, none of the three features make sufficient decisions. Theoretically, we hope that the LIME model's explanation technique can integrate multiple types of information to make decisions.

3.5.2 Model interpretation based on vessel features

In order to explore how the BPNN model identifies wood species based on the geometric features of vessel cells, a simple linear model would be used as a local model to fit the BPNN model. Specifically, we

Replaced the activation function of each hidden unit with a ReLU function to achieve this transformation. Then, the linear model's output was calculated using the following formula(13).

$$y = \sum_{i=1}^{n} w_i x_i + b \tag{13}$$

Among them, n was the number of features, w_i was the weight of the i-th feature, x_i was the value of the i-th feature, and b was the bias term. This relatively simple linear model had good interpretability, which could obtain the interpretation results of the model for predicting specific conduit data. As shown in Figure 7, the geometric features of the conduit cells had a specific impact on the model identification when the BPNN model identified 3 pieces of wood conduit data in Table 6 (Swirszcz et al., 2009; Lozano et al., 2011).

The BPNN model relies mainly on the characteristics of vessel's area of lumina (VEAOL) and vessel's area(VEA) for identifying three types of *Cyclobalanopsis* wood species. VEAOL plays a vital role in identifying *Cyclobalanopsis gilva* and *Cyclobalanopsis glauca*, while VEA plays a significant role in identifying *Cyclobalanopsis chungii* (Figure 7). By analyzing the vessel's substantial rate (VESR) feature values in combination with the prior knowledge from Tables 3, 6, the average value of *Cyclobalanopsis gilva* is 0.118, and the sample value for model interpretation is 0.1428; the average value of *Cyclobalanopsis chungii* is 0.156, and the sample value for model interpretation is 0.1427; the average value of *Cyclobalanopsis glauca* is 0.138, and the sample value for model interpretation is 0.0949. From Figure 7A, the model considers that a value between 0.14 and 0.15 has a negative effect on identifying the cell data as *Cyclobalanopsis gilva*; from Figure 7B, the model considers that a value between 0.14 and 0.15 has a positive effect on identifying the cell data as *Cyclobalanopsis chungii*; from Figure 7C, the model considers that a value between 0.09 and 0.11 has a negative effect on identifying the cell data as *Cyclobalanopsis glauca*. Therefore, the model conforms well to our prior knowledge in judging the VESR feature point.

3.5.3 Model interpretation based on wood fiber features

In order to explore how the SVM model identified tree species based on the geometric features of wood fiber cells, a simple linear model would be used as a local model to fit the SVM. For the SVM support vector machine model, the interpretation method of LIME technology was as follows: first, we calculated the weight of each support vector in a specific instance. It could be calculated by the following formula (14).

$$w_i = \alpha_i y_i K(x_i, x) \tag{14}$$

Among them, α_i was the Lagrange multiplier of the i-th support vector, y_i was the class label corresponding to the i-th support vector, $K(x_i, x)$ was the value of the kernel function, x_i was the feature vector of the i-th support vector, and x was the feature vector of the specific instance.

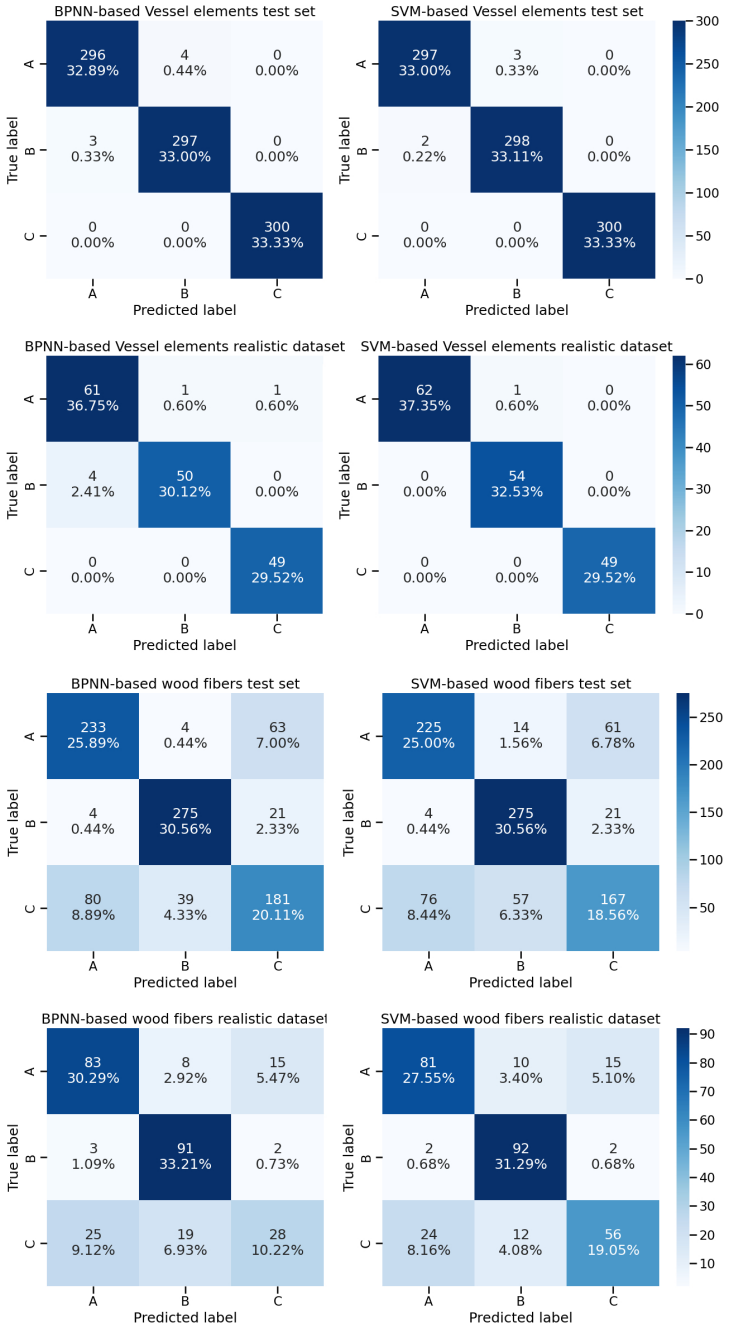


FIGURE 6
Identification results based on vessel cell and wood fiber cell geometry data for three tree species, Note: A, B and C in the diagram represent *Cyclobalanopsis chungii*, *Cyclobalanopsis gilva* and *Cyclobalanopsis glauca* respectively.

Next, the weights of all support vectors were added up to obtain the explanation result for a specific instance (15).

$$y = \sum_{i=1}^{m} w_i \tag{15}$$

Among them, m was the number of support vectors (Swirszcz et al., 2009; Lozano et al., 2011).

Using this method, It could be obtained the weight of each feature in the SVM model for a specific instance and how these weights contribute to the model's prediction results (Figure 7).

The SVM model mainly relies on the features of fiber's area of lumina (AOL) and fiber's area (CA) to identify three types of *Cyclobalanopsis* wood species, with AOL having the most significant weight and CA coming second (Figure 7). Analyzing the AOL feature value more specifically, the SVM model believes that *Cyclobalanopsis gilva* should have the largest AOL, followed by *Cyclobalanopsis glauca*, and finally *Cyclobalanopsis chungii*. By observing Table 2 and extracting prior knowledge, it can be seen that the average value of *Cyclobalanopsis gilva* is 89.580, *Cyclobalanopsis glauca* is 21.006, and *Cyclobalanopsis chungii* is

10.3389/fpls.2023.1203836

TABLE 6 Data on specific cell characteristics of three tree species used for LIME model interpretation.

	species	*Cyclobalanopsis gilva*	*Cyclobalanopsis chungii*	*Cyclobalanopsis glauca*
Vessel elements	Wall thickness/	11.5074	5.66473	0.05444
	Tangential diameter/	114.326	157.148	165.83
	Tangential diameter of lumina/	102.818	145.819	157.268
	ratio of wall to lumina	0.11192	0.0777	4.28088
	Area/μm^2	10858.1	22712.4	22385.7
	Area of lumina/μm^2	9306.97	19471.3	20260.5
	Substantial rate	0.14285	0.1427	0.09494
	Circumference/μm^2	369.944	536.544	531.726
	Circumference of lumina/μm^2	342.762	497.819	505.881
Wood fibers	Area/μm^2	196.275	194.301	63.3392
	Area of lumina/μm^2	59.5437	12.484	19.016
	Substantial rate	0.69663	0.93575	0.69977

18.198. Therefore, the SVM model conforms to our prior knowledge in analyzing the AOL feature value.

4 Conclusion

In this study, two machine learning models, the BP neural network and SVM models, were established based on quantitative geometric characteristics data of vessels and wood fibers to identify the three specie s of *Cyclobalanopsis*, and explained those models with LIME-based model interpretation.

1. The machine learning model constructed based on the geometric characteristics data of vessel elements could effectively identify the three species. Additionally, SVM model was of higher prediction accuracy than BPNN model.
2. The CTGAN model could be effectively applied to enhance geometric characteristic dataset of wood species. The machine learning models trained on the dataset enhanced by the CTGAN model had a high recognition rate for the geometric characteristics of actual microscopic features.
3. The use of LIME model interpretation techniques can effectively verify whether the decision and analysis of the wood identification model conforms to human knowledge. In the field of wood science, model interpretation techniques should be supplemented and further discussed and researched as a model evaluation direction beyond the traditional recognition rate index.

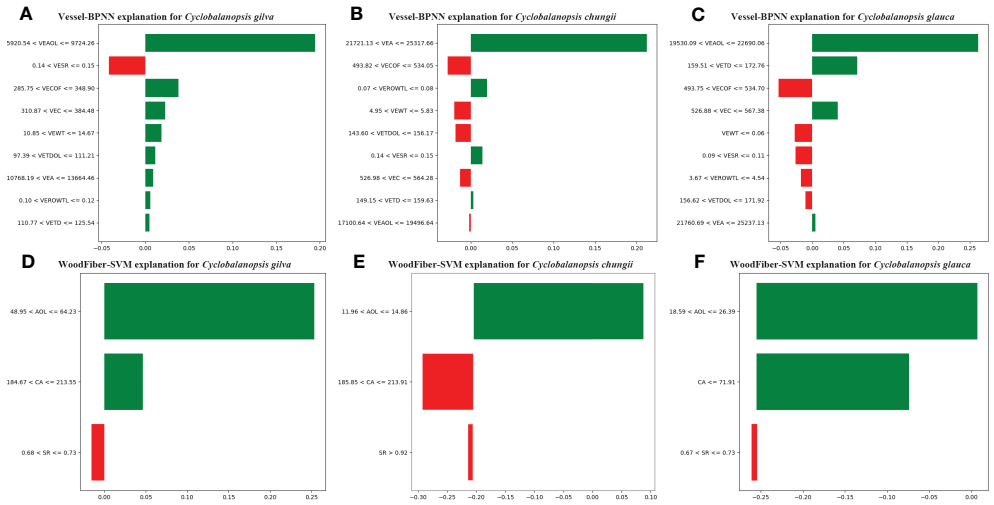


FIGURE 7
LIME Model Interpretation: How Machine Learning Identifies Tree Species Based on Geometric Features of Vessel Cells **(A–C)** and Wood Fiber Cells **(D–F)**, with the Following Letter Symbols Representing: Vessel Wall Thickness (VEWT), Vessel Tangential Diameter (VETD), Vessel Tangential Diameter of Lumina (VETDOL), Vessel Ratio of Wall to Lumina (VEROWTL), Vessel Area (VEA), Vessel Area of Lumina (VEAOL), Vessel Substantial Rate (VESR), Vessel Circumference (VEC), Vessel Circumference of Lumina (VECOF), Fiber Area (CA), Fiber Substantial Rate (SR), and Fiber Area of Lumina (AOL).

The CTGAN and LIME technologies have been preliminarily verified in our work, but there are still some limitations. Firstly, this research work deserves a deeper replication on a larger microstructure data set of wood, which will further demonstrate the significance of our work on a larger scale. Secondly, the LIME model explanation technology can not only build modeling analysis based on numerical features such as micro features but also has a place in image analysis, although it is not perfect, it is worthy of exploration. Finally, LIME is just one of the many artificial intelligence model explanation technologies, and more model explanation technologies should be introduced to the tree species identification field to find the optimal explanation model (Lundberg and Lee, 2017; Shrikumar et al., 2017).

In terms of future research, we offer the following suggestions. Firstly, we recommend replicating and validating the methods on a larger data set. Secondly, models, including YOLO, SegNet, UNet, and the SAM model (Redmon et al., 2016; Badrinarayanan et al., 2017; Huang et al., 2020; Kirillov et al., 2023), which has recently gained popularity, have the potential for introducing wood cell structure analysis and feature extraction. This potential could lead to a significant decrease in the cost of manually collecting features (Yang et al., 2022a; Yang et al., 2022b). Lastly, to create true, industrialized, and intelligent forestry applications, it would be beneficial to further combine model interpretation technology with traditional wood anatomy and wood recognition models.

Data availability statement

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

Author contributions

XW and CL conducted the preparation of *Cyclobalanopsis* wood microsections and captured geometric features measurements. BC contributed to the construction of the BPNN model, while JL provided guidance on determining the geometric feature range of the wood used in this study. WZ provided the construction of the CTGAN model and SVM model, and conducted model explanations based on LIME for the machine learning models. WZ, BC, and ZY conducted data analysis and image rendering, and together wrote the manuscript, with XG providing guidance and oversight on the overall experimental process and manuscript revision. All authors provided operational feedback and worked together to improve the content of the paper. All authors contributed to the article and approved the submitted version.

Funding

This study was supported by Forestry Promotion Project of Fujian Province in China (Grant #: 2021TG13).

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

Agatonovic-Kustrin, S., and Beresford, R. (2000). Basic concepts of artificial neural network (ann) modeling and its application in pharmaceutical research. *J. Pharm. Biomed. Anal.* 22, 717–727. doi: 10.1016/S0731-7085(99)00272-1

Alqarni, A. A., and El-Alfy, E.-S. M. (2022). Improving intrusion detection for imbalanced network traffic using generative deep learning. *Int. J. Adv. Comput. Sci. Appl.* 13. doi: 10.14569/IJACSA.2022.01304109

Angyalossy, V., Pace, M. R., Evert, R. F., Marcati, C. R., Oskolski, A. A., Terrazas, T., et al. (2016). Iawa list of microscopic bark features. *IAWA J.* 37, 517–615. doi: 10.1163/22941932-20160151

Assefa, S. A., Dervovic, D., Mahfouz, M., Tillman, R. E., Reddy, P., and Veloso, M. (2020). "Generating synthetic data in finance: opportunities, challenges and pitfalls," in *Proceedings of the First ACM International Conference on AI in Finance*, Vol. 1–8.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). "Segnet: a deep convolutional encoder-decoder architecture for image segmentation," in *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39. 2481–2495.

Bergesen, H. O., Parmann, G., and Thommessen, Ø.B. (2018). "Convention on international trade in endangered species of wild fauna and flora (cites)," in *Yearbook of International Cooperation on Environment and Development 1998–99*, Routledge. 156–157.

Bourou, S., El Saer, A., Velivassaki, T.-H., Voulkidis, A., and Zahariadis, T. (2021). A review of tabular data synthesis using gans on an ids dataset. *Information* 12, 375. doi: 10.3390/info12090375

Carlquist, S. (2013). *Comparative wood anatomy: systematic, ecological, and evolutionary aspects of dicotyledon wood* (Berlin, Germany: Springer Science & Business Media).

Chang, Q., Chen, Q., and Wang, X. (2005). "Scaling gaussian rbf kernel width to improve svm classification," in *2005 international conference on neural networks and brain (IEEE)*, Vol. 1. 19–22.

Cifuentes, A., Peña, J-M, Suarez, F, Larre, O, and Ramırez, D. (2023). A modified CTGAN-Plus-Features based method for optimal asset allocation. *Tech. rep. arXiv*.

Coday, A E., and Maun, K. W. (1997). *Identification of hardwoods: a microscope key* (Bucknalls Lane, Watford, Herts: Constructions Research Communications Ltd).

Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advanc Neural* 63, 139–144. doi: 10.1145/3422622

DeOliveira, J., Gerych, W., Koshkarova, A., Rundensteiner, E., and Agu, E. (2022). "Har-ctgan: a mobile sensor data generation tool for human activity recognition," in *2022 IEEE International Conference on Big Data (Big Data) (IEEE)*. 5233–5242.

Fang, M. L., Dhami, D. S., and Kersting, K. (2022). "Dp-ctgan: differentially private medical data generation using ctgans," in *Artificial Intelligence in Medicine: 20th International Conference on Artificial Intelligence in Medicine, AIME 2022, Halifax, NS, Canada*, , June 14–17, 2022. 178–188.

Finkeldey, R., Leinemann, L., and Gailing, O. (2010). Molecular genetic tools to infer the origin of forest plants and wood. *Appl. Microbiol. Biotechnol.* 85, 1251–1258. doi: 10.1007/s00253-009-2328-6

Friedrichs, F., and Igel, C. (2005). Evolutionary tuning of multiple svm parameters. *Neurocomputing* 64, 107–117. doi: 10.1016/j.neucom.2004.11.022

Gasson, P., Miller, R., Stekel, D. J., Whinder, F., and Ziemińska, K. (2010). Wood identification of dalbergia nigra (cites appendix i) using quantitative wood anatomy, principal components analysis and naïve bayes classification. *Ann. Bot.* 105, 45–56. doi: 10.1093/aob/mcp270

Gauch, H. G., Hwang, J. G., and Fick, G. W. (2003). Model evaluation by comparison of model-based predictions and measured values. *Agron. J.* 95, 1442–1446. doi: 10.2134/agronj2003.1442

Grabner, M., Salaberger, D., and Okochi, T. (2009). "The need of high resolution-x-ray ct in dendrochronology and in wood identification," in *2009 Proceedings of 6th International Symposium on Image and Signal Processing and Analysis (IEEE)*. 349–352.

Hafemann, L. G., Oliveira, L. S., and Cavalin, P. (2014). "Forest species recognition using deep convolutional neural networks," in *2014 22Nd international conference on pattern recognition (IEEE)*. 1103–1107.

Han, G., Liu, S., Chen, K., Yu, N., Feng, Z., and Song, M. (2022). "Imbalanced sample generation and evaluation for power system transient stability using ctgan," in *Intelligent Computing & Optimization: Proceedings of the 4th International Conference on Intelligent Computing and Optimization 2021 (ICO2021) 3 (Springer)*. 555–565.

He, T., Marco, J., Soares, R., Yin, Y., and Wiedenhoeft, A. C. (2019). Machine learning models with quantitative wood anatomy data can discriminate between swietenia macrophylla and swietenia mahagoni. *Forests* 11, 36. doi: 10.3390/f11010036

Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., et al. (2020). "Unet 3+: a full-scale connected unet for medical image segmentation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (IEEE)*. 1055–1059.

Hwang, S., Kobayashi, K., and Sugiyama, J. (2020). "Evaluation of a model using local features and a codebook for wood identification," in *In IOP Conference Series: Earth and Environmental Science (IOP Publishing)*, Vol. 415. 012029.

Hwang, S.-W., Kobayashi, K., Zhai, S., and Sugiyama, J. (2018). Automated identification of lauraceae by scale-invariant feature transform. *J. Wood Sci.* 64, 69–77. doi: 10.1007/s10086-017-1680-x

Hwang, S.-W., and Sugiyama, J. (2021). Computer vision-based wood identification and its expansion and contribution potentials in wood science: a review. *Plant Methods* 17, 1–21. doi: 10.1186/s13007-021-00746-1

Ingre, B., and Yadav, A. (2015). "Performance analysis of nsl-kdd dataset using ann," in *2015 international conference on signal processing and communication engineering systems (IEEE)*. 92–96.

Jansen, S., Kitin, P., De Pauw, H., Idris, M., Beeckman, H., and Smets, E. (1998). Preparation of wood specimens for transmitted light microscopy and scanning electron microscopy. *Belgian J. Bot.* 13 (1), 41–49.

Joachims, T. (1998). Making large-scale SVM learning practical. *Tech. rep. Tech. Rep*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., et al. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*.

Kobayashi, K., Hwang, S.-W., Okochi, T., Lee, W.-H., and Sugiyama, J. (2019a). Non-destructive method for wood identification using conventional x-ray computed tomography data. *J. Cultural Heritage* 38, 88–93. doi: 10.1016/j.culher.2019.02.001

Kobayashi, K., Kegasa, T., Hwang, S.-W., and Sugiyama, J. (2019b). Anatomical features of fagaceae wood statistically extracted by computer vision approaches: some relationships with evolution. *PLoS One* 14, e0220762. doi: 10.1371/journal.pone.0220762

Koch, G., Haag, V., Heinz, I., Richter, H.-G., and Schmitt, U. (2015). Control of internationally traded timber-the role of macroscopic and microscopic wood identification against illegal logging. *J. Forensic Res.* 6, 1000317. doi: 10.4172/2157-7145.1000317

Kuroda, K. (1987). Hardwood identificatlon using a microcomputer and iawa codes. *IAWA J.* 8, 69–77. doi: 10.1163/22941932-90001030

Kwon, O., Lee, H. G., Lee, M.-R., Jang, S., Yang, S.-Y., Park, S.-Y., et al. (2017). Automatic wood species identification of korean softwood based on convolutional neural networks. *J. Korean Wood Sci. Technol.* 45, 797–808. doi: 10.5658/WOOD.2017.45.6.797

Lee, J., and Lee, O. (2021). Ctgan vs tgan? which one is more suitable for generating synthetic eeg data. *J. Theor. Appl. Inf. Technol.* 99.

Lens, F., Liang, C., Guo, Y., Tang, X., Jahanbanifard, M., da Silva, F. S. C., et al. (2020). Computer-assisted timber identification based on features extracted from microscopic wood sections. *IAWA J.* 41, 660–680. doi: 10.1163/22941932-bja10029

Lions, A. (2011). Convention on international trade in endangered species of wild fauna and flora.

Liu, S., He, T., Wang, J., Chen, J., Guo, J., Jiang, X., et al. (2022). Can quantitative wood anatomy data coupled with machine learning analysis discriminate cites species from their look-alikes? *Wood Sci. Technol.* 56, 1567–1583. doi: 10.1007/s00226-022-01404-y

Lozano, A., Swirszcz, G., and Abe, N. (2011). "Group orthogonal matching pursuit for logistic regression," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics (JMLR Workshop and Conference Proceedings)*. 452–460.

Lundberg, S. M., and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* 30.

Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, Atlanta, Georgia, USA, Vol. 30. 3.

Mai, C., Schmitt, U., and Niemz, P. (2022). A brief overview on the development of wood research. *Holzforschung* 76, 102–119. doi: 10.1515/hf-2021-0155

Martins, J., Oliveira, L., Nisgoski, S., and Sabourin, R. (2013). A database for automatic classification of forest species. *Mach. Vision Appl.* 24, 567–578. doi: 10.1007/s00138-012-0417-5

McAvaney, B. J., Covey, C., Joussaume, S., Kattsov, V., Kitoh, A., Ogana, W., et al. (2001). "Model evaluation," in *Climate change 2001: the scientific basis. contribution of WG1 to the third assessment report of the IPCC (TAR)* (Cambridge: Cambridge University Press), 471–523.

Milli, S., Schmidt, L., Dragan, A. D., and Hardt, M. (2019). "Model reconstruction from model explanations," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 1–9.

Mirza, M., and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv*. doi: 10.48550/arXiv.1411.1784

Mishra, S., Sturm, B. L., and Dixon, S. (2017). Local interpretable model-agnostic explanations for music content analysis. *ISMIR* 53, 537–543.

Mohan, S., Venkatachalapathy, K., and Sudhakar, P. (2014). An intelligent recognition system for identification of wood species. *J. Comput. Sci.* 10, 1231. doi: 10.3844/jcssp.2014.1231.1237

Ohyama, M., Baba, K., and Itoh, T. (2001). Wood identification of japanese cyclobalanopsis species (fagaceae) based on dna polymorphism of the intergenic spacer between trn t and trn l 5' exon. *J. Wood Sci.* 47, 81–86. doi: 10.1007/BF00780554

Peltola, T. (2018). Local interpretable model-agnostic explanations of bayesian predictive models *via* kullback-leibler projections. *arXiv preprint arXiv*.

Peña, J.-M., Suárez, F., Larré, O., Ramírez, D., and Cifuentes, A. (2023). A modified ctgan-plus-features based method for optimal asset allocation. *arXiv preprint arXiv*.

Peng, B., Zhang, A., and Zhang, T. (2021). Credit scoring model in imbalanced data based on cnn-atcn. Preprint from Research Square, 04 Aug 2021. doi: 10.21203/rs.3.rs-714980/v1

Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Wortman Vaughan, J. W., and Wallach, H. (2021). "Manipulating and measuring model interpretability," in *Proceedings of the 2021 CHI conference on human factors in computing systems*. 1–52.

Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv*.

Ravindran, P., Costa, A., Soares, R., and Wiedenhoeft, A. C. (2018). Classification of cites-listed and other neotropical meliaceae wood images using convolutional neural networks. *Plant Methods* 14, 1–10. doi: 10.1186/s13007-018-0292-9

Ravindran, P., and Wiedenhoeft, A. C. (2020). Comparison of two forensic wood identification technologies for ten meliaceae woods: computer vision versus mass spectrometry. *Wood Sci. Technol.* 54, 1139–1150. doi: 10.1007/s00226-020-01178-1

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.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1135–1144.

Richter, H. G., Grosser, D., Heinz, I., and Gasson, P. E. (2004). Iawa list of microscopic features for softwood identification. *Iawa J.* 25, 1–70. doi: 10.1163/22941932-90000349

Romagnolj, M., Sarlatto, M., Terranova, F., Bizzarrj, E., and Cesettj, S. (2007). Wood identification in the cappella palatina ceiling (12th century) in palermo (sicily, italy). *Iawa J.* 28, 109–124. doi: 10.1163/22941932-90001628

Rosa da Silva, N., De Ridder, M., Baetens, J. M., Van den Bulcke, J., Rousseau, M., Martinez Bruno, O., et al. (2017). Automated classification of wood transverse cross-section micro-imagery from 77 commercial central-african timber species. *Ann. For. Sci.* 74, 1–14. doi: 10.1007/s13595-017-0619-0

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv*.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature* 323, 533–536. doi: 10.1038/323533a0

Sharma, V., Yadav, J., Kumar, R., Tesarova, D., Ekielski, A., and Mishra, P. K. (2020). On the rapid and non-destructive approach for wood identification using atr-ftir spectroscopy and chemometric methods. *Vibrational Spectrosc.* 110, 103097. doi: 10.1016/j.vibspec.2020.103097

Shrikumar, A., Greenside, P., and Kundaje, A. (2017). "Learning important features through propagating activation differences," in *International conference on machine learning (PMLR)*. 3145–3153.

Sugiarto, B., Prakasa, E., Wardoyo, R., Damayanti, R., Dewi, L. M., Pardede, H. F., et al. (2017). "Wood identification based on histogram of oriented gradient (hog) feature and support vector machine (svm) classifier," in *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE) (IEEE)*. 337–341.

Sun, Y., Lin, Q., He, X., Zhao, Y., Dai, F., Qiu, J., et al. (2021). Wood species recognition with small data: a deep learning approach. *Int. J. Comput. Intell. Syst.* 14, 1451–1460. doi: 10.2991/ijcis.d.210423.001

Swirszcz, G., Abe, N., and Lozano, A. C. (2009). Grouped orthogonal matching pursuit for variable selection and prediction. *Adv. Neural Inf. Process. Syst.* 22.

Torfi, A., Fox, E. A., and Reddy, C. K. (2022). Differentially private synthetic medical data generation using convolutional gans. *Inf. Sci.* 586, 485–500. doi: 10.1016/j.ins.2021.12.018

Tou, J. Y., Lau, P. Y., and Tay, Y. H. (2007). "Computer vision-based wood recognition system," in *Proceedings of International workshop on advanced image technology*. 197–202.

Vellido, A., Martín-Guerrero, J. D., and Lisboa, P. J. (2012). "Making machine learning models interpretable," in *ESANN (Citeseer)*, Vol. 12. 163–172.

von Arx, G., Carrer, M., Crivellaro, A., De Micco, V., Fonti, P., Lens, F., et al. (2021). Q-net–a new scholarly network on quantitative wood anatomy. *Dendrochronologia* 70, 125890.

Von Arx, G., Crivellaro, A., Prendin, A. L., Čufar, K., and Carrer, M. (2016). Quantitative wood anatomy–practical guidelines. *Front. Plant Sci.* 7, 781.

Wheeler, E. A. (2011). Inside wood–a web resource for hardwood anatomy. *Iawa J.* 32, 199–211. doi: 10.1163/22941932-90000051

Wheeler, E. A., and Baas, P. (1998). Wood identification-a review. *IAWA J.* 19, 241–264. doi: 10.1163/22941932-90001528

Wheeler, E. A., Baas, P., and Gasson, P. E. (1989). Iawa list of microscopic features for hardwood identification. *IAWA Journal (International Association of Wood Anatomists)* 10 (3), 219–332.

Wiedenhoeft, A. C., Simeone, J., Smith, A., Parker-Forney, M., Soares, R., and Fishman, A. (2019). Fraud and misrepresentation in retail forest products exceeds us forensic wood science capacity. *PloS One* 14, e0219917. doi: 10.1371/journal.pone.0219917

Xiao, Y., Wang, H., and Xu, W. (2014). Parameter selection of gaussian kernel for one-class svm. *IEEE Trans. cybernetics* 45, 941–953. doi: 10.1109/TCYB.2014.2340433

Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. *Adv. Neural Inf. Process. Syst.* 32.

Yadav, A. R., Anand, R. S., Dewal, M., and Gupta, S. (2014). "Analysis and classification of hardwood species based on coiflet dwt feature extraction and weka workbench," in *2014 International Conference on Signal Processing and Integrated Networks (SPIN) (IEEE)*. 9–13.

Yadav, A. R., Anand, R. S., Dewal, M., and Gupta, S. (2015a). Hardwood species classification with dwt based hybrid texture feature extraction techniques. *Sadhana* 40, 2287–2312. doi: 10.1007/s12046-015-0441-z

Yadav, A. R., Anand, R. S., Dewal, M., and Gupta, S. (2015b). Multiresolution local binary pattern variants based texture feature extraction techniques for efficient classification of microscopic images of hardwood species. *Appl. Soft Computing* 32, 101–112. doi: 10.1016/j.asoc.2015.03.039

Yadav, A. R., Dewal, M., Anand, R. S., and Gupta, S. (2013). "Classification of hardwood species using ann classifier," in *2013 Fourth national conference on computer vision, pattern recognition, image processing and graphics (NCVPRIPG) (IEEE)*. 1–5.

Yang, Z., Li, L., and Luo, W. (2022a). Pdnet: improved yolov5 nondeformable disease detection network for asphalt pavement. *Comput. Intell. Neurosci.* 2022. doi: 10.1155/2022/5133543

Yang, Z., Ni, C., Li, L., Luo, W., and Qin, Y. (2022b). Three-stage pavement crack localization and segmentation algorithm based on digital image processing and deep learning techniques. *Sensors* 22, 8459. doi: 10.3390/s22218459

Yuliastuti, E. Suprijanto, and Sasi, S. R. (2013). "Compact computer vision system for tropical wood species recognition based on pores and concentric curve," in *2013 3rd International Conference on Instrumentation Control and Automation (ICA) (IEEE)*. 198–202.

Zhang, J., Nieminen, K., Serra, J. A. A., and Helariutta, Y. (2014). The formation of wood and its control. *Curr. Opin. Plant Biol.* 17, 56–63. doi: 10.1016/j.pbi.2013.11.003

Zhang, Q.-s., and Zhu, S.-C. (2018). Visual interpretability for deep learning: a survey. *Front. Inf. Technol. Electronic Eng.* 19, 27–39. doi: 10.1631/FITEE.1700808

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 18 July 2023
DOI 10.3389/fpls.2023.1224884

Check for updates

OPEN ACCESS

EDITED BY
Zhenghong Yu,
Guangdong Polytechnic of Science and Technology, China

REVIEWED BY
Sijia Yu,
Rutgers, The State University of New Jersey- Busch, United States
Danfeng Hong,
Aerospace Information Research Institute (CAS), China

*CORRESPONDENCE
Jiong Mu
jmu@sicau.edu.cn

RECEIVED 18 May 2023
ACCEPTED 29 June 2023
PUBLISHED 18 July 2023

CITATION
Shuai L, Chen Z, Li Z, Li H, Zhang B, Wang Y and Mu J (2023) Real-time dense small object detection algorithm based on multi-modal tea shoots. *Front. Plant Sci.* 14:1224884. doi: 10.3389/fpls.2023.1224884

COPYRIGHT
© 2023 Shuai, Chen, Li, Li, Zhang, Wang and Mu. 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 dense small object detection algorithm based on multi-modal tea shoots

Luyu Shuai[1,2], Ziao Chen[3], Zhiyong Li[1,2], Hongdan Li[1,2], Boda Zhang[1,2], Yuchao Wang[4] and Jiong Mu[1,2]*

[1]College of Information Engineering, Sichuan Agricultural University, Ya'an, China, [2]Ya'an Digital Agricultural Engineering Technology Research Center, Sichuan Agricultural University, Ya'an, China, [3]College of Law, Sichuan Agricultural University, Ya'an, China, [4]College of Mechanical and Electrical Engineering, Sichuan Agricultural University, Ya'an, China

Introduction: The difficulties in tea shoot recognition are that the recognition is affected by lighting conditions, it is challenging to segment images with similar backgrounds to the shoot color, and the occlusion and overlap between leaves.

Methods: To solve the problem of low accuracy of dense small object detection of tea shoots, this paper proposes a real-time dense small object detection algorithm based on multimodal optimization. First, RGB, depth, and infrared images are collected form a multimodal image set, and a complete shoot object labeling is performed. Then, the YOLOv5 model is improved and applied to dense and tiny tea shoot detection. Secondly, based on the improved YOLOv5 model, this paper designs two data layer-based multimodal image fusion methods and a feature layerbased multimodal image fusion method; meanwhile, a cross-modal fusion module (FFA) based on frequency domain and attention mechanisms is designed for the feature layer fusion method to adaptively align and focus critical regions in intra- and inter-modal channel and frequency domain dimensions. Finally, an objective-based scale matching method is developed to further improve the detection performance of small dense objects in natural environments with the assistance of transfer learning techniques.

Results and discussion: The experimental results indicate that the improved YOLOv5 model increases the mAP50 value by 1.7% compared to the benchmark model with fewer parameters and less computational effort. Compared with the single modality, the multimodal image fusion method increases the mAP50 value in all cases, with the method introducing the FFA module obtaining the highest mAP50 value of 0.827. After the pre-training strategy is used after scale matching, the mAP values can be improved by 1% and 1.4% on the two datasets. The research idea of multimodal optimization in this paper can provide a basis and technical support for dense small object detection.

KEYWORDS

dense small object detection, multimodal image fusion, RGB-D-IR, scale matching, frequency domain, attention mechanism, tea shoots

1 Introduction

In recent years, the aging trend of agricultural labor has significantly intensified, and the difficulty in recruiting and expensive labor has limited the development of the tea industry (Han et al., 2014). The manual picking of premium tea accounts for about 60% of the labor used for managing the whole tea plantation, while excellent high-grade tea is picked with delicate leaf tips that grow in different positions, postures, and densities, making it difficult for machine picking especially in the unstructured environment with wind and light changes (Xu et al., 2022). Thus, it is essential to study intelligent tea-picking technology to promote the development of the tea industry. The key to realizing automated tea picking is the accurate identification of tea shoots. In recent years, with the development and application of computer technology, the accurate identification of tea shoots based on image processing has become a research hotspot (Lin et al., 2019).

Since there are obvious color differences between tea shoots and old leaves and tree trunks, color features can be used to extract shoot regions in the image, so the early research on tea shoot segmentation is mainly based on color features. The primary process of traditional image processing algorithms based on color space involves image pre-processing, color feature selection, segmentation, and other steps (Bojie et al., 2019). To further address the issue that tea leaf segmentation under natural conditions is easily affected by the external environment, such as old leaves, branches, and soil, and obscured and overlapping tea leave. Machine learning methods have been introduced for identification by extracting and synthesizing various feature sample data for training, and standard methods for tee shoot identification are developed based on features such as color, texture, and shape, combined with the use of K-mean clustering, support vector machine methods, Bayesian discriminant methods, and cascade classifiers. Recognition methods based on traditional machine vision rely on image pre-processing and data conversion, and unreasonable pre-processing will significantly affect the accuracy of the model (Karunasena and Priyankara, 2020) (Li et al., 2021).

The algorithm based on deep learning has high accuracy, providing a basis for studying intelligent tea shoot-picking equipment in complex backgrounds. To alleviate the influence of a complex environment on the performance of the detection model, (Xiaoxiao et al., 2019) employed a pre-segmentation method and then used the improved YOLO series of medium and large-scale network models to detect tea shoots with an average accuracy of 84.2%. To promote the deployment of models for detecting tea shoots to picking leaf tips, lightweight models have received much attention from researchers. (Xu et al., 2022) exploited the fast detection capability of YOLOv3 and the high-precision classification capability of DenseNet201 through a cascaded network to detect tea shoots accurately. Although the above methods have relatively high accuracy, robustness, and generalization performance, they are difficult to detect adequate tea shoots in complex environments on low arithmetic devices in farmland due to the high dependence of deep learning network models on arithmetic power. Thus, researchers have investigated the accuracy, speed, and lightness of model detection simultaneously (Cao et al., 2022). proposed a tea shoot detection algorithm that fuses GhostNet and YOLOv5; (Li Y. et al., 2022) designed a YOLOv3-SPP deep learning algorithm based on channel and layer pruning, which reduced the number of parameters, model size, and inference time while achieving efficient and accurate tea shoot detection. Note that few studies have focused on crop objects that are dense and minutely difficult. However, in the study of small target detection problems, remote sensing image target detection has achieved excellent results. (Wu et al., 2019) presented a detector called ORSIm, which effectively improves the accuracy of small target detection in optical remote sensing images by integrating different channel features, feature learning, and fast image pyramid matching and enhancement strategies. To reduce the difficulty in infrared small target detection, (Wu et al., 2023) proposed an interactive cross-notice nested U-Net network called UIU-Net. However, UIU-Net models infrared small target detection as a semantic segmentation problem, which increases the cost of labeling. Therefore, this study improves the detection performance of dense and tiny tea shoots by improving the target detection model and adopting migration learning techniques.

The above studies took only RGB images as the input to the network. Nevertheless, in an unstructured environment, a single sensor provides limited information to detect shoot targets under various difficulties, such as different lighting conditions, the similar color of tea shoots to the background, the small size of tea shoots, dense tea shoots, overlapping tea shoots, branch and leaf occlusion, as well as different poses. To overcome these difficulties, the approach of using multimodal data can be adopted since there is a certain complementarity and consistency between multimodal information. Although RGB images can reflect features such as color, brightness, and texture of objects, they can only provide two-dimensional (2D) details. With the further development of image acquisition devices, the availability of multimodal data for object detection in agricultural environments has increased greatly, such as depth images, infrared images, etc. (Sun et al., 2022). Depth images contain information about the distance from the object to the sensor, which can reflect the depth and three-dimensional (3D) morphology of the object. So, depth images have more unique edge features and shape features that can be exploited to better distinguish between foreground and background. Meanwhile, infrared images collect information about the heat distribution of the object, which can reflect the temperature and thermal radiation characteristics of the object. Most importantly, depth and infrared images are less affected by illumination and viewing angle, and they can be used to perform stable target detection in complex environments. Thus, in recent years, research work has been devoted to using multimodal information to improve the performance of crop detection. For instance, (Tao and Zhou, 2017) extracted improved 3D descriptors (Color-FPFH) that incorporate color features and 3D geometric features from pre-processed point clouds to obtain richer feature information to enhance the accuracy of detecting apples. (Gan et al., 2018) designed an algorithm for green citrus fruit detection by

integrating image alignment, information fusion, fruit classification, and detection into a single step to realize real-time detection. Experimental results indicate that the fusion of color and thermal images can effectively improve the detection of unripe green citrus fruits. Additionally, some studies use depth information to exclude complex backgrounds in agricultural environments to enhance the detection performance of target objects in RGB images. For example, (Lin et al., 2019) presented a depth filter and Bayesian classifier-based image segmentation method based on red-green-blue-depth (RGB-D) images to remove complex backgrounds. This improves citrus detection and localization accuracy in a natural outdoor orchard environment. (Fu et al., 2020) developed a faster R-CNN-based apple detection method using RGB images and depth features in a dense leafy wall tree. The background was first eliminated using a depth threshold of 1.2 m to obtain the foreground RGB image. Then, the detection results of the original RGB image and the foreground RGB image were compared by using two different pre-trained network architectures (ZFNet and VGG16). The results demonstrated that removing the background tree using the depth filter can improve the fruit detection accuracy by 2.5%.

Methods for effective fusion methods of multimodal information have attracted much attention. In multimodal image target detection, the fusion methods for different information can be usually divided into three types: data layer fusion, feature layer fusion, and decision layer fusion. First, data layer fusion methods treat multimodal data as indistinguishable multichannel data and can exploit the inherent complementarity between different modalities to supplement the incomplete information in the input stage. For instance, (Gené-Mola et al., 2019) collected RGB images, depth images, and infrared images of apples simultaneously and performed range-correction on the signal intensity to solve the signal attenuation problem. The detection of apples was achieved by applying the Faster R-CNN model to five channels of input images (color (RGB), depth (D), and distance-corrected intensity signal (S)). The results indicate that the F1-score improves by 4.46% when depth and range-corrected intensity channels are added, and an F1-score of 0.898 and an AP of 94.8% are obtained when all channels are used. (Liu et al., 2019) proposed a method to fuse aligned RGB images, NIR images, and deep convolutional neural networks for kiwifruit detection. In their study, two different fusion methods were investigated: image fusion (fusing RGB and infrared images on the input layer) and feature fusion (combining the feature maps of two VGG16 networks with separate input RGB and NIR images). The results showed that the highest AP value of 90.7% was achieved by using the image fusion method. (Rong et al., 2023) applied a multimodal (RGB images and depth images) data fusion approach to optimize the input of YOLOv5 to reduce the effect of background on false tomato recognition and improved the recall of unripe tomatoes with a detection accuracy of 97.9% by the improved YOLOv5-4D. However, the crude data layer fusion method may result in information redundancy and noise propagation with limited enhancement effect, affecting the quality and accuracy of the fused data. The second type of fusion method, i.e., the feature layer fusion method, inputs multimodal images into parallel branches, extracts independent features at different scales in different modes, and then fuses the features. For instance, (Wu et al., 2021) developed a new multimodal remote sensing image classification network called CCR-Net. CCR-Net uses features from different modalities obtained by a CNN extractor and fuses them more compactly, allowing better processing and analysis of multimodal remote sensing data. (Hong et al., 2021) designed a new supervised algorithm for GCNs, called miniGCNs. miniGCNs jointly uses CNNs and GCNs to extract more diverse and differentiated feature representations for hyperspectral image classification tasks. However, both are based on image classification tasks. (Sun et al., 2022) proposed a noise-tolerant RGB-D feature fusion network for outdoor fruit detection to integrate RGB feature information, depth feature information, and an attention-based fusion module to adaptively fuse multimodal features to remove the adverse effects of depth noise and focus perception on the essential parts of the features. The proposed NT-FFN achieves an AP50 value of 95.4%. However, the inappropriate feature fusion approach in the feature layer fusion method may increase the difficulty of model learning and aggravate the imbalance of the network learning modality. The third type of feature fusion method, i.e., the decision layer fusion method, fuses the detection results of the last stage. For example, (Tu et al., 2018) adopted a faster region-based convolutional neural network (Faster R-CNN) to detect passion fruit for color images and depth images, respectively, and the two detection results based on RGB images and depth images were combined to improve the detection performance. (Lin et al., 2022) developed a regression network with multi-branch architecture to extract and fuse RGB, depth, and geometric features easily. The proposed post-fusion architecture significantly improved the fresh weight detection accuracy of lettuce shoots at different growth periods. However, the decision-level fusion method may consume a lot of computational resources due to the repeated computation of other multimodal branches, and the process learns the features of individual modalities independently without considering the correlation between different modal information. Therefore, to realize efficient real-time detection of tea shoots in an agricultural intelligent picking environment, this study investigates two data layer-based multimodal information fusion methods and a feature layer-based multimodal information fusion method, respectively. Meanwhile, a lightweight frequency domain attention mechanism module is designed for the feature layer fusion method to effectively fuse feature information across modalities.

To efficiently detect small targets of dense tea shoots in complex environments, this study improves the architecture of the YOLOv5 target detection model. Additionally, to make up for the deficiency of RGB image-based tea shoot detection, this study designs two data layer-based multimodal fusion methods and a feature layer-based multimodal fusion method based on the YOLOv5 model and designs a cross-modal fusion module based on frequency domain and attention mechanism. The main contributions of this study are summarized below:

1. A tea image dataset of the natural environment is constructed. It contains aligned RGB images, depth

images, and infrared images; the RGB images are annotated with tea shoot objects.
2. The architecture of the YOLOv5 model is modified and adjusted to improve the detection performance of the model for dense and tiny tea shoots.
3. The scale matching method is optimized based on the object scale. The generalization and robustness of the tea shoot detection model are improved by applying transfer learning techniques.
4. Two multimodal fusion methods based on the data layer and one multimodal fusion method based on the feature layer are investigated. Meanwhile, a cross-modal fusion module based on frequency domain and attention mechanism is designed to learn complementary information by adaptively focusing key regions in intra- and inter-modal frequency domain dimension and channel dimension to improve the performance of the tea shoot detector.

2 Materials and methods

2.1 Data

2.1.1 Data acquisition

The dataset used in this study was obtained at the National Tea Tree Breeding Farm, Mengdingshan Tea Modern Agricultural Park, Ya'an City, Sichuan Province, China. The images were taken on the evening of 09/03/2023 and 19/03/2023, the prime time for famous tea harvesting. This study took Microsoft Kinectv2 as the image acquisition device, which integrates an RGB camera and a depth sensor that works following the TOF principle. The sensor provides three types of data: a color image, a depth image that can generate a 3D point cloud of the scene, and a received infrared backscattered intensity image.

In the data acquisition process, the Microsoft Kinect v2 depth camera was fixed on a triangular stand, with one end of the camera being connected to 220V outdoor mobile power and the other end being connected to a laptop *via* USB 3.0. The depth image, infrared image, color, and depth information aligned low-resolution image were captured simultaneously on the computer by calling the API of PyKinectV2 (Kinect/PyKinect2). First, a depth image, an infrared image, and an aligned image (RGB) with both color and depth information were captured simultaneously; then, they were resized to 512×424 pixels; finally, the images were mirrored and inverted separately and saved. The RGB image was stored in 24 bits, the infrared image in 16, and the depth image in 8. The depth camera was placed vertically from 0.5-1.0 m away from the top of the tea. To reduce the effect of bright light on sensor performance under outdoor conditions, all data were captured from 5:00 to 7:00 PM on an overcast day. Table 1 presents the parameters and specifications of the equipment used in the data acquisition process.

2.1.2 Data preparation

A multimodal image dataset consisting of RGB, infrared, and depth images was obtained after data acquisition, each with a resolution of 512×424 pixels. The original image schematic is shown in the first row of Figure 1. Since the depth sensor has a larger vertical field of view than the color camera, the RGB, infrared, and depth images were cropped by removing the bottom and top images that do not provide RGB information, and the image resolution became 521×360 pixels, as shown in the second row of Figure 1.

In the data annotation process, tea shoots were manually annotated using the COCO Annotator (Stefanics et al., 2022)

TABLE 1 Acquisition equipment specifications.

Device	Specifications	Parameter
RGB-D Sensor	Manufacturer and model	Microsoft Kinectv2
	RGB channel resolution (pixels)	1920 × 1080
	RGB channel field-of-view (FOV)	84.1° × 53.8°
	IR and Depth channel resolution (pixels)	512 × 424
	IR and Depth channel FOV	70° × 60°
	Working range (m)	0.5–8
Notebook Computer	Manufacturer and model	ASUS
	Processor	AMD Ryzen 7 6800H with Radeon Graphics 3.20 GHz
	RAM	16.0 GB
Outdoor mobile power	Manufacturer and model	St. Xinlong
	Size	255×165×145mm
	Power capacity	90000mAh
	Output voltage	220V

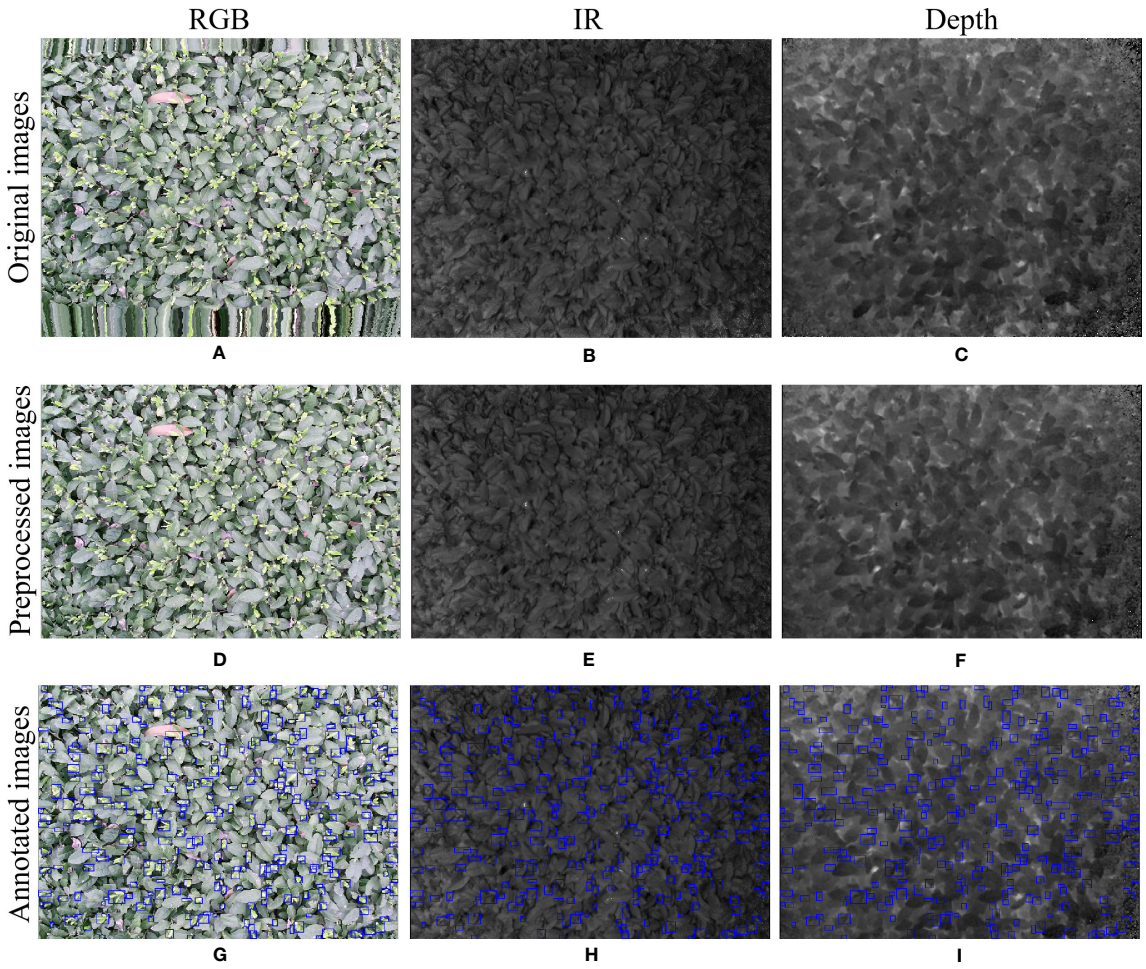
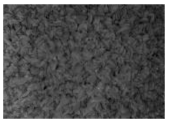
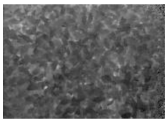
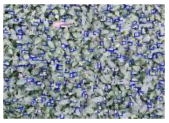
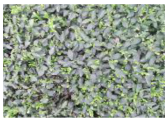
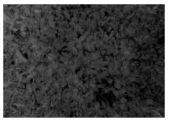
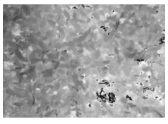
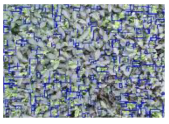


FIGURE 1
RGB images, IR images, and Depth images are represented from left to right. **(A-C)** captured original image; **(D-F)** cropped image; **(G-I)** annotated image. .

online annotation software for RGB images only. To simulate the complexity of tea shoot growth in a natural environment and reflect the effectiveness of the detector, tea shoots with less than 75% occlusion and tiny tea shoots were annotated with absolute pixels larger than 2×2 pixels. Each image annotation process took 0.5-0.6 hours, and each image contains 200-400 tea shoot targets with an absolute scale of about 30×30 pixels. To achieve a low manual annotation cost and investigate the effect of multimodal images on the performance of tea shoot detection, RGB, infrared, and depth images shared a common set of labels: the annotation result on RGB images. An example of the image after mapping the labeling results to infrared and depth images is shown in the third row in Figure 1.

This study collected 100 sets of multimodal image data on 09/03/2023 and 19/03/2023, respectively, 200 sets in total. Each dataset contains one RGB, infrared, and depth image, as well as the corresponding labels. Table 2 shows the distribution of the datasets

TABLE 2 Distribution of data sets and image examples.

Datasets		Collection time	Number	RGB	IR	Depth	Label
Dataset3	Dataset1	2023.03.09	100				
	Dataset2	2023.03.19	100				

and example images. Dataset1 and Dataset2 represent the datasets collected on 09/03/2023 and 09/03/2023, respectively. Dataset 3 represents the set of Dataset 1 and Dataset 2 datasets.

Since the color camera has a more significant horizontal field of view than the depth sensor, the original high-resolution color image (1920×1080 pixels) and the RGB image (521×360 pixels) used in this study were unaligned, and this study aimed to investigate the detection method and model for dense small targets in low-resolution images. Therefore, this study only used the low-resolution RGB images and the aligned infrared and depth images as experimental data. In future work, we will explore the problem of image alignment and super-resolution-assisted small target detection based on high-resolution and low-resolution images, and the original high-resolution color images will be used.

2.2 Methods

2.2.1 YOLOv5s baseline and improvement architecture

YOLO (You Only Look Once) (Redmon et al., 2016) is a classic single-stage target detection network. The YOLOv5 (Jocher et al., 2022) model is widely used in various target detection tasks because of its flexibility and versatility. It uses CSPNet (Cross Stage Partial Network) (Wang et al., 2020) as the backbone to extract feature information and SPP (Spatial Pyramid Pooling) (He et al., 2015) to extract multi-scale depth features and then fuse the features at different scales through a feature pyramid constructed by PANet (Path Aggregation Network) (Liu et al., 2018), and the final results are output through three detection heads P3, P4, and P5. The depth and width of the YOLOv5 model depend on the bottleneck layer and several convolutional kernels, whereas the YOLOv5s model has a small size and fast inference speed, which is beneficial for real-time target detection in realistic scenarios. This is the reason why this study chooses YOLOv5s as the baseline. However, since the baseline model is usually designed for detecting medium and large targets, there are some limitations in the detection of small objects. YOLOv5s mainly includes the Focus layer, the design of the CSP1_n module, the number of stacks, and the PANet architecture. This study will elaborate on their limitations and the corresponding improvement measures for dense and tiny tea shoot detection. Figures 2A, B show the architectures of the YOLOv5 model and our improved YOLOv5s_improve model, respectively, and Figure 2C shows the detailed construction of the modules that may be included in these two models.

The limitations and improvements are analyzed as follows:

1. From Focus to Conv: Focus is a lightweight convolutional layer. To reduce computational cost and speed up network training and inference, the Focus layer divides the input into four parts; convolutional operations are performed on each part separately, and the results are stacked finally to form the output feature map. However, this approach may sacrifice the accuracy of small target detection. Therefore, to better capture the feature information of small targets, this study uses replaces the Focus layer with a superficial Conv layer to increase the perceptual field of the model and the feature representation.
2. From "3693" to "8833": The backbone of YOLOv5 used convolution with a step size of 2 in the early stage to halve the feature size. As the network deepens, the feature size retained for multi-scale target detection is much smaller than the size of the original input image. This low-resolution feature map does not contain information that can be used to reliably distinguish tiny objects. (Ning et al., 2023) effectively improved the performance of small object detection by increasing the shallow layers (the convolutional layers in the high-resolution stage) in the ResNet (He et al., 2016) and HRNet (Sun et al., 2019), thereby using fewer convolutional layers in the later stages of the network. The experimental results indicated that the early downsampling leads to information loss and difficulty in representing the features of small targets. Similarly, the number of CSP1_n modules in each phase of the YOLOv5 backbone network is modified to allocate more resources to handle higher-resolution features, and the number of CSP1_n modules in the post-backbone stage of the network is reduced to not introduce additional computational burden. The original YOLOv5 backbone contains four CSP1_n modules, and the number of modules is 3, 6, 9, and 3 in order. Through several experimental adjustments, this study finds that the optimal number of CSP1_n modules is 8, 8, 3, and 3 in order.
3. From CSP2 to C3_DSConv: In the CSP2 module of the neck, the standard convolution operation may cause the small object model of tea shoots to overfit and introduce an enormous computational burden. (Nascimento et al., 2019) proposed a flexible quantized convolution operator DSConv that uses inexpensive integer operations instead of single-precision operations while maintaining the kernel weights and output on the probability distribution. This study replaces the standard convolution in the neck CSP2 module with DSConv to ensure the lightweight and real-time characteristics of the tea shoot detection model.
4. From PANet to FPN: The main idea of PANet is to obtain higher-level semantic information through aggregation and transfer, but it requires a lot of computational resources and time and may lead to information loss and model overfitting, and PANet focuses on the improvement of detection accuracy of medium and large targets. FPN (Feature Pyramid Network) (Lin et al., 2016) obtains better scale adaptation and semantic information through feature transfer and fusion, which helps to preserve the delicate features and information required for small object detection and effectively reduces the complexity of the model. Thus, this study replaces the PANet structure with FPN.

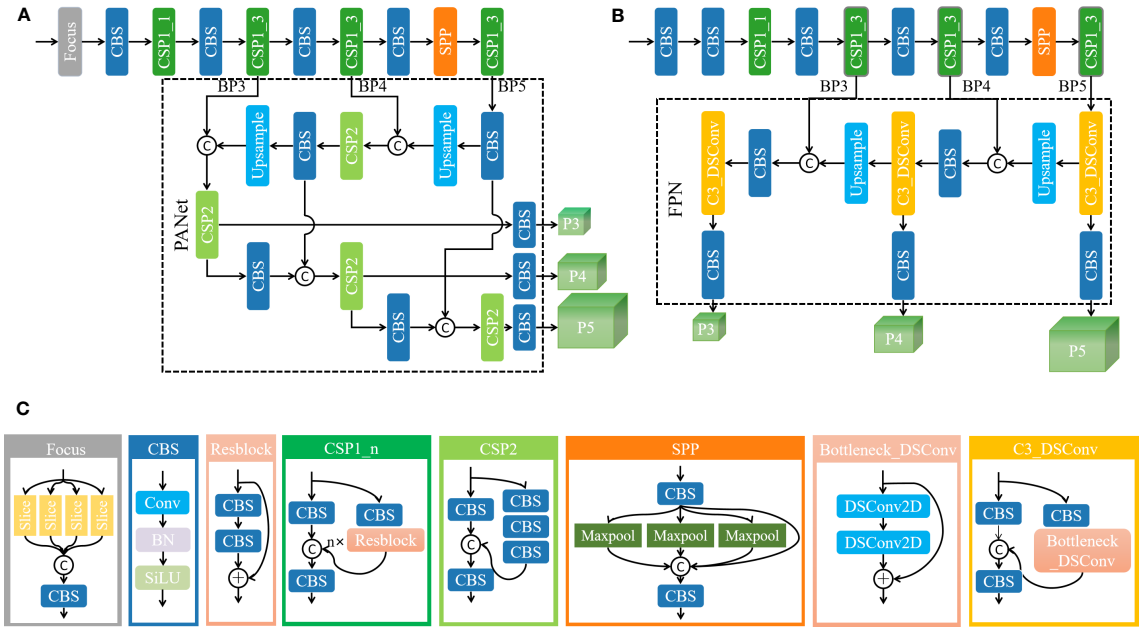


FIGURE 2
Model architecture diagram and detailed module construction diagram. **(A)** YOLOv5s model architecture diagram; **(B)** YOLOv5s_improve model architecture diagram; **(C)** detailed construction of the modules that may be included in the model.

2.2.2 Multimodal object detection architecture

2.2.2.1 Multimodal image object detection

To fully utilize the complementary information between RGB, infrared, and depth images of tea shoots to enhance the ability of the model to detect and localize tea shoots, two data layer-based fusion methods and a feature layer-based fusion method is established in this study. Besides improving the quality of intra-modal and inter-modal information fusion, a simple and effective FFA module is designed in this study by using the feature layer-based data fusion method. The input and the backbone of the models of the three fusion methods in this study are illustrated in Figure 3.

Method 1 uses a simple data layer fusion approach. As shown in method (A) in Figure 3, through several repetitive comparative experiments, the best weighting coefficients are first derived for RGB, infrared, and depth images, and they are 0.6, 0.2, and 0.2, respectively. Secondly, the RGB, infrared, and depth images are fused by simple pixel-level summation with the best weighting coefficients, respectively. Then, the synthesized images are fed into the single-stream object detection backbone for feature extraction. Finally, BP3, BP4, and BP5 features are provided to the model head for detection.

Method 2 uses data layer fusion based on channel mapping. Again, the best weighting coefficients are derived for infrared and depth images by repeated experiments with multiple comparisons of 0.5 and 0.5, respectively. Then, the infrared and depth images are fused by simple pixel-level summation with the best weighting coefficients. The obtained image A is taken as the fourth channel of the image to obtain a four-channel RGBA image by stitching it with the color RGB image. Next, the RGBA image is fed into the designed 4-channel single-stream object detection backbone for feature extraction, and finally, BP3, BP4, and BP5 features are provided to the model head for detection. The details are shown in method (B) in Figure 3.

Method 3 uses feature layer fusion. The infrared and depth images are first stitched into a single three-channel image (D_IR_IR) to preserve as much information as possible under each modality; then, the stitched and colored RGB images are fed into the designed dual-stream object detection backbone to extract features, and finally, BP3, BP4, and BP5 features are provided to the model head for detection. The detailed design of YOLOv5s-Multimodal, a multimodal image fusion architecture based on feature layers, is presented in Figure 3C. In the YOLOv5s_Multimodal model, this study uses YOLOv5s_improve as the backbone of two branches, but the parameters in the two backbones are not shared. The same backbone structure is used to extract features from D_IR_IR and RGB images under each modality. In the intermediate stage of the backbone, the features are fused by the frequency domain-based cross-modal fusion attention module (FFA) to facilitate the interaction and fusion of modalities, and the fused features are fed to the RGB stream and the D_IR_IR stream respectively for feature extraction in depth.

2.2.2.2 Cross-modal fusion attention module based on frequency domain

RGB, infrared, and depth images have their strengths and weaknesses, and their information is usually complementary but contains noise. There are better solutions than simply fusing or processing RGB, infrared, and depth images. However, noisy information can be filtered and calibrated using features from another modality, so this study proposes FFA, and its structure is shown in Figure 4.

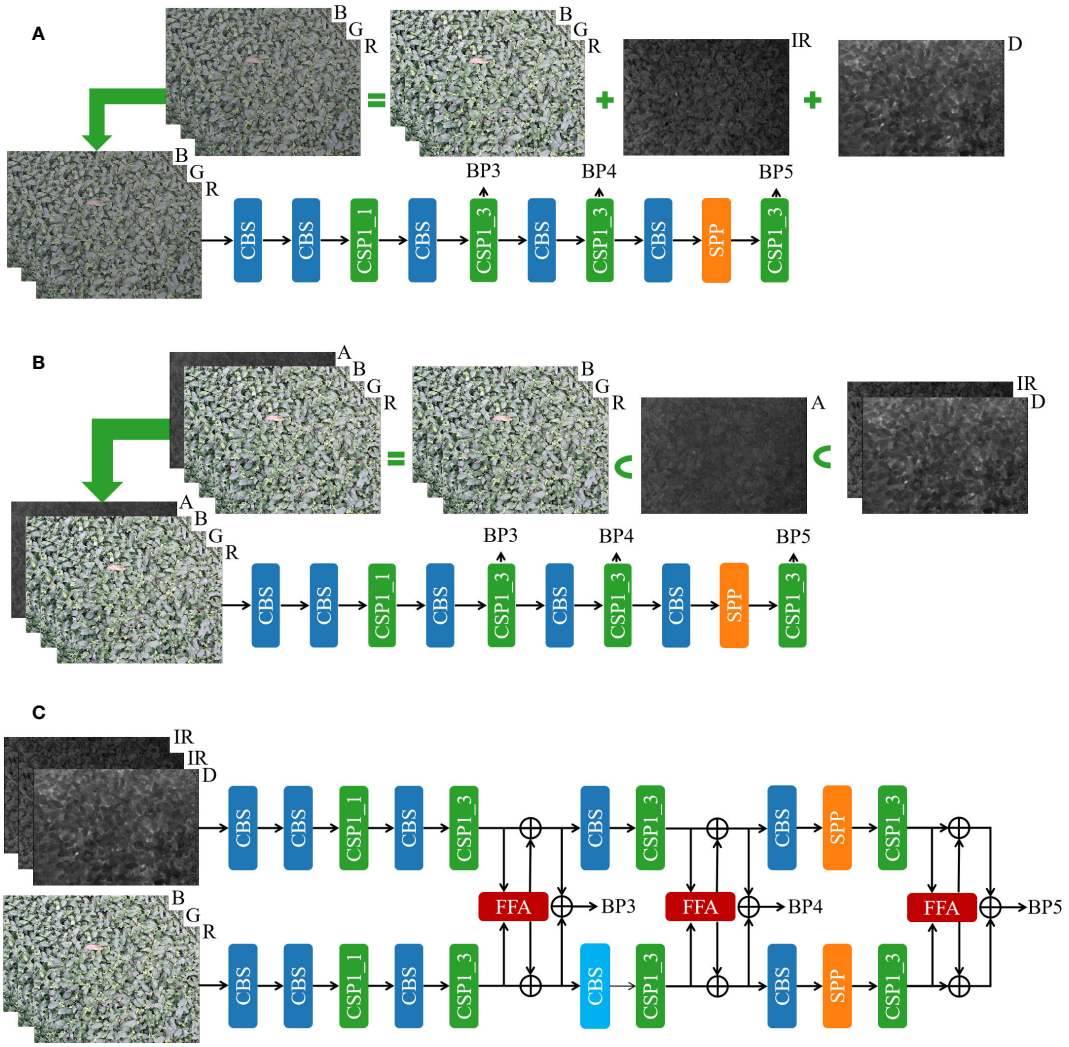


FIGURE 3
Three fusion methods for multimodal images. **(A)** Data layer-based fusion method 1; **(B)** Data layer-based fusion method 2; **(C)** Feature layer-based fusion method.

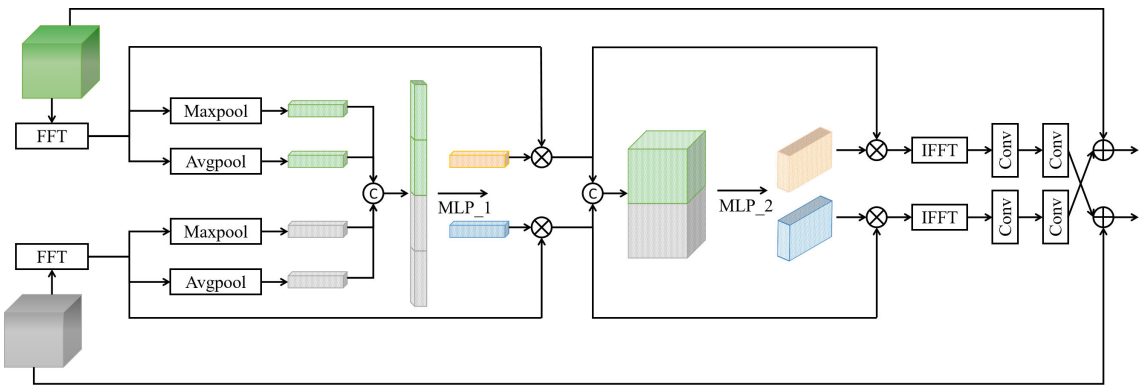


FIGURE 4
Structure of the FFA module.

To reduce expensive computations, improve the inference speed of the model and better preserve the spatial and semantic information of the images, this study chooses to filter, enhance, and fuse the information of different modalities in the frequency domain. To resolve the noise and uncertainty in other modalities and to calibrate and extract the frequency feature information in various modalities, this study infers the attention map along the channel dimension and frequency dimension in turn and then multiplies the attention map with the feature map in the frequency domain to perform adaptive frequency domain feature fusion optimization. To facilitate feature extraction and interaction between modes, this study enhances information interaction between other methods by simple convolution and cross-fusion.

Spatial domain to frequency domain: feature maps F_{rgb} and F_{depth_ir} are respectively converted to F_{f_rgb} and $F_{f_depth_ir}$ in the frequency domain using FFT. Equations (1-2) show the corresponding 2D FFT.

$$F_{f_rgb}(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F_{rgb}(x,y)e^{-j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{1}$$

$$F_{f_depth_ir}(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F_{depth_ir}(x,y)e^{-j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{2}$$

where $F(x,y)$ is a feature map of size $M \times N$, and equations (1) and (2) are evaluated for the discrete variables u and v with $u = 0,1,2,\ldots,M-1$ and $v = 0,1,2,\ldots,N-1$.

Information fusion and enhancement of channel dimensions: First, global pooling operations are performed on the frequency -domain feature maps F_{f_rgb} and $F_{f_depth_ir}$ respectively to obtain global frequency -domain feature information, and both global average pooling and global maximum pooling are used to retain as much information as possible. Then, four resultant vectors are generated and stitched to form a richer frequency -domain feature representation. Next, the frequency -domain feature information is further extracted and fused by the MLP_1 layer. Subsequently, the sigmoid operation is performed to obtain the weights, and the weights are divided into $W^C_{f_rgb}$ and $W^C_{f_depth_ir}$ by the split operation. Finally, the weights are multiplied with the input frequency-domain feature maps F_{f_rgb} and $F_{f_depth_ir}$ to obtain the frequency-domain feature maps $F^C_{f_rgb}$ and $F^C_{f_depth_ir}$, respectively. In this way, the information enhancement and complementation of the channel dimension of RGB and Depth_IR features are realized. The whole process is shown in Equations (3-7).

$$F_{f_rgb_c} = Concat\left(\left(MaxPool(F_{f_rgb})\right),\left(AvgPool\left(F_{f_rgb}\right)\right)\right) \tag{3}$$

$$F_{f_depth_ir_c} = Concat\left(\left(MaxPool(F_{f_depth_ir})\right),\left(AvgPool\left(F_{f_depth_ir}\right)\right)\right) \tag{4}$$

$$W^C_{f_rgb}\ , W^C_{f_depth_ir} = \mathcal{F}_{split}\left(\delta\left(\mathcal{F}_{MLP_1}\left(Concat\left(F_{f_rgb_c}\ , F_{f_depth_ir_c}\right)\right)\right)\right) \tag{5}$$

$$F^C_{f_rgb} = W^C_{f_rgb} \circledast F_{f_rgb} \tag{6}$$

$$F^C_{f_depth_ir} = W^C_{f_depth_ir} \circledast F_{f_depth_ir} \tag{7}$$

where δ represents the Sigmoid operation.

Information fusion and enhancement in the frequency domain: first, the Concat operation is performed on frequency -domain feature maps $F^C_{f_rgb}$ and $F^C_{f_depth_ir}$ to obtain a richer frequency -domain feature representation. Then, after MLP_2 layers, which are two 1×1 convolution and nonlinear transform RELU operations, more features are extracted to obtain a complex frequency -domain feature representation. Next, the sigmoid operation is performed to obtain the weights, and the weights are divided into $W^F_{f_rgb}$ and $W^F_{f_depth_ir}$ by the split operation. Finally, the weights are multiplied with the input frequency-domain feature maps $F^C_{f_rgb}$ and $F^C_{f_depth_ir}$ to obtain the frequency-domain feature maps $F^F_{f_rgb}$ and $F^F_{f_depth_ir}$, respectively. In this way, the information enhancement and complementarity of the frequency dimension of RGB and Depth_IR features are realized. The whole process is shown in Equations (8-11).

$$F^C_{f_rgb_depth_ir} = Concat(F^C_{f_rgb}\ , F^C_{f_depth_ir}) \tag{8}$$

$$W^F_{f_rgb}\ , W^F_{f_depth_ir} = \mathcal{F}_{split}\left(\delta\left(Conv_{1\times1}\left(RELU\left(Conv_{1\times1}\left(F^C_{f_rgb_depth_ir}\right)\right)\right)\right)\right) \tag{9}$$

$$F^F_{f_rgb} = W^F_{f_rgb} \circledast F^C_{f_rgb} \tag{10}$$

$$F^F_{f_depth_ir} = W^F_{f_depth_ir} \circledast F^C_{f_depth_ir} \tag{11}$$

where δ represents the Sigmoid operation.

Frequency domain to spatial domain: IFFT is performed on feature maps $F^F_{f_rgb}$ and $F^F_{f_depth_ir}$ to convert them back to feature maps F^{CF}_{rgb} and $F^{CF}_{depth_ir}$ in the spatial domain, respectively. The corresponding 2D IFFT is shown in Equations (12-13).

$$F^{CF}_{rgb}(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F^F_{f_rgb}(u,v)e^{j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{12}$$

$$F^{CF}_{depth_ir}(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F^F_{f_depth_ir}(u,v)e^{j2\pi(\frac{ux}{M}+\frac{vy}{N})} \tag{13}$$

where $x = 0,1,2,\ldots,M-1$, $y = 0,1,2,\ldots,N-1$.

Re-enhancement of purified information: To obtain a better feature representation, two convolution operations are used to enhance the feature information extracted in the above process, and the information is fed to the RGB stream and Depth_IR stream respectively for the next stage of feature extraction and fusion by cross-fusion. Equation (14-15) shows the purified information re-enhancement operation.

$$F_{rgb} = Conv_{1\times1}\left(Conv_{1\times1}\left(F^{CF}_{depth_ir}\right)\right) \oplus F_{rgb} \tag{14}$$

$$F_{depth_ir} = Conv_{1\times1}\left(Conv_{1\times1}\left(F^{CF}_{rgb}\right)\right) \oplus F_{depth_ir} \tag{15}$$

2.2.3 Objective-based scale matching

The influence of uncontrollable factors in the natural environment, such as light, temperature, and humidity, leads to different growth states of tea shoots. Particularly, tea shoots proliferate from early March to early April, as shown in Dataset1 and Dataset2, which exhibit large differences in length, volume, posture, and color, although only ten days. This poses a challenge to the generalizability and robustness of the detection model. Figure 5 shows the number and relative scale distribution of tea shoot objects in the two datasets.

From Figure 5, it can be observed that: In Dataset1, the total number of tea shoots exceeds 20,000, the distribution of tea shoots is relatively dense, the width and height of tea shoots are similar in attitude, and the relative scale of over 90% of the tea shoots is less than 5%. In Dataset2, the total number of tea shoots is close to 25,000, the distribution of tea shoots is very dense, the width, height, and posture of tea shoots are different, and the relative scale of over 90% of tea shoots is less than 10%. Overall, both datasets are dense, making it challenging to find targets. The difference between them is that Dataset1 has fewer samples and more minor relative scale differences, while Dataset2 has more samples and larger relative scale differences.

(Yu et al., 2020) found that the problem of scale mismatch reduces the accuracy of feature representation and detection models, and a smaller dataset may lead to model overfitting. To improve the generalization and robustness of the detector for detecting tea shoots of different periods under the condition of small samples, this study uses a simple scale-matching method combined with migration learning techniques to improve the detection performance of the model. The targets in Dataset2 are scaled to align with the relative scales of the targets in Dataset1. Then, the best weights obtained from training using the aligned dataset are used as pre-training weights to guide the detection model to fine-tune the parameters on Dataset1 to improve the detection capability of the detector for Dataset1. This

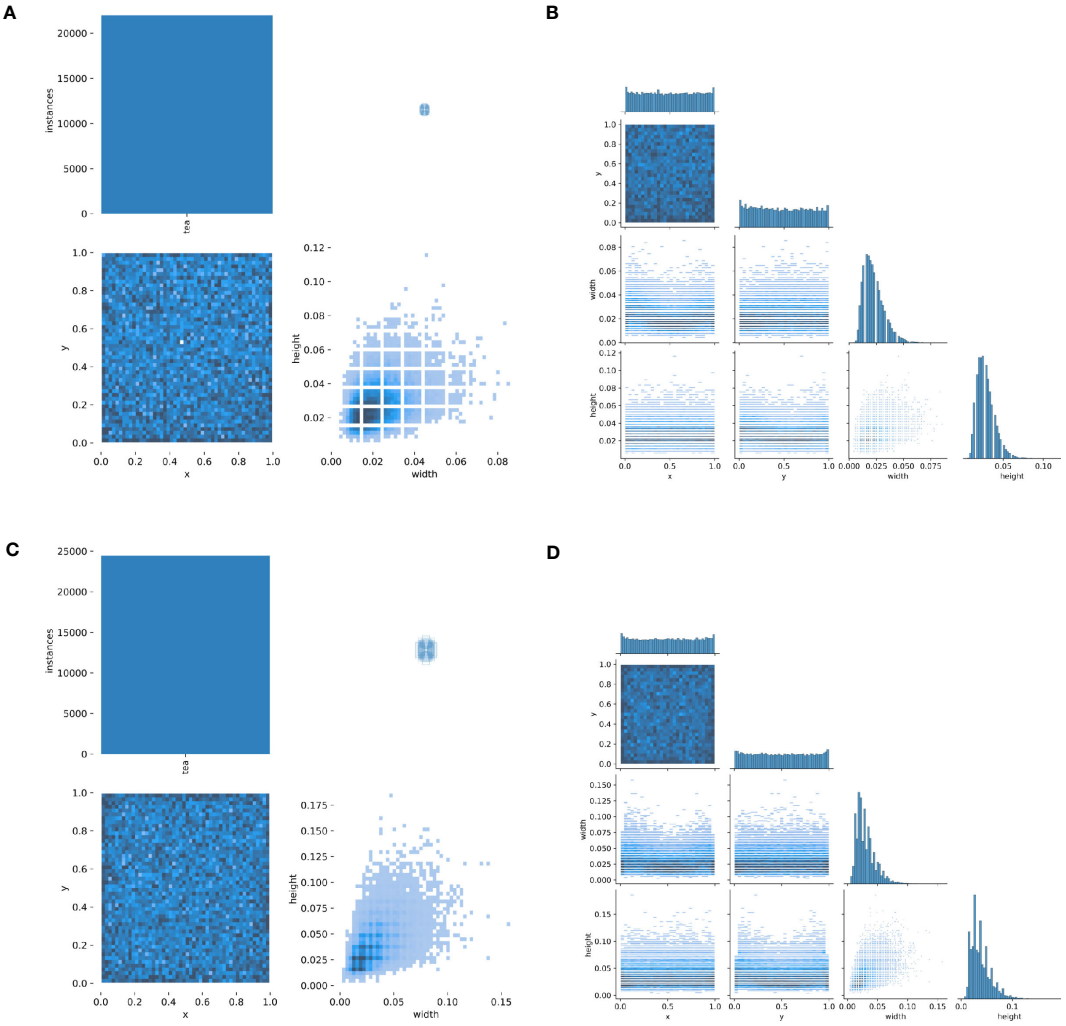


FIGURE 5
Distribution of tea shoot objects in Dataset1 and Dataset2 datasets. **(A)** the total number of targets and the relative width and height scales of target boxes in Dataset1; **(B)** the total number of objects and the relative width and height scales of object boxes in Dataset2; **(C)** the relative width and height scales and distribution of objects in Dataset1; **(D)** the relative width and height scales and distribution of objects in Dataset2.

facilitates the distribution of features between the pre-trained dataset of the aligned network and the dataset learned by the detector, enabling the model to better utilize the information at small scales.

The specific procedure is as follows: first, the average scale (s_1, s_2) of the two datasets Dataset1 and Dataset2, and their distributions are calculated by statistical data methods, and the scale scaling factors (a_{12}, a_{21}) between the two datasets are obtained. Then, search, judgment, and scaling operations are performed for all targets in the images. For instance, for Dataset2, if the relative scale of an object is larger than the average scale s1, the target object is keyed out according to the label box, followed by scaling the object according to the scale scaling factor a_{21}, and then the object is put back to the original position to keep the center position unchanged. Additionally, to not damage the contextual structure information of the target object, this study uses the adjacent pixel-based image interpolation method to recover the empty part caused by scaling the target object, and the same processing is conducted for Dataset1. Figure 6 shows the image comparison effect of the objective-based scale matching method.

2.2.4 Loss function

The loss function used to detect tea shoots in this paper consists of three components: confidence loss function, classification loss function, and boundary regression prediction loss function, as shown in Equation (16).

$$L_O = L_{Confidence} + L_{Classification} + L_{Box} \tag{16}$$

3 Results and discussion

3.1 Experimental details

The experiment was conducted on a computer running Windows 10 operating system, and the hardware and software parameters are listed in Table 3. The official YOLOv5 version 6.1 (Jocher et al., 2022) codebase was taken, and the modifications described in sections 2.2.1 and 2.2.2 were implemented on top of it. The training was performed using the SGD optimizer. The initial learning rate was 1E-2, the final learning rate was 1E-5, and the weights decayed to 5E-3. After a momentum of 0.8 was used in the first three warm-up phases, it became 0.937. The training process was run for 300 epochs with a batch size of 4. Online data enhancement methods such as horizontal flip, random rotation, color change, and mosaic, were used during the training to enhance the sample diversity.

3.2 Evaluation metrics

In this study, floating point operations per second (GFLOPs), precision (Precision), recall (Recall), and average precision (mAP)

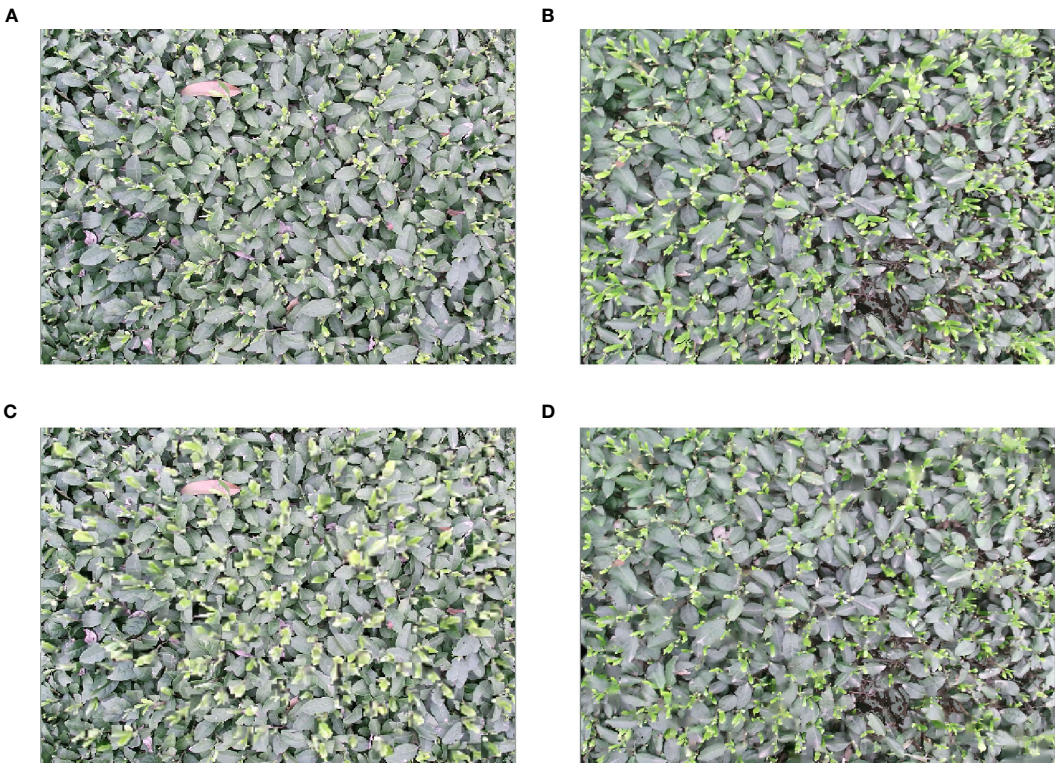


FIGURE 6
Objective-based scale matching method. **(A)** Example image in Dataset1; **(B)** Example image in Dataset2; **(C)** Example image after Dataset1 is aligned to Dataset2 scale; **(D)** Example image after Dataset2 is aligned to Dataset1 scale.

TABLE 3 Software and hardware parameters.

Accessories	Model
Operating system	Windows 10
CPU	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz
RAM	128 GB
GPU	NVIDIA Quadro RTX 5000
Development environments	Python3.8, Pytorch1.10.1, CUDA10.2

were taken as evaluation metrics for measuring model complexity and performance. The calculation formulas of these metrics are shown in Equations (17-21).

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

$$Recall = \frac{TP}{TP + FN} \tag{18}$$

$$mAP = \frac{\sum_1^N AP}{N} = \sum_1^N \int_0^1 P\frac{(R)dR}{N} \tag{19}$$

$$Parameter = C_{in} \times C_{out} \times K \times K \tag{20}$$

$$GFLOPs = (2C_{in}K^2 - 1) \times H_{out}W_{out}C_{out} \tag{21}$$

The parameter denotes the number of parameters of the model. GFLOPs is a metric of the computational power of the model, and a smaller GFLOPs value indicates that the model has less computational burden and can respond to requests faster. The two metrics visually represent the complexity of the model. TP, FP, and FN denote the number of correctly detected objects, incorrectly detected objects, and undetected tea shoot objects, respectively. Precision is the probability that a tea shoot is predicted to be a positive sample among the actual positive samples. The recall is the probability of tea shoots being predicted as positive among the actual positive samples. AP represents the average precision, a combination of precision and recall. The mAP is the average of AP of different categories, where N is the number of types; in this experiment, there is only one category of tea shoots, so N is 1. In this study, mAP50 and mAP95 refer to the mAP values when the value of IOU is taken at 50% and 95%, respectively.

3.3 Ablation and comparison experiments

This section validates the models and methods selected and designed in this study through ablation experiments and comparison experiments. First, a set of comparison experiments was designed to verify the validity of the baseline model selected in this study. Then, a group of ablation experiments based on the modified baseline model was carried out to demonstrate the effectiveness of the improved method adopted in this study. Next, the superiority of the proposed method was verified by designing a set of comparative experiments of multimodal image target detection using different fusion methods and approaches. Finally, a set of ablation experiments was designed to verify the effectiveness of the migration learning and scale -matching methods.

3.3.1 Validation of the baseline framework

In this experiment set, 200 color RGB tea shoot images in Datatset3 were used as the experimental dataset, and it was divided into a training set, a validation set, and a test set at the ratio of 8:1:1. The dataset was trained and validated on models of YOLOv3 (Redmon and Farhadi, 2018), YOLOv4 (Bochkovskiy et al., 2020), YOLOv5, YOLOv6 (Li C. et al., 2022), YOLOv7 (Wang et al., 2022), and YOLOv8 (Jocher et al., 2023), and the test results and model performance are shown in Table 4. To ensure fairness, no pre-training weights were used for all models in the training process, and the testing environment and configuration were identical during the experiments.

Although YOLOv8 obtained the highest mAP50 value, its number of parameters was 1.5 times larger than that of the YOLOv5s model, and its GFLOPs was 1.8 times higher than that of the YOLOv5s model. YOLOv3, YOLOv4, and YOLOv7, although their number of parameters and GFLOPs were smaller, had relatively low mAP50 values, and especially, YOLOv3 and YOLOv4 had a lower recall. YOLOv6 performed relatively poorly on small targets with dense tea shoots. Overall, YOLOV5 is much smaller and more lightweight than the other models in terms of parameter size and GFLOPS, although its mAP50 value is lower than the highest value. Therefore, YOLOv5s is easier to

TABLE 4 Comparative results of detection capabilities of different YOLO frameworks and baseline models.

Method	Parameters	GFLOPs	P	R	mAP50	mAP95
YOLOv3	8666692	12.9	0.686	0.585	0.647	0.254
YOLOv4	5874116	20.0	0.636	0.608	0.799	0.377
YOLOv5	7012822	15.8	0.825	0.733	0.801	0.425
YOLOv6	18500000	45.17	0.779	0.715	0.623	0.322
YOLOv7	6007596	13.0	0.814	0.733	0.703	0.326
YOLOv8	11125971	28.4	0.819	0.732	0.802	0.459

deploy in practical application scenarios. The above results validate the selection of YOLOv5s as the baseline model in this study.

3.3.2 Validation of baseline model improvements

In this set of experiments, 200 color RGB tea shoot images in Datatset3 were used as the experimental dataset, and they were divided into a training set, a validation set, and a test set at the ratio of 8:1:1. "From Focus to conv" (NoFocus)," From 3693 to 8833" (BH)," From CSP2 to C3_DSConv" (C3_DSConv), and "From PANet to FPN" (FPN) modular architectures and methods were added to the baseline model, respectively. Table 5 presents the experimental results. Note that no pre-training weights were used for all models during training, and the testing environment and configuration were identical during the experiments.

Overall, the mAP50 of the model was improved after the modules and methods described in Section 2.2.1 were added to the baseline model. Particularly, the recall of tea shoots was significantly enhanced when all the improved methods were used, indicating that our proposed method benefits the detection of tea shoots that are prone to miss-detection. Meanwhile, the number of model parameters and GFLOPs was optimized, which is consistent with our original intention to achieve real-time detection of dense and tiny tea shoots through a lightweight model. Note that the accuracy was significantly improved when the BH strategy was used (aggravating the computation of the early stages of the network). Still, the GFLOPs were also increased by introducing more computation. For this purpose, this study used C3_DSConv to reduce the computational effort, and it can be seen that the GFLOPs were significantly reduced without affecting the accuracy.

Additionally, this study demonstrates the performance of the YOLOv5s model under other BH strategies. The details are presented in Table 6. First, it can be seen that relative to the distribution of CSP1_n modules of the original YOLOv5s model, the model detection accuracy and especially the recall were significantly improved by using the method of early calculation of the weighted network. Second, the optimal performance was achieved when the number of CSP1_n modules in the four stages of the backbone was set to 8, 8, 3, and 3, respectively.

3.3.3 Comparison of multimodal image fusion methods

In this set of experiments, the Dataset3 dataset was used as the experimental dataset, and it was divided into a training set, a validation set, and a test set at the ratio of 8:1:1. However, it is worth noting that the data were preprocessed differently according to different modal fusion methods. This is shown in detail in Section 2.2.2. Also, to further validate the effectiveness and superiority of our proposed baseline model and the multimodal feature fusion model, different experimental models were compared. The performance of the data layer fusion approach was compared on the YOLOv5s baseline and improved models. The performance of the feature layer fusion approach was compared on the CFT model proposed by (Qingyun et al., 2021), the HINet proposed by (Park, 2022), and the YOLOv5-Multimodal model designed in this study. Besides, to show the impact of the baseline improvement-based approach and the introduction of the FFA model, Without_FFA and Without_Improve were added as the ablation experiments for the YOLOv5-Multimodal model. No pre-training weights were used for all models in the training process, and the test environments and configurations were identical during the experiments. Table 7 presents the specific comparison results.

Overall, the detection accuracy of tea shoots was improved after the multimodal fusion method was used, indicating that the information in different modalities is complementary, and our conjecture in Section 2.2.2 is validated. Regarding the various fusion methods, the multimodal image fusion method using channel-based (Data_Fusion1) achieves a more considerable accuracy gain than the multimodal image fusion method using pixel-by-pixel (Data_Fusion2). However, it increases the number of parameters by a smaller amount. Meanwhile, the multimodal image fusion method with a feature layer introduces more parameters than the multimodal image fusion method based on the data layer. Notably, the mAP50 value of the model decreased when HINet was used directly. Since the HINet model extracts high-frequency information in the frequency domain, so it loses more low-frequency information to guide the detection of small targets. Also, the information is not filtered and aligned in the cross-modal fusion process, thereby introducing some noise that affects the training and convergence of the model. For the GPT model, although the detection accuracy was improved, the use of the multi-head self-attentive mechanism (MHSA) (Vaswani et al., 2017) in the cross-modal fusion module introduces a large number of parameters and computational effort, which is not acceptable in a low-cost agricultural application environment.

In contrast, the model YOLOv5s_Multimodal proposed in this study significantly reduced the number of parameters by purifying, fusing, and enhancing multimodal information in the frequency

TABLE 5 Results of ablation experiments with improved baseline model.

NoFocus	BH	FPN	C3_DSConv	Parameters	GFLOPs	P	R	mAP50	mAP95
				7012822	15.8	0.825	0.733	0.801	0.425
√				7012822	15.8	0.835	0.742	0.808	0.427
	√			6746326	16.4	0.841	0.750	0.814	0.444
		√		5979478	14.6	0.836	0.750	0.814	0.441
			√	7016278	13.7	0.826	0.749	0.805	0.426
√	√	√	√	**5715670**	**13.5**	**0.841**	**0.751**	**0.818**	**0.448**

Bold indicates the best experimental results.
The "√" symbol indicates the use of the policy, method, or module.

TABLE 6 Performance demonstration of the YOLOv5s model under other BH strategies.

Number of CSP1_n modules	Parameters	P	R	mAP50	mAP95
3,6,9,3	7012822	0.825	0.733	0.801	0.425
5,7,6,1	6859030	0.842	0.753	0.811	0.435
4,8,6,2	6889814	0.838	0.740	0.803	0.427
7,8,3,1	6736022	0.844	0.745	0.813	0.442
6,7,4,2	6694934	0.836	0.747	0.813	0.436
10,6,4,1	6705238	0.843	0.746	0.810	0.439
9,8,2,2	6746326	0.839	0.746	0.813	0.443
8,7,3,3	6705238	0.840	0.752	0.811	0.437
9,6,3,3	6705238	0.837	0.747	0.807	0.437
9,9,3,1	6746326	0.839	0.753	0.811	0.441
8,9,2,2	6746326	0.836	0.748	0.811	0.440
8,8,3,3	**6746326**	**0.841**	**0.750**	**0.814**	**0.444**

Bold indicates the best experimental results.

domain and obtained the best mAP50 value for the tea shoot detection. Meanwhile, by comparing the use of YOLOv5s and YOLOv5s_improve models in different fusion methods, it was found that both YOLOv5s_improve models performed optimally, which again demonstrated the superiority and robustness of the dense and tiny tea shoot detector designed in this study. Note that when the Without_FFA model was used, i.e., directly summing and fusing the features under two modes, the mAP50 value reached the lowest value, which was even lower than that of the unimodal target based on the YOLOv5s model. To analyze this result, the feature maps and 3D surface maps of the first fusion stage of the Without_FFA model and YOLOv5s_Multimodal model are shown in Figure 7.

Figures 7C, F reveals that when the features extracted in different modalities are directly summed and fused, the resulting feature maps are relatively noisy, and the target edges will be more obvious for the pairs. This is because the coarse and cluttered feature information deteriorates the training and convergence of the model. However, when the FFA module was used to calibrate, purify, and enhance the feature information within and between each modality, the noise in the feature maps was significantly reduced. The tea shoot targets were more prominent, and the edges were more clearly defined. It can be seen from Figures 7B, E that in the 3D image with preserved spatial information, the tea shoots do not show significant gradient differences from the background compared to the direct summation mode of the multimodal feature information. However, after the FFA module was used again, the tea shoots exhibited noticeable gradient differences with the background leaves, which is beneficial for identifying and localizing tea shoots. Also, this demonstrates the effectiveness and superiority of our proposed FFA module on the multimodal tea shoot dataset.

3.3.4 Verification of scale matching

To investigate and validate the effectiveness of the scale-matching-based transfer learning method in tea shoot detection, a

TABLE 7 Comparison of experimental results of different fusion methods and different models.

Fusion Method	Model	Parameters	P	R	mAP50	mAP95
Data_Fusion1	YOLOv5s_3ch	7012822	0.833	0.740	0.804	0.446
	YOLOv5s_improve_3ch	5715670	0.848	0.754	0.820	0.446
Data_Fusion2	YOLOv5s_4ch	7013974	0.832	0.742	0.808	0.436
	YOLOv5s_improve_4ch	5722230	0.848	0.756	0.824	0.460
Feature_Fusion	GPT(s)	44500982	0.840	0.745	0.810	0.426
	HINet(s)	23738982	0.821	0.731	0.794	0.413
	Without_FFA	11261174	0.807	0.718	0.774	0.394
	Without_Improve	26424892	0.834	0.742	0.809	0.429
	YOLOv5s_Multimodal	**24764092**	**0.850**	**0.759**	**0.827**	**0.447**

Bold indicates the best experimental results.

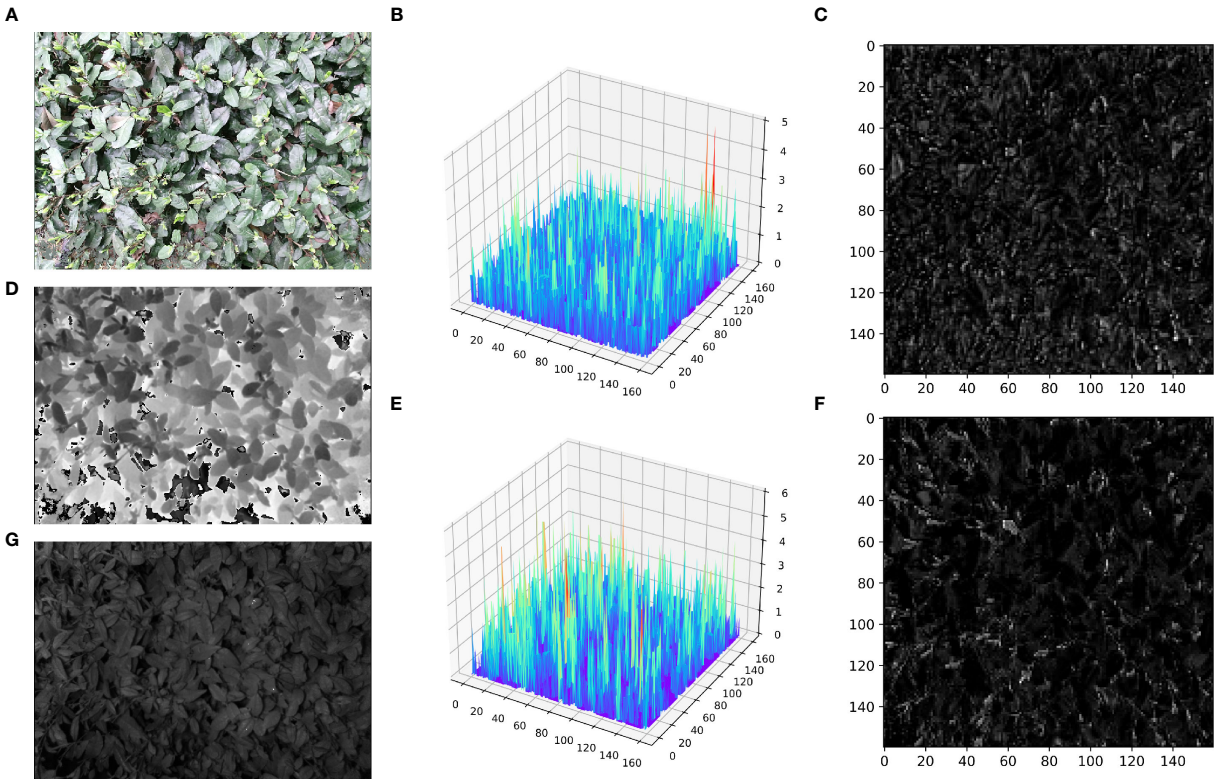


FIGURE 7
First fusion stage feature map visualization. **(A)** input RGB image; **(D)** input Depth image; **(G)** input IR image; **(C)** feature map of the first fusion stage in the Without_FFA model; **(B)** 3D surface map corresponding to the feature map of the first fusion stage in the Without_FFA model; **(F)** feature map of the first fusion stage in the YOLOv5s_Multimodal model phase in the YOLOv5s_Multimodal model; **(E)** 3D surface map corresponding to the feature map of the first fusion phase in the YOLOv5s_Multimodal model.

set of ablation comparison experiments was designed in this study. In the experiments, the color RGB image datasets in Dataset1 and Dataset2 were used as the experimental datasets, called Tea1 and Tea2, respectively, and they were divided into a training set, a validation set, and a test set at a ratio of 8:1:1, and YOLOv5s and YOLOv5s_improve were used as the experimental models. Firstly, this study compared the performance of the two models on Tea1 and Tea2. Secondly, Tea1 was aligned to the scale of Tea2 according to the scale matching method to obtain Tea1up, and the performance of the two models on Tea1up was compared. Similarly, Tea2 was aligned to the scale of Tea1 according to the scale-matching method to obtain Tea2d, and the performance of the two models on Tea2d was compared. Finally, the best weights obtained by training Tea2 and Tea2d were used as pre-training weights to train the model on Tea1 (denoted as Tea2_Tea1 and Tea2d_Tea1, respectively), and the best weights obtained by training on Tea1 and Tea1up were used as pre-training weights to train the model on Tea2 (denoted as Tea1_Tea2 and Tea1up_Tea2). The specific comparison results are given in Figure 8. Note that the test environment and configuration during the experiments are identical.

Figure 8 shows that the precision, recall, and mAP50 values of the Tea1 and Tea2 datasets were reduced when their scales were aligned to that of the original dataset. This may be because the difficulty of small object detection was exacerbated by the reduced scale of Tea2. Besides, since Tea1 ignored the small object objects in the image edges when increasing the scale, it resulted in fewer small target samples, thus affecting the training and convergence of the model. However, the model accuracy improvement could be stronger when Tea1 and Tea2 were used to guide each other's learning, and the scale mismatch problem may arise. When the scale-aligned datasets Tea2d and Tea1up were used to guide the model to learn on the Tea1 and Tea2 datasets, respectively, the detection accuracy was significantly improved. Additionally, to more clearly compare the performance of different scale datasets and pre-training strategies during model training and validation, the localization loss curve of the YOLOv5s_improve model on the validation set is shown in Figure 9.

Figures 9A, B show that when the pre-training weights were used, the initial values of the localization loss were significantly lower, with relatively small curve oscillations, and the loss converged relatively quickly. However, the localization loss converged best when the corresponding scale was used as the pre-training dataset. This also demonstrates the effectiveness of the target-based scale-matching method used in this study in guiding the small target detection task.

3.4 Heat map visualization

To more intuitively illustrate the impact of model improvements, explicitly modifying the baseline model for dense

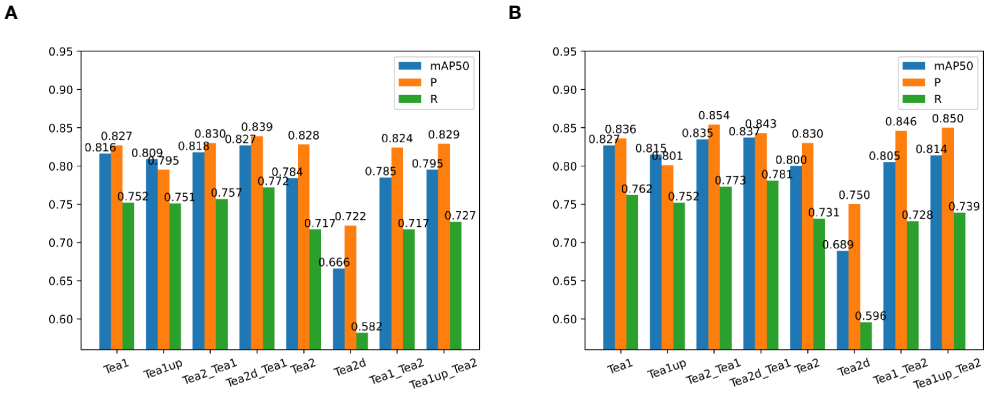


FIGURE 8
Comparison of experimental results of different models using objective-based scale matching and migration learning. **(A)** indicates the performance on the YOLOv5s model using different scale datasets and training strategies; **(B)** shows the performance on the YOLOv5s_improve model using different scale datasets and training strategies.

and small targets, and the effectiveness of multimodal feature fusion methods, this study used a gradient-weighted class activation mapping (Grad-CAM) (Selvaraju et al., 2016) to visualize the model considering the target based on tea shoots. Grad-CAM can exploit the gradient of any target concept to flow into the final convolution layer, thereby generating a rough localization map and displaying it in the form of weights, where the weight values are shown in red, yellow, green, and blue colors in decreasing order. The redder the color in the corresponding graph, the more critical the region for tea shoot detection. Figure 10 shows the heat map visualization results for different models under different inspection conditions.

Both YOLOv5s_improve and YOLOv5s-Multimodal models perform better than YOLOv5s in various cases, e.g., the color of tea shoots is similar to the background, tea shoots are relatively sparse, the target scale is rather large, tea shoots are dense and tiny, the color of tea shoots differs from the background, and the target scale is relatively large. Note that when the tea shoot has a similar color to the leaf and its background is difficult, the YOLOv5s model collects minimal information and does not focus on many tiny tea shoot objects. However, YOLOv5s_improve focuses on more tiny tea shoot objects by enhancing the retention and extraction of detailed texture features. However, it is difficult for YOLOv5s and YOLOv5_improve to focus on the groups of tea shoots with high overlap, especially the tiny tea shoots in the overlap case where the tea shoots are relatively dense and overlapping occlusion occurs. However, the multimodal model YOLOv5s-Multimodal has multi-class information input, so it can find more tea shoots and has better segmentation ability for tea shoot groups with high overlap. Besides, it is no longer limited to the part of the stem tip. The model also considers the related connecting stems, leaves, and stems. This demonstrates the superiority of YOLOv5s-Multimodal for tea shoot detection.

3.5 Visualization of results

To more intuitively compare the performance of different detection models and different fusion methods on the tea shoot detection task in a natural environment, this study performed a

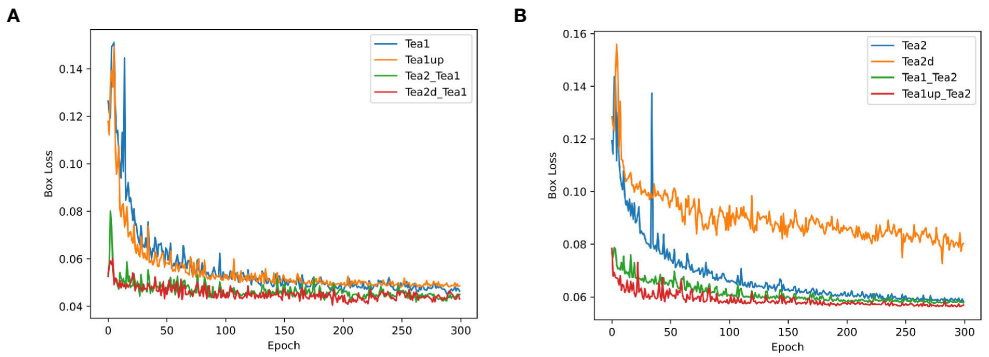


FIGURE 9
Plots of box loss curves on the YOLOv5s_improve model for different scale datasets and pre-training strategies. **(A)** Box loss profile plots of Tea1 at different scales and pre-training strategies; **(B)** Box loss profile plots of Tea2 at different scales and pre-training strategies.

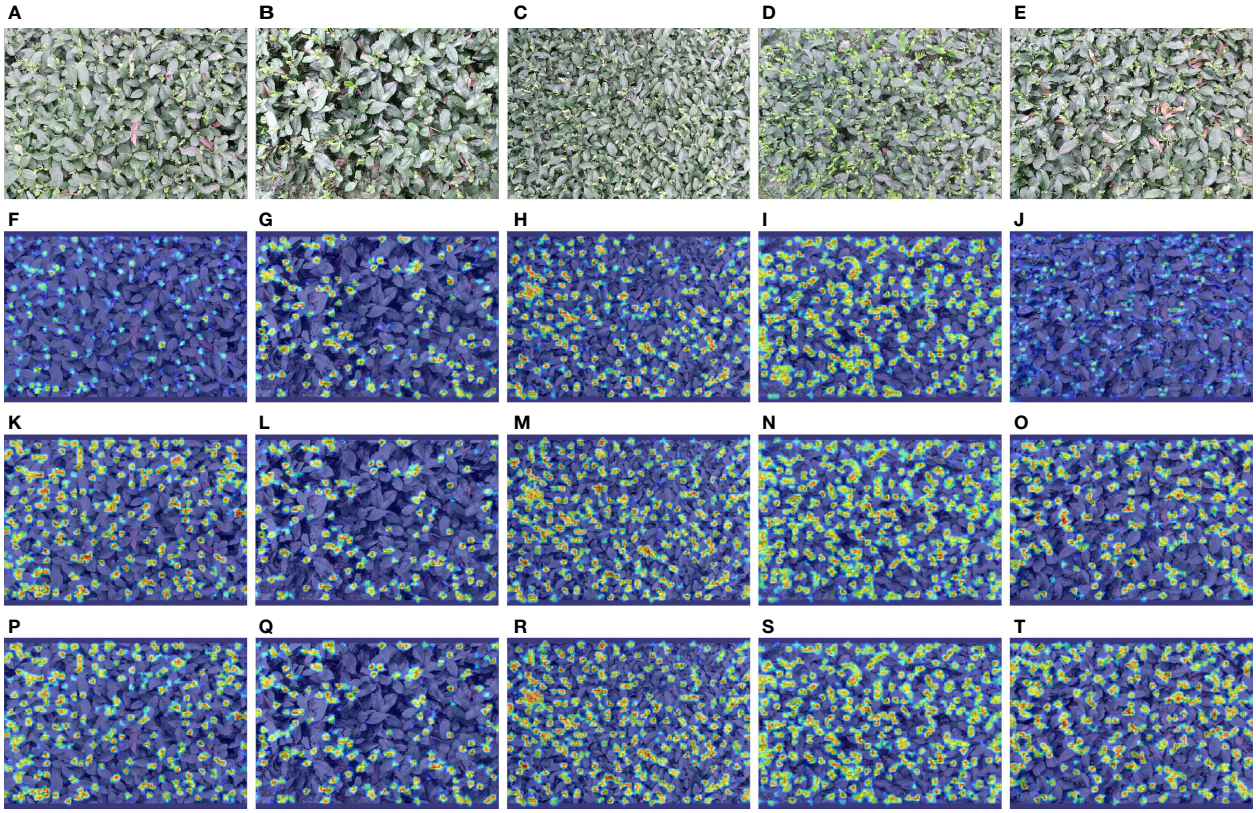


FIGURE 10
Heat map visualization results for different models with different detection conditions. **(A–E)** The input images; **(F–J)** The results of YOLOv5s; **(K–O)** The results of YOLOv5s_improve; **(P–T)** The results of YOLOv5s-Multimodal.

comparative analysis of the visualization results of different types of samples after recognition. In this study, YOLOv5s (single modal), YOLOv5s_improve (single modal), YOLOv5s_improve_3ch (multimodal), YOLOv5s_improve_4ch (multimodal), and YOLOv5s-Multimodal (multimodal) were used on the test set of the corresponding experimental dataset. The inference was conducted, and the performance of these models under different detection conditions is shown in Figure 11.

In Figure 11, the first column shows relatively sparse and tiny tea shoot targets. The second column shows rather large and sparse tea shoot objects. The third column shows relatively dense and small tea-shoot objects. The fourth column shows relatively large and thick tea shoot objects, and the fifth column shows rather complex tea shoot backgrounds. Overall, under different challenging conditions, YOLOv5s_imporve and multimodal-based fusion methods can substantially reduce false negatives (FN), and there is a significant increase in true positives (TP) of YOLOv5s-Multimodal visualization results, which again demonstrates the superiority and robustness of our proposed method.

4 Conclusion

This study aims to improve the detection accuracy of dense and tiny tea shoots in a natural environment and realize real-time object detection. In this paper, a real-time dense and small tea shoot target detection algorithm is designed based on multimodal image data, baseline detection model architecture, multimodal image fusion method, scale matching, and migration learning techniques.

First, to make up for dense and tiny tea shoot detection in a complex environment, this paper uses the Conv layer to replace the Focus layer in the YOLOv5s baseline, which is easy to lose detailed information. This helps to extract features for tea shoot detection by enhancing the computation of the early stage of the network while using DSConv to balance the introduced computation and improve the model's attention to detail texture, and the recall of targets at different scales is enhanced by the FPN structure. The improved model achieves an accuracy of 84.1%, a recall of 75.1%, and a mAP50 value of 81.8% on low-resolution RGB tea shoot images, showing an improvement of 1.6%, 1.8%, and 1.7% compared to the original YOLOv5s model.

Second, to make up for the deficiency of RGB image-based tea shoot detection, two data layer-based multimodal fusion method and one feature layer-based multimodal fusion method are investigated in this paper. Compared with the images based on a single modality, the mAP50 values of Data_Fusion1 and Data_Fusion2 are improved by 1.9% and 2.3%, respectively. Besides, the Feature_Fusion method proposed in this paper achieves the highest mAP50 value of 82.7% at a relatively small number of parameters compared to other feature layer-based

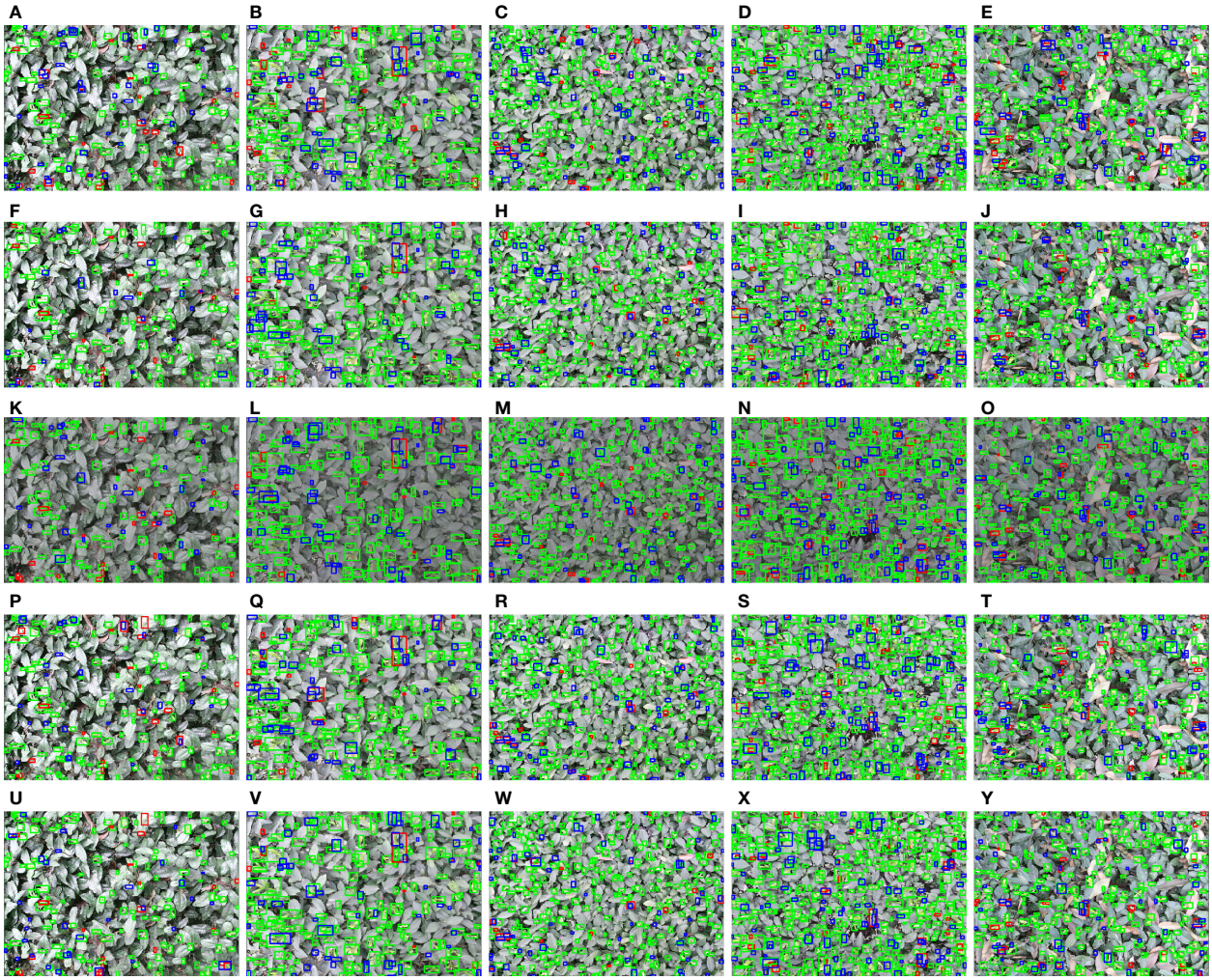


FIGURE 11
Visualization results of different detection models and methods under different detection conditions. **(A–E)** The test results of YOLOv5s; **(F–J)** The test results of YOLOv5s_improve; **(K–O)** The test results of YOLOv5s_improve_3ch; **(P–T)** The test results of YOLOv5s_improve_4ch; **(U–Y)** The test results of YOLOv5s-Multimodal. The green, blue, and red boxes indicate true positive (TP), false positive (FP), and false negative (FN) predictions, respectively.

multimodal fusion methods. This study mainly introduces a frequency domain-based cross-modal attention fusion module to perform purify, align, fuse, and enhance multimodal information with minor computational effort and parameters. Thus, more complementary information beneficial to detecting dense and tiny tea shoots in complex environments is obtained. Although the feature layer-based multimodal fusion approach proposed in this study introduces a larger number of parameters compared with the data layer-based multimodal fusion approach, the former achieves optimal performance, providing a reference for feature layer-based multimodal fusion approaches. In the future, we will continue to consider the feature layer-based multimodal fusion approach in model lightweight.

Finally, to investigate the differences and effects of training at different scales, this study designed comparison experiments on two tea shoot datasets with target scale differences, and their detection results in different periods were compared. It can be found that small-scale target detection is very complex. To improve the accuracy and recall of tea shoot detection in various scales, this study uses migration learning techniques and scale matching to align datasets of different scales and mutually guide the models to learn at the corresponding scales, thereby improving the performance of small target detection.

However, there are still some drawbacks and limitations in this study. First, although the tea shoot samples used for training in this study are about 50,000, the model's generalization still needs to be enhanced because the image data are relatively small and do not contain all natural scenes. Secondly, affected by the data acquisition equipment, there are some voids and noises in the acquired depth maps and infrared images, and in the future, we will consider using techniques such as depth estimation, depth enhancement, and image denoising to obtain high-quality depth images and infrared images. Finally, also affected by the data acquisition equipment, the Kinectv2 device could initially acquire high-resolution RGB images; however, since the color camera has a different field of view from the depth camera, the acquired high-resolution images are not

aligned with the depth images and infrared images, and the existing alignment techniques based on traditional image processing have some errors. This cannot be neglected in the detection task of dense and small tea shoots. In the future, we will consider introducing a deep learning-based image alignment method and combining it with super-resolution techniques to further improve the detection performance of dense and tiny tea shoots.

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

LS conceptualized this study, conducted experiments, wrote the original draft, and revised the manuscript. LS and ZL wrote the manuscript and performed the experiments. LS, ZC, and ZL made the experimental plan, supervised the work, and revised the manuscript. BZ and HL performed the data analysis and revised the manuscript. LS and ZC made the experimental plan and revised the manuscript. JM and YW evaluated the developed technique and revised the manuscript. LS designed the experimental plan, supervised the work, and revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding

This work was funded by the Research and application of key technologies for intelligent spraying based on machine vision (key technology research project) of Sichuan Provincial Department of Science and Technology (grant number 22ZDYF0095).

Acknowledgments

Thanks to all the partners of AI Studio for their support.

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

Jsbroks/coco-annotator: :pencil2: web-based image segmentation tool for object detection, localization, and keypoints. Available at: https://github.com/jsbroks/coco-annotator (Accessed May 14, 2023).

Kinect/PyKinect2: wrapper to expose kinect for Windows v2 API in Python. Available at: https://github.com/Kinect/PyKinect2 (Accessed May 14, 2023).

Releases · ultralytics/yolov5. Available at: https://github.com/ultralytics/yolov5/releases (Accessed May 15, 2023).

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

Bojie, Z., Dong, W., Weizhong, S., Yu, L., and Ke, W. (2019). "Research on tea bud identification technology based on HSI/HSV color transformation," in *2019 6th International Conference on Information Science and Control Engineering (ICISCE)*. (IEEE), 511–515. doi: 10.1109/ICISCE48695.2019.00108

Cao, M., Fu, H., Zhu, J., Cai, C., Cao, M., Fu, H., et al. (2022). Lightweight tea bud recognition network integrating GhostNet and YOLOv5. *Math. Biosci. Eng.* 19 (12), 12897–12914. doi: 10.3934/MBE.2022602. (n.d.).

Fu, L., Majeed, Y., Zhang, X., Karkee, M., and Zhang, Q. (2020). Faster r–CNN–based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting. *Biosyst. Eng.* 197, 245–256. doi: 10.1016/J.BIOSYSTEMSENG.2020.07.007

Gan, H., Lee, W. S., Alchanatis, V., Ehsani, R., and Schueller, J. K. (2018). Immature green citrus fruit detection using color and thermal images. *Comput. Electron Agric.* 152, 117–125. doi: 10.1016/J.COMPAG.2018.07.011

Gené-Mola, J., Vilaplana, V., Rosell-Polo, J. R., Morros, J. R., Ruiz-Hidalgo, J., and Gregorio, E. (2019). Multi-modal deep learning for Fuji apple detection using RGB-d cameras and their radiometric capabilities. *Comput. Electron Agric.* 162, 689–698. doi: 10.1016/J.COMPAG.2019.05.016

Han, Y., Xiao, H., Qin, G., Song, Z., Ding, W., and Mei, S. (2014). Developing situations of tea plucking machine. *Engineering* 06, 268–273. doi: 10.4236/ENG.2014.66031

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 (9), 346–361. doi: 10.1007/978-3-319-10578-9_23

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

Hong, D., Gao, L., Yao, J., Zhang, B., Plaza, A., and Chanussot, J. (2021). Graph convolutional networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 59, 5966–5978. doi: 10.1109/TGRS.2020.3015157

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Fang, J., et al. (2022). *ultralytics/yolov5: v6. 1-TensorRT, TensorFlow edge TPU and OpenVINO export and inference*. *Zenodo* doi: 10.5281/zenodo.1234

Jocher, G., Chaurasia, A., and Qiu, J. (2023). *YOLO by Ultralytics (Version 8.0.0)*. [Computer software]. https://github.com/ultralytics/ultralytics. doi: 10.5281/zenodo.1234

Karunasena, G. M. K. B., and Priyankara, H. D. N. S. (2020). Tea bud leaf identification by using machine learning and image processing techniques. *Int. J. Sci. Eng. Res.* 11, 624–628. doi: 10.14299/IJSER.2020.08.02

Li, W., Chen, R., and Gao, Y.y. (2021). "Automatic recognition of tea bud image based on support vector machine," in *Advanced Hybrid Information Processing: 4th EAI International Conference, ADHIP 2020*, Binzhou, China. (Springer International Publishing) Vol. 348, 279–290. doi: 10.1007/978-3-030-67874-6_26

Li, Y., He, L., Jia, J., Chen, J., Lyu, J., and Wu, C. (2022). High-efficiency tea shoot detection method *via* a compressed deep learning model. *Int. J. Agric. Biol. Eng.* 15, 159–166. doi: 10.25165/J.IJABE.20221503.6896

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., et al. (2022) *YOLOv6: a single-stage object detection framework for industrial applications*. Available at: https://arxiv.org/abs/2209.02976v1 (Accessed May 15, 2023).

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2016) *Feature pyramid networks for object detection*. Available at: https://arxiv.org/abs/1612.03144v2 (Accessed May 14, 2023).

Lin, Z., Fu, R., Ren, G., Zhong, R., Ying, Y., and Lin, T. (2022). Automatic monitoring of lettuce fresh weight by multi-modal fusion based deep learning. *Front. Plant Sci.* 13. doi: 10.3389/FPLS.2022.980581

Lin, G., Tang, Y., Zou, X., Li, J., and Xiong, J. (2019). In-field citrus detection and localisation based on RGB-d image analysis. *Biosyst. Eng.* 186, 34–44. doi: 10.1016/J.BIOSYSTEMSENG.2019.06.019

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018) *Path aggregation network for instance segmentation.* Available at: https://arxiv.org/abs/1803.01534v4 (Accessed May 14, 2023).

Liu, Z., Wu, J., Fu, L., Majeed, Y., Feng, Y., Li, R., et al. (2019). Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion. *IEEE Access* 8, 2327–2336. doi: 10.1109/ACCESS.2019.2962513

Nascimento, M. G., Do Prisacariu, V., and Fawcett, R. (2019). "DSConv: efficient convolution operator," in *Proceedings of the IEEE International Conference on Computer Vision*. 5147–5156. doi: 10.1109/ICCV.2019.00525

Ning, J., Guan, H., and Spratling, M. (2023). Rethinking the backbone architecture for tiny object detection. *arXiv preprint* arXiv:2303.11267. 103–114. doi: 10.5220/0011643500003417

Park, S.-H. J.-S. S. B. S. (2022). High-frequency interchange network for multispectral object detection. *J. Korea Institute Inf. Commun. Eng.* 26, 1121–1129. doi: 10.6109/JKIICE.2022.26.8.1121

Qingyun, F., Dapeng, H., and Zhaokui, W. (2021). Cross-modality fusion transformer for multispectral object detection. *arXiv preprint* arXiv:2111.00273. doi: 10.48550/arXiv.2111.00273

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

Redmon, J., and Farhadi, A. (2018) *YOLOv3: an incremental improvement*. Available at: https://arxiv.org/abs/1804.02767v1 (Accessed May 15, 2023).

Rong, J., Zhou, H., Zhang, F., Yuan, T., and Wang, P.(2023). Tomato cluster detection and counting using improved YOLOv5 based on RGB-d fusion. *Comput. Electron Agric.* 207, 107741. doi: 10.1016/J.COMPAG.2023.107741

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2016). Grad-CAM: Visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vis.* 128, 336–359. doi: 10.1007/s11263-019-01228-7

Stefanics, D., and Fox, M. (2022). COCO Annotator: Web-Based image segmentation tool for object detection, localization, and keypoints. *ACM SIGMultimedia Records* 13 (3), 1–1. doi: 10.1145/3578495.3578502

Sun, Q., Chai, X., Zeng, Z., Zhou, G., and Sun, T. (2022). Noise-tolerant RGB-d feature fusion network for outdoor fruit detection. *Comput. Electron Agric.* 198, 107034. doi: 10.1016/J.COMPAG.2022.107034

Sun, K., Xiao, B., Liu, D., and Wang, J. (2019). "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 5693–5703. doi: 10.1109/CVPR.2019.00584

Tao, Y., and Zhou, J. (2017). Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking. *Comput. Electron Agric.* 142, 388–396. doi: 10.1016/J.COMPAG.2017.09.019

Tu, S., Xue, Y., Zheng, C., Qi, Y., Wan, H., and Mao, L. (2018). Detection of passion fruits and maturity classification using red-Green-Blue depth images. *Biosyst. Eng.* 175, 156–167. doi: 10.1016/J.BIOSYSTEMSENG.2018.09.004

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*. Available at: https://arxiv.org/abs/1706.03762v5 (Accessed May 14, 2023).

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*. Available at: https://arxiv.org/abs/2207.02696v1 (Accessed May 15, 2023).

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 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 390–391. doi: 10.1109/CVPRW50498.2020.00203

Wu, X., Hong, D., and Chanussot, J. (2021). Convolutional neural networks for multimodal remote sensing data classification. *IEEE Trans. Geosci. Remote Sens.* 60, 1–10. doi: 10.1109/TGRS.2021.3124913

Wu, X., Hong, D., and Chanussot, J. (2023). UIU-Net: U-Net in U-Net for infrared small object detection. *IEEE Trans. Image Process.* 32, 364–376. doi: 10.1109/TIP.2022.3228497

Wu, X., Hong, D., Tian, J., Chanussot, J., Li, W., and Tao, R. (2019). ORSIm detector: a novel object detection framework in optical remote sensing imagery using spatial-frequency channel features. *IEEE Trans. Geosci. Remote Sens.* 57, 5146–5158. doi: 10.1109/TGRS.2019.2897139

Xiaoxiao, S., Shaomin, M., Yongyu, X., Zhihao, C., and Tingting, S. (2019). Detection algorithm of tea tender buds under complex background based on deep learning. *J. Hebei University (Natural Sci. Edition)* 39, 211. doi: 10.3969/J.ISSN.1000-1565.2019.02.015

Xu, W., Zhao, L., Li, J., Shang, S., Ding, X., and Wang, T. (2022). Detection and classification of tea buds based on deep learning. *Comput. Electron Agric.* 192, 106547. doi: 10.1016/J.COMPAG.2021.106547

Yu, X., Gong, Y., Jiang, N., Ye, Q., and Han, Z. (2020). "Scale match for tiny person detection," in *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*. 1257–1265. doi: 10.1109/WACV45572.2020.9093394

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 24 July 2023
DOI 10.3389/fpls.2023.1226329

Check for updates

OPEN ACCESS

EDITED BY
Changcai Yang,
Fujian Agriculture and Forestry University, China

REVIEWED BY
Chi Chen,
Wuhan University, China
Taogang Hou,
Beijing Jiaotong University, China

*CORRESPONDENCE
Yong Li
yongli@gxu.edu.cn

RECEIVED 21 May 2023
ACCEPTED 26 June 2023
PUBLISHED 24 July 2023

CITATION
Guo Z, Goh HH, Li X, Zhang M and Li Y (2023) WeedNet-R: a sugar beet field weed detection algorithm based on enhanced RetinaNet and context semantic fusion. *Front. Plant Sci.* 14:1226329. doi: 10.3389/fpls.2023.1226329

COPYRIGHT
© 2023 Guo, Goh, Li, Zhang 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.

WeedNet-R: a sugar beet field weed detection algorithm based on enhanced RetinaNet and context semantic fusion

Zhiqiang Guo[1], Hui Hwang Goh[1], Xiuhua Li[1,2], Muqing Zhang[2] and Yong Li[1]*

[1]School of Electrical Engineering, Guangxi University, Nanning, China, [2]Guangxi Key Laboratory of Sugarcane Biology, Guangxi University, Nanning, China

Accurate and dependable weed detection technology is a prerequisite for weed control robots to do autonomous weeding. Due to the complexity of the farmland environment and the resemblance between crops and weeds, detecting weeds in the field under natural settings is a difficult task. Existing deep learning-based weed detection approaches often suffer from issues such as monotonous detection scene, lack of picture samples and location information for detected items, low detection accuracy, etc. as compared to conventional weed detection methods. To address these issues, WeedNet-R, a vision-based network for weed identification and localization in sugar beet fields, is proposed. WeedNet-R adds numerous context modules to RetinaNet's neck in order to combine context information from many feature maps and so expand the effective receptive fields of the entire network. During model training, meantime, a learning rate adjustment method combining an untuned exponential warmup schedule and cosine annealing technique is implemented. As a result, the suggested method for weed detection is more accurate without requiring a considerable increase in model parameters. The WeedNet-R was trained and assessed using the OD-SugarBeets dataset, which is enhanced by manually adding the bounding box labels based on the publicly available agricultural dataset, i.e. SugarBeet2016. Compared to the original RetinaNet, the *mAP* of the proposed WeedNet-R increased in the weed detection job in sugar beet fields by 4.65% to 92.30%. WeedNet-R's average precision for weed and sugar beet is 85.70% and 98.89%, respectively. WeedNet-R outperforms other sophisticated object detection algorithms in terms of detection accuracy while matching other single-stage detectors in terms of detection speed.

KEYWORDS

precision farming, deep learning, object detection, weed recognition, sugar beets

1 Introduction

Damage caused by weeds on fields is a significant factor influencing agricultural progress. Weeds in the field compete with crops for sunshine, water, and nutrients, resulting in a deterioration in crop quality and a fall in crop output, which causes substantial losses to the agricultural economy. With the rapid development of agricultural mechanization and information technologies, it is anticipated that automatic weeding robots will be widely applied in weed management, achieving the goals of reducing pesticide use, conserving resources, protecting the ecological environment, and increasing agricultural yields. Vision-based weeding robots for weed management rely heavily on the detection and identification of weeds (Li et al., 2022). Complex farming landscapes with dynamically changing, unstructured, and various conflicting noise characteristics make it challenging for weeding robots to detect and find weeds in the field. In addition, the diversity of weed morphology at various growth phases and the complexity of the soil background in which weeds grow aggravate the difficulties of weed detection. Consequently, weed detection and localization in the field remains a difficult undertaking (Wang, 2019).

In recent years, significant progress has been made in machine-vision-based weed detection approaches. However, the field of weed detection on farmland still faces persistent challenges, including the scarcity of available weed datasets, the presence of monotonous backgrounds, limited availability of diverse learning samples, the inability to achieve end-to-end solutions, and low detection accuracy. These challenges continue to pose obstacles for researchers and practitioners in the field. To address the aforementioned challenges and foster the advancement of deep academic-based target detection technology in the field of weed detection on farmland, we reconstructed an weed dataset of about 5000 images with annotation labels of bounding boxes based on the publicly available agricultural dataset SugarBeets2016 and named it OD-SugarBeets. In the meantime, we present WeedNet-R, an object identification network based on the one-stage framework network RetinaNet, for weed recognition and localisation in sugar beet fields. Inspiring by the work of Najibi et al. (2017) and Deng et al. (2019) on face detection, WeedNet-R incorporates numerous context modules in the neck of RetinaNet to combine feature maps with varying receptive field sizes from distinct layers. Utilizing context modules improves the WeedNet-R's capacity to represent context information, hence enhancing its weed identification precision. Moreover, a learning rate adjustment method combining an untuned exponential warmup schedule and cosine annealing technique for the Adam optimizer is implemented during model training in order to increase the network's ability to seek for its global optimal solution. In addition, we present a crop-first non-maximum suppression strategy to eliminate repeated prediction bounding boxes below a certain confidence level. The object that is anticipated by the network to be both weeds and crop is favored to be crop to minimize the possibility of crop being erroneously removed.

The following is a summary of this article's primary contributions. (1) We propose the WeedNet-R weed detection model for sugar beet fields, which is based on RetinaNet. Multiple context modules are added to WeedNet-R's neck in order to expand the network's receptive field. As a result, the accuracy of weed recognition is enhanced without a major increase in model parameters. (2) An untuned exponential warmup schedule is set for the Adam optimizer during WeedNet-R training, thereby enhancing the network's search potential for global optimal solutions. (3) Nearly 5,000 images from the SugarBeet2016 dataset were manually re-labeled with bounding boxes to address the limitation that the dataset cannot be utilized directly to object detection techniques. The SugarBeet2016 update dataset has been published to a public repository[1] for the development and assessment of other weed algorithms.

The remaining sections are organized as follows. In Section 2, the relevant works on deep learning-based weed detection systems from recent years are briefly discussed. In Section 3, the picture dataset and proposed method for weed detection will be introduced. Section 4 describes the experimental conditions and associated assessment metrics for weed detection. Section 5 contains the entire experimental findings analysis and commentary. Finally, in Section 6 we end our task.

2 Related works

In recent years, the field of weed detection has witnessed a growing interest in deep learning and image recognition-based approaches. Within this context, two main machine vision-based strategies have emerged: individual or pixel-level classification and object detection or instance segmentation.This section provides a comprehensive review of individual or pixel-based classification methods and object detection or instance-based segmentation methods. Subsequently, we provide concise definitions of key concepts related to network enhancements, including context information, focus loss, and warmup schedule.

2.1 Individual or pixel level classification-based methods

Individual level classification-based approaches use the entire image as the model input and differentiate between weeds and crops based on the classification of the image. This approach has been widely employed in weed detection investigations in the past. Olsen et al. (2019) employed Inception-v3 and ResNet-50 as baseline models to test weed classification performance on the DeepWeeds public images dataset (https://github.com/AlexOlsen/DeepWeeds). The average classification accuracy of these models is 95.1% and 95.7%, respectively. Hu et al. (2020). developed graph convolution to characterise RGB images as multi-scale graphs in order to generate deep feature representations at a fine-grained level, and the average classification accuracy on DeepWeeds was 98.1%. Espejo-Garcia et al. (2020) integrated convolutional neural networks with standard machine learning classifiers in order to

1 Available at https://github.com/GOOJJJ/WeedNet-R/

capitalise on the powerful feature extraction capabilities of convolutional neural networks and the high classification performance of machine learning classifiers. Consequently, the DenseNet-SVM model earned an F1 score of 99.29% on the picture dataset of various Greek farms.

Typically, the pixel level classification-based algorithms categorise each pixel in the detected image into one of three categories: crop, weed, and background, thereby separating weed and crop from the background. Recent investigations have been undertaken on the basis of this concept. (Lottes et al., 2016) suggested an encoding-decoding model based on fully convolutional networks (FCN) to distinguish crop and weed from the background by including spatial information from image sequences. Sa et al. (2017) proposed a pixel-wise segmentation network named 'weedNet' based on SegNet (Badrinarayanan et al., 2017) to classify weeds and crops in UVA's images. And Bosilj et al. (2020) included transfer learning into SegNet for weed recognition in various types of crops to reduce the necessary retraining time and labeling effort for new crop types. Image segment improvement techniques have also attracted the interest of researchers. By combining NIR image information, Wang et al. (2020) increased the resilience of segmentation algorithms against diverse lighting situations. In their work, the best mean intersection over union (mIoU) for pixel-wise segmentation was 88.91%. In addition, Fawakherji et al. (2019). employed a deep network based on the UNet for pixel-wise semantic segmentation, background removal, and ROIs extraction. A CNNs-based classifier was then applied to classify the retrieved ROIs as crop or weed. However, neither individual-level classification nor pixel-level classification can simultaneously classify and locate weeds end-to-end. And they require image additional pre-processing and post-processing techniques to detect the distribution of weeds and crops in the images.

2.2 Object detection or instance segmentation-based methods

Unlike individual or pixel-level classification-based methods, object detection-based methods for weed detection discover all objects of interest using prediction bounding boxes including category information. In recent years, object detection approaches based on deep learning have garnered increasing interest for weed detection and location on agriculture. Jiang et al. (2019) suggested a two-stage network with Inception-ResNet v2 as the backbone based on Faster R-CNN and transfer learning to detect in-row weeds in cotton fields. However, the quantity of weeds in the image datasets used was very limited, making detection easier. Gao et al. (2020) suggested a data augmentation approach for training samples and combined synthetic and original field images to train the YOLOv3-based model, which produced a *mAP* of 0.829%. However, the dataset utilized was quite limited and contained a monotonous soil background, and the strategy of increasing the original dataset via picture synthesis alone could result in model overfitting. Jin et al. (2021) suggested a system based on deep learning to detect weeds in vegetable fields. A trained CenterNet model was initially used to locate vegetable plants with bounding boxes. Then, image segmentation was utilized to identify weeds outside the vegetable-bounding boxes. It is evident that this detection method is not end-to-end, as the complicated image post-processing will require a significant amount of CPU resources. In addition, the color-index-based picture segmentation method is highly sensitive to illumination and plant colour, therefore the algorithm's capacity for generalization may be limited. Zhuang et al. (2022). assessed the effectiveness of five distinct object identification models for the detection of broadleaf weeds in wheat seedlings. Since none of these models have a recall rate more than 0.58, the researchers concluded that these models are insufficient to detect weeds in wheat without improvement. In a new study, researchers are investigating a method based on instance segmentation for detecting the contours and locations of weeds in images of farmland. For instance, Champ et al. (2020) trained and evaluated a Mask R-CNN model for field weed detection using a data set containing 2489 image samples, achieving a pretty good detection accuracy. In actual weed control, weed eradication efficiency reached up to 60 percent. However, instance segmentation-based methods for weed detection demand more computing resources than object detection-based methods.

Recently, Transformer has shown great success in Natural Language Processing (NLP). It has also been applied to computer vision tasks, yielding excellent results (Dosovitskiy et al. 2021). Transformer-based object detectors, such as Swin-Transformer (Liu et al., 2021), DETR (Zhu et al, 2020), and DINO (Zhang et al., 2022a), have emerged and been applied in weed detection tasks (Zhang et al., 2022b).

2.3 Context information

In a convolutional neural network, the receptive field (RF) represents the capacity of the convolutional unit to sense the size of the input region. Typically, the receptive field size is calculated beginning with the first layer of the input feature map, and different convolutional layers have varying receptive field sizes. As demonstrated in Figure 1, the theoretical receptive field (TRF) of a convolutional neural network increases as the number of convolutional layers increases in depth. The greater the value of RF, the larger the region of the raw input that the output feature map sees, which may imply more global and higher-level semantic characteristics. However, for deep learning models, the effective receptive field (ERF) has a greater impact than the fixed TRF of the networks. In order to expand the ERF of the model, context information is utilized and enhanced by fusing the model's characteristics with RFs of varying sizes from different layers. Najibi et al. (2017) accomplished more efficient contextual modelling by adding additional convolutional filtering layers to each prediction module of the SSH face detection network, hence obtaining a larger ERF. PyramidBox (Tang et al., 2018) has developed a context-aware prediction module that retains rich context information from multiple feature layers. Deng et al. (2019) introduced independent contextual modules to the five feature layers of the FPN of the single-stage face detector to raise

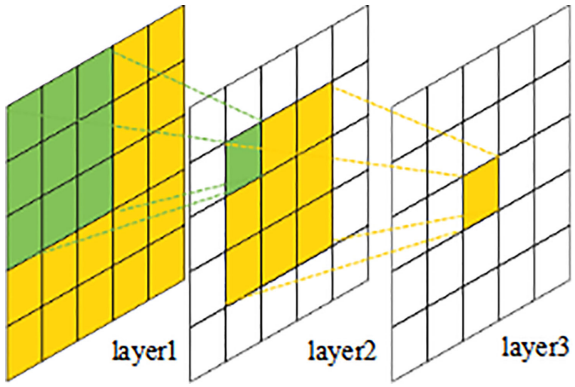


FIGURE 1
Receptive field sizes of different layers.

the ERF of the network, hence enhancing its rigid context semantic modelling capabilities.

2.4 Focal loss

The majority of early classical object identification algorithms employed the cross entropy function as the classification loss of the object detection network. However, the weight of the conventional cross entropy loss function is the same for all instances (easy positive, hard positive, easy negative and hard negatives, as shown in Figure 2). In the case of example imbalance, a large number of simple negative cases will predominate, whereas a small number of hard positive and hard negative examples will not play a role, hence complicating model optimization during training. To address the issue of imbalanced examples during model training, Lin et al. (2020) presented the focal loss function, which focuses the model's attention on the acquisition of challenging cases. As confidence in the proper class improves, the focal loss function introduces a dynamic scaling factor based on the cross-entropy function that decays to zero. As a result, this scaling factor can automatically down-weight the contribution of easy cases during training and fast centre the model's attention on challenging examples. RetinaNet, a one-stage object detector able to match the speed of earlier one-stage detectors while surpassing the accuracy of all contemporary two-stage detectors, was used to test the effectiveness of focus loss.

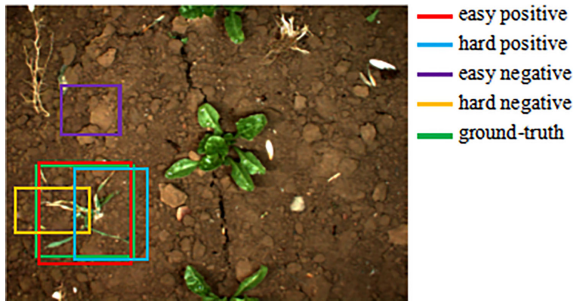


FIGURE 2
The distribution of different examples.

2.5 Warmup schedule

Adam optimizer, an adjustable learning rate gradient descent method, has become increasingly popular in recent years for training models in deep learning due to its rapid convergence and great efficiency. Nonetheless, according to a recent study (Liu et al., 2020) the problematically huge variance of the adaptive learning rate in the early stage of model training is the primary reason of poor model convergence. Ma and Yarats (2021) demonstrate that even if the model is started to a local minimum, the Adam optimizer's early parameters update may exhibit significant non-regularity. The most popular method for enhancing Adam's stability is to include a warming schedule during model training to reduce significant or divergent variance (Liu et al., 2020; Ma and Yarats 2021). Typically, the warmup schedule is established during the first few epochs or partial steps of model training. During the warmup period, the learning rate is reduced to a low amount. The training with a low learning rate increases the likelihood that the randomly initialized model's weights will stabilize.

3 Materials and methods

3.1 Weed dataset

SugarBeets2016 (University of Bonn, Germany), a huge agricultural robotics dataset for weed classification, localization, and mapping, serves as the basis for our investigations (Chebrolu et al., 2017). The collection contains three months of data acquired by the BoniRob robotic platform from a sugar beet field near Bonn, Germany. The data is collected two to three times per week, on average, and covers the pertinent growth stages for robotic intervention and weed control. The RGB images of SugarBeets2016 were captured by the JAI AD-130GE multi-spectral camera mounted on the bottom of the BoniRob robot from a top-down perspective and saved in the PNG format with lossless compression and a uniform size of 1296 936 pixels. Since the time of weed management in sugar beet fields is typically during the rapid growth period of the sugar beet leaves, rather than during the crop seedling stage when the weed morphology is most comparable to that of the crop, the sugar beet leaves have a similar morphology to the weeds. As a result, the images from the period of rapid leaf growth were chosen as our experimental material, and these images were obtained 20 days after sugar beet growth began above ground. Using the labelme tool (https://github.com/wkentaro/labelme), all the items in the 4,817 images of farmland were categorized as either sugar beet or weed, as depicted in Figure 3. A human expert manually identifies all the objects inside the detected image and encloses the region containing these things with closed rectangular boxes; the category and location information of these objects are then recorded to a local XML file. There were 9,419 sugar beet items and 9,349 weed objects in the labelled dataset.

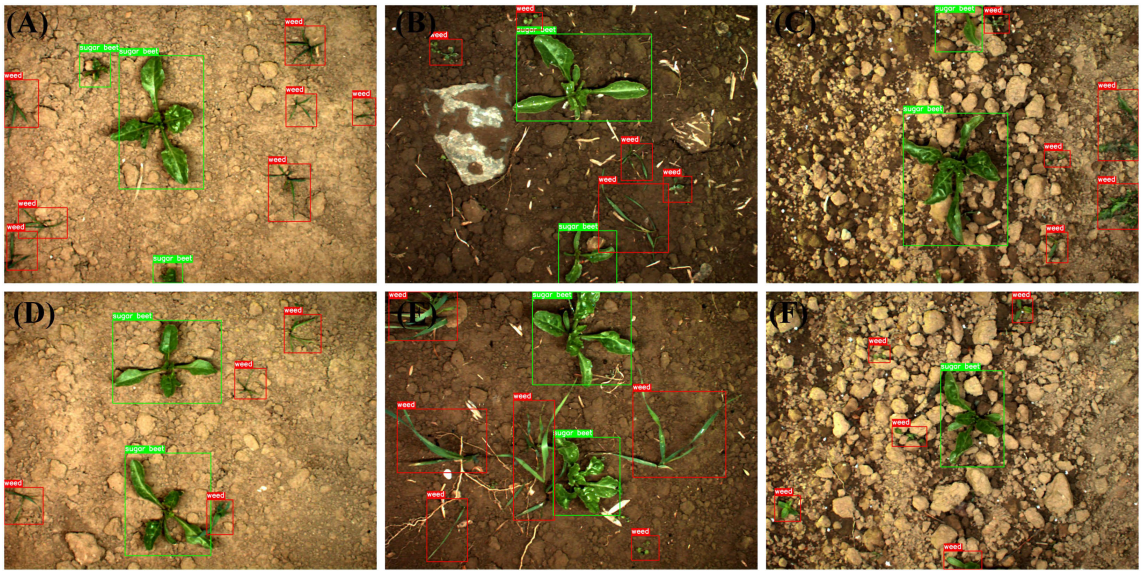


FIGURE 3
Sample images and annotations: **(A-C)** from training set, **(D-F)** from test set.

3.2 RetinaNet based weed and sugar beet detection model

3.2.1 Context module

As shown in Figure 4, the context module contains four convolutional layers, where the coefficient of 'C' represents the number of input or output channels of convolutional layer. *Conv-k* denotes a convolution layer with s stride size of 1 and a kernel size of $k \times k$ (default is 3×3), *BN* is bath normalization, *ReLU* denotes activation function. The *CB* block indicates the addition of bath normalization after the convolutional output, and *CBL* block indicates the addition of the *Leak ReLU* activation function to the *CB* module. The branch y_2 consisting of *CBL1* and *CB2* has a total stack of 2 convolutional layers, so the output of this layer has a receptive field size relative to the input equal to the receptive field size of a 5×5 convolution. Similarly, the branch y_3 consisting of *CBL1*, *CBL2* and *CB3* has a stack of *3* convolution layers, so the size of receptive field of this layer is equal to the 7×7 convolution layers. The $Y_j = [y_1, y_2, y_3]$ are calculated as shown in Equation (1), where $f_c(\cdot)$ is a convolutional operation with a kernel size of 3×3 and a step size of 1, $f_{cb}(\cdot)$ is the use of batch normalization after $f_c(\cdot)$. $f_{cbl}(\cdot)$ is the addition of Leaky ReLU activation function on top of $f_{cb}(\cdot)$. The convolutional layer outputs Y_j obtained from the input feature maps after convolutional operations in different layers are finally fused with semantic information of different scale by a concatenation method. Suppose $P_i = [P_3, P_4, P_5]$ are the feature maps from FPN, and $\dot{P}_i = [\dot{P}_3, \dot{P}_4, \dot{P}_5]$ are the feature outputs after a context module, P_i, $\dot{P}_i \in R^{H_i \times W_i}$, $H_i \times W_i$ is the size of feature map P_i. As a result, V_i is calculated be the Equation (2), where $([y_1, y_2, y_3])_{\text{concat}}$ is the stacking of Y_i in concatenation. $f_{ReLU}(\cdot)$ is the ReLU activation function applied after the stacking.

$$Y_j = \begin{cases} y_1 = f_c(f_{cb}(P_i)) \\ y_2 = f_{cb}(f_{cbl}(y_1)) \\ y_2 = f_{cb}(f_{cbl}(f_{cbl}(y_1))) \end{cases} \quad (1)$$

$$\dot{P}_i = f_c(f_{ReLU}([y_1, y_2, y_3])_{concat}) \quad (2)$$

3.2.2 General architecture of WeedNet-R

Based on RetinaNet and context modules, a one-stage object detection network, the proposed WeedNet-R weed detection network is enhanced in terms of its sophistication and applicability. As seen in Figure 5, the WeedNet-R consists of (A)

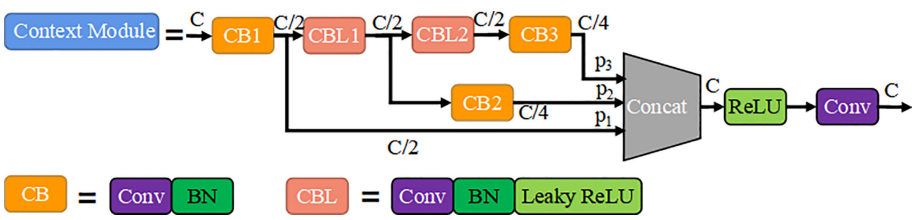


FIGURE 4
Context module.

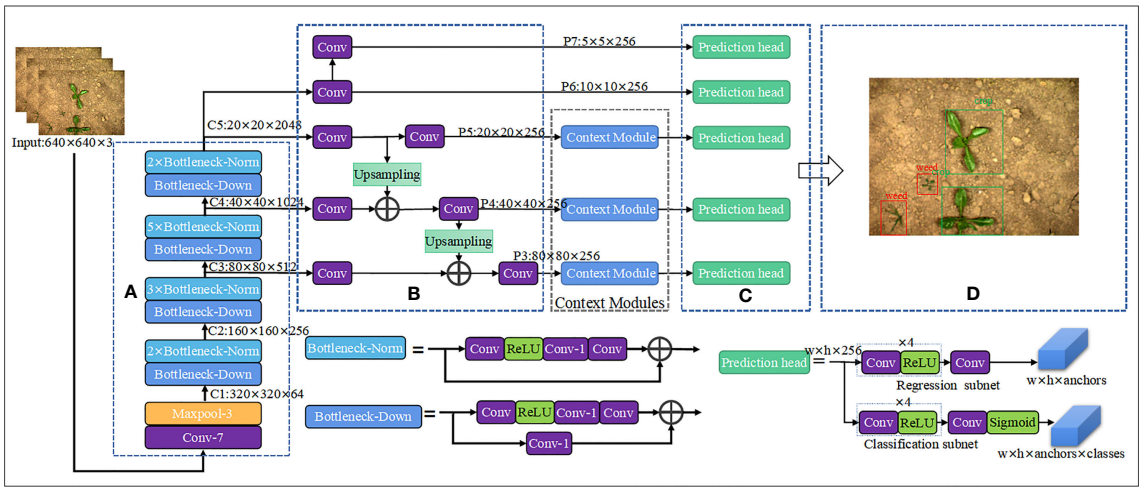


FIGURE 5
Overall architecture of WeedNet-R: **(A)** Backbone, **(B)** FPN, **(C)** Classification and regression sub-nets, **(D)** Outputs.

the feature extraction backbone (Backbone), (B) the multi-scale feature pyramid network (FPN), (C) the classification sub-net and regression sub-net, (D) the outputs.

The WeedNet-R uses ResNet50 as the foundation for feature extraction, and the C2-C5 feature layers extracted by ResNet50 are given to the FPN to generate five feature maps P3-P7 with varying scale sizes. Lastly, P3-P7 feature maps are fed into the classification and regression sub-nets for object classification and bounding box regression, respectively. WeedNet-R's classification and regression sub-nets share the same network weight parameters to reduce the size of the model. Additionally, three context modules are introduced between the three bottom layers (P3-P5) of the FPN and the classification and regression sub-nets in WeedNet-R in order to fuse context information with various receptive field sizes from different levels. In Figure 5, w and h denote the width and height of feature map respectively, *anchors* is number of anchors assigned for each spatial position of feature maps and *classes* is number of object classes.

3.2.3 Loss function

The loss function in WeedNet-R is defined as Equation (3). N_{pos} is the number of positive samples where the prior anchors match the ground-truth labels, i is any positive or negative samples, and j denotes any positive sample. $L_{total}$is the total loss function of WeedNet-R. $L_{cls}$is the classification loss function. $L_{reg}$is the regression loss function.

$$L_{total}(\{P_i\},\{t_i\}) = \frac{1}{N_{pos}}\sum_i L^i_{cls} + \frac{1}{N_{pos}}\sum_j L^j_{reg} \tag{3}$$

The classification loss function (L_{cls}) is calculated according to the focal loss of Equation (4), where $\alpha_t$and $p_t$are defined as Equation (5) and Equation (6), respectively. The hyperparameter of $\alpha \in [0,1]$is a weight factor to balance the weights of positive and negative samples, the p represents the prediction probability that the sample matches the ground truth. $log(p_t)$ is the cross-entropy function as Equation (7). A factorization $(1-p_t)^\gamma$ consisting of another hyperparameter of $\gamma$and p_t is used to balance the weight of positive and negative samples in the training process. The larger the value of γ, the larger the proportion of the loss of the simple samples in the total loss.

$$L_{cls} = -\alpha_t(1-p_t)^\gamma log(p_t) \tag{4}$$

$$\alpha_t = \begin{cases} \alpha & if \;\; y=1 \\ 1-\alpha & otherwise \end{cases} \tag{5}$$

$$p_t = \begin{cases} p & if \;\; y=1 \\ 1-p & otherwise \end{cases} \tag{6}$$

$$log(p_t) = \begin{cases} -log(p) & if(y=1) \\ -log(1-p) & otherwise \end{cases} \tag{7}$$

The regression loss (L_{reg}) represents the smooth L1 loss of the bounding box regression, which is shown in Equation (8). Here, $t_i = [t_x, t_y, t_w, t_h]$and $t_i^* = [t_x^*, t_y^*, t_w^*, t_h^*]$represent the center coordinates, width and height of the predicted bounding box and the ground-truth bounding box, respectively. The definition of $smooth_{L1}$is shown in Equation (9).

$$L_{reg} = \sum_{ie(x,y,w,h)} smooth_{L1}(t_i - t_i^*) \tag{8}$$

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & if|x|<1 \\ |x|-0.5 & otherwise \end{cases} \tag{9}$$

3.2.4 Untuned warmup schedule

Here, an untuned exponential warmup schedule is utilized to alter the learning rate during the initial phase of training. Figure 6 depicts the adjustment of the remaining training period's learning rate using the cosine annealing process. This adjustment affects the entire training period. Equation (10) determines the number of training steps consumed by the untuned exponential warmup

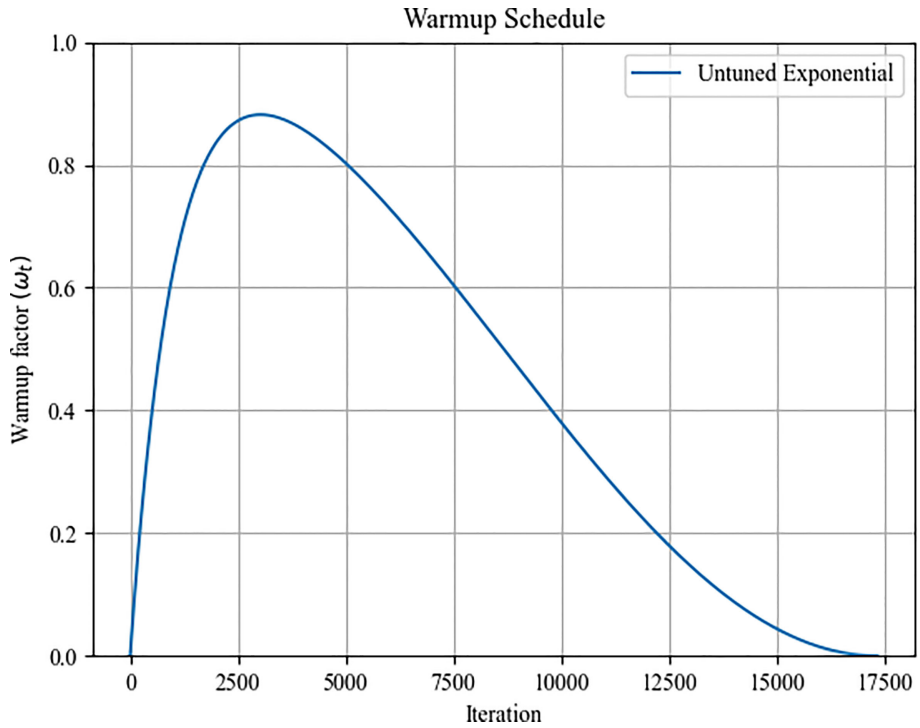


FIGURE 6
The training schedule learning rate curve.

routine throughout the training period. β_2 is the Adam optimizer second-momentum coefficient, which takes the default value of 0.999. The learning rate factor $\omega(t)$ of the untuned exponential warmup schedule is calculated by Equation (11) The final learning rate $lr(t)$ of Adam optimizer is calculated by Equation (12), which is the product of $\omega(t)$ and the initial learning rate.

$$warmup_period = 2/(1-\beta 2) \tag{10}$$

$$\omega(t) = 1 - exp(-(1-\beta 2)t) \tag{11}$$

$$lr(t) = lr_{init} \times \omega(t) \tag{12}$$

3.2.5 Crop-first non-maximal suppression

There could be a few exceptions for WeedNet-R during test, which may produce repeated predicted bounding boxes for the same object under a specified confidence threshold. To address this issue, we suggested a crop-first, non-maximal suppression technique for removing anticipated bounding boxes that are repeated. The crop-first non-maximal suppression method is similar to traditional non-maximal suppression method for object detection, but the starting-point of which is to limit the likelihood of crops being destroyed inadvertently, weeds suspected of being crops are frequently given precedence during weeding control. The method firstly separates all prediction results into two groups: sugar beet (crop) bounding boxes and weed bounding boxes. The *IoUs* of each sugar beet bounding box relative to all weed bounding boxes are then determined. Lastly, based on the results of the calculations, any predicted bounding boxes for weed with an *IoU* greater than the given threshold are eliminated.

4 Experiment settings

4.1 Experimental dataset split

At this study, the experimental dataset is randomly divided into a training-validation set and a test set in a ratio of 8:2. One tenth of the training-validation set is randomly partitioned into a validation set, which is used to observe the convergence of the model during training and to identify the best model after training. The remainder of the training-validation set is utilized for model training as the train set. Table 1 provides a summary of the employed dataset's information. The training set has 3,466 images, the validation set contains 387 images, and the test set contains 964 images. Moreover, it can be observed that the ratio of weed objects to sugar beet objects in each subset is near to one-to-one, indicating that the category of the data is balanced.

4.2 Model training and parameter setting

All object detection models in this research were developed using PyTorch 1.2 and Python 3.6 on the Windows 10 operating system. On a PC equipped with a 11 GB Nvidia GeForce GTX2080Ti GPU, a 3.50GHz Intel(R) Core(TM) i9-10920X CPU processor., and 32 GB of main memory, the models were trained

TABLE 1 Dataset split and statistics of different categories.

	Number of images	Number of weeds	Number of sugar beets
Train subset	3466	6641	6750
Valid subset	387	730	779
Test subset	964	1970	1890
Total	4817	9349	9419

and evaluated. To accelerate model convergence, the weights of Resnet50, the backbone network of WeedNet-R, were initialized by an ImageNet-pretrained model. The Adam algorithm was chosen as the model training optimization approach, and the starting learning rate (lr) was set to 0.0001. The initial momentum coefficient article β_1 was set to 0.9, whereas the second momentum coefficient β_2 was set to 0.999. The Untuned Exponential Warmup method and the cosine annealing procedure were used to alter the learning rate. The number of samples in each mini-batch was eight, and the model was iterated twenty training epochs. In addition, some training parameters or settings, such as random image flipping, matched RetinaNet (Lin et al., 2020). To compare the performance of WeedNet-R and RetinaNet (baseline), RetinaNet was trained and evaluated using the same settings as WeedNet-R. During model training, the values of the loss function after each epoch iteration were recorded, and model convergence was determined by validating the model on the validation set, as depicted in Figures 7A, B).

4.3 Evaluation metrics

In this article, measures such as mean average precision (*mAP*), size of model parameters, and forward inference time were used to evaluate the effectiveness of the neural network model, and these metrics were computed at an *IoU* threshold of 0.5. The terms of the *IoU* are specified by Formula (13). The mean average precision (*mAP*) is the mean of the average precision (*AP*) of all categories in a multi-category object detection, as defined in Equation (14).

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \tag{13}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{14}$$

where $area(B_p \cap B_{gt})$ is the area of intersection of the predicted bounding box (B_p) and the groundtruth bounding box (B_{gt}). $area(B_p \cup B_{gt})$ is the area of the union of B_p and B_{gt}. Average Precision (*AP*) is related to *precision* and *recall*, which are calculated by Equation (15) and Equation (16), respectively. Where T_p(true positive) represents the number of predicted results with *IoU* >threshold, F_P (false positive) represents the number with $IoU \leq$ threshold. F_n (false negative) represents the number of true bounding boxes not detected. The confusion matrix for T_p, F_n, and F_p is shown in Table 2.

$$Precision = \frac{T_p}{T_p + F_p} \tag{15}$$

$$Recall = \frac{T_p}{T_p + F_n} \tag{16}$$

According to the prediction confidence, a set of recall and accuracy regarding the results of the forecast are calculated individually for various confidence thresholds. Obtaining the P-R curve p(r) using recall as the horizontal axis and precision as the vertical axis. The final step in calculating the average accuracy of a single category is to solve the integral between the P-R curve and the horizontal axis, as shown in Equation (17).

$$AP = \int_0^1 p(r)dr \tag{17}$$

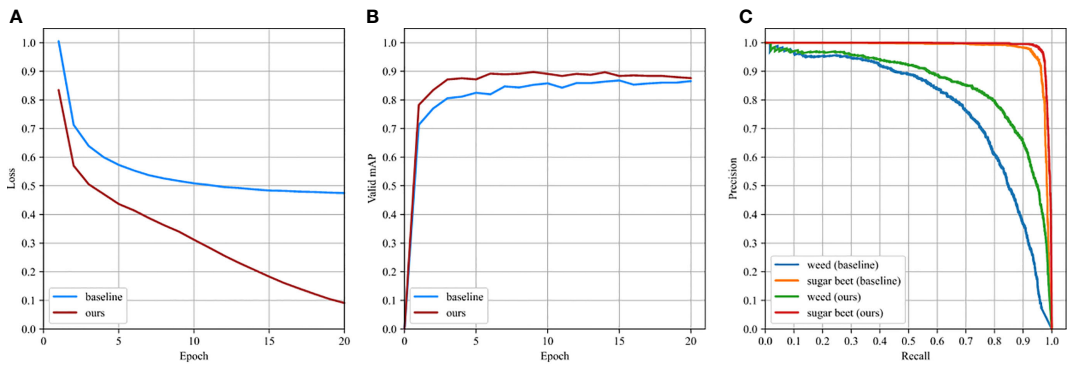


FIGURE 7
Comparison between baseline and ours: **(A)** loss curve, **(B)** validation *mAP*, **(C)** P-R curve in test subset.

TABLE 2 Definition relationships between predicted and true values.

Ground Truth	Predicted	
	Positive	Negative
Positive	True Positive (T_p)	False Negative (F_n)
Negative	False Positive (F_p)	True Positive (T_n)

4.4 Comparison with the state-of-the-art object detection models

To further demonstrate the efficacy and superiority of the enhanced model, the performance of WeedNet-R was compared to that of other advanced object detection methods, with the exception of the baseline model (RetinaNet), under identical experimental conditions. Faster R-CNN (Ren et al., 2017), SSD (Liu et al., 2016), YOLOv3 (Redmon and Farhadi, 2018), YOLOv4 (Bochkovskiy et al., 2020), CenterNet (Zhou et al., 2019), YOLOX (Ge et al., 2021). YOLOv7 (Wang et al., 2022) are the models compared. The *AP*, *mAP*, number of model parameters, forward inference time, etc. measured on a test set from sugarbeet2016 serve as the primary comparative performance indicators. In consideration of the modest variation in image input size between detectors, the input image size for training and testing is equally scaled to be 640 by 640 pixels or near to that size. In addition, both Faster R-CNN and CenterNet utilise ResNet50 and VGG16 as their extraction backbone. Yolov3 and YOLOv4 both utilise Darknet53 with 53 layers as their backbone. And the standard CSPDarknet53 was used as backbone of YOLOX. The aforesaid settings make these comparison models match an equivalent level of model parameters as the WeedNet-R, thus ensuring the fairness of the comparison.

5 Experiment results

5.1 Detection performance on dataset

Figure 7C depicts the P-R curves of the models on test data. The validation and test set images were utilised to evaluate the performance of the respective models. Table 3 displays the evaluation's results WeedNet-R only increased the number of parameters by 4.4% compared to the baseline model, while the *mAP* metric in the validation set improved by 2.93%, and the average accuracy (*AP*) for weed and sugar beet detection rose by 4.61% and 1.21%, respectively. In the meantime, the test set reflected the increased detection performance of the proposed model: the *mAP* improved by 4.65% to 92.30%, while the average accuracy of weed and sugar beet identification improved by 8.01% to 85.70% and 1.29% to 98.89%, respectively. Due to the fact that the P-R curve reflects the variable relationship between recall and accuracy at different confidence thresholds, the better the performance of the detector is represented, the closer the shape contained by the P-R curve and the coordinate axis is to a square. Figure 7C) shows that the precision of WeedNet-R is marginally higher than that of the baseline model for sugar beet plant detection and significantly higher than that of the baseline model for weed detection at the same recall rate. The aforementioned findings demonstrate that our improved weed detection system outperforms RetinaNet.

WeedNet-R outperforms the other six detection algorithms on the test set in terms of both the *AP* in individual categories and the *mAP* when compared to other sophisticated object detectors, as shown Table 4. The detection capabilities of the suggested method are much superior to those of previous methods. WeedNet-R's *mAP* is 10.65% greater than the most inaccurate Fast R-CNN. Compared with the latest SOTA object detector YOLOv7, our approach's *mAP* is 0.8% higher than it. It is worth noting that YOLOv7 applied some complex data augmentation approaches such as mosaic during training, but weedNet-R did not apply complex data augmentation approach. In addition, WeedNet-R has a lesser number of parameters than the YOLO series of algorithms (YOLO v3, YOLO v4, and YOLOX) and a somewhat greater number of parameters than SSD CenterNet and YOLOv7, which are noted for their simple architecture. Consequently, our strategy achieves optimal detection accuracy while also ensuring more acceptable model parameters. It is worth noting that the inference speed of all the models utilized in our experiments was slow, primarily due to the utilization of a relatively older graphics card from the 1080 series. This older graphics card exhibits a significant performance gap compared to the latest advanced graphics cards available in the market.

5.2 Ablation experiment results

As indicated in Table 5, a number of ablation experiments were conducted to determine the efficacy of each modified module. According to the findings of the ablation experiments, the addition of context module×5 and context module×3 to RetinaNet enhances its *mAP* by 2.17 and 1.99 percentage points, respectively. Combinations of context module×5 and untuned warmup that are added to RetinaNet increase its *mAP* metrics by 4.41%. Adding context module×3 and untuned exponential warmup learning rate

TABLE 3 Comparison of the detection performance of the RetinaNet model (baseline) before improvement and WeedNet-R.

method	valid set (%)			test set (%)			parameters	inference times (ms)
	weed *AP*	sugar-beet *AP*	*mAP*	weed *AP*	sugar-beet *AP*	*mAP*		
RetinaNet	76.38	97.27	86.82	77.69	97.61	87.65	36.4M	66.4
WeedNet-R	80.99	98.51	89.75	85.70	98.89	92.30	38.0M	52.8

TABLE 4 Comparison of the detection performance of WeedNet-R with different target detectors.

	backbone	weed *AP*	sugar-beet *AP*	*mAP*	parameters	inference time (ms)
Faster R-CNN	Resnet50	66.28%	97.02%	81.65%	28.3M	80.1
SSD	VGG16	69.16%	97.11%	83.14%	23.7M	27.2
YOLOv3	Darknet53	82.82%	96.99%	89.90%	61.5M	62.7
YOLOv4	CSPDarknet53	81.60%	97.36%	89.48%	63.9M	40.3
CenterNet	Resnet50	80.79%	97.06%	88.93%	32.7M	42.2
YOLOX	CSPDarknet53	81.90%	90.88%	86.39%	54.2M	32.1
YOLOv7	E-ELAN	84.20%	98.80%	91.50%	36.5M	21.1
WeedNet-R	Resnet50	85.70%	98.89%	92.30%	38.0M	52.8

adjustment approach to RetinaNet results in WeedNet-R, which achieves the maximum *mAP* value of 92.30%. Specifically, its *AP* scores for weed detection are improved by 8.01% to 85.70%. In conclusion, adding three context modules or five context modules is effective, although the trick with three context modules yields slightly higher detection accuracy with fewer parameters.

5.3 Visualization

This section validates the usefulness of WeedNet-R for real input images by conducting visualization experiments and an analysis of the test dataset. As depicted in Figures 8A–I) are the prediction results of the experimental models for three representative images from the test dataset with varying background complexity, and Figure 8J is the ground truth. The complexity of the backdrop and the number of objects in the three selected images increase from left to right in order to evaluate the performance of various algorithms under varying scenarios. As shown in Figures 8A, B), compared to RetinaNet, the proposed method provides more accurate prediction results, greater confidence in the classification of the items within the predicted bounding boxes, and more precise placements for the predicted bounding boxes. Compared to other sophisticated detection algorithms, sugar beet identification performance was comparable, with the exception of CenterNet, which made more incorrect predictions. For weed detection, Faster R-CNN suffers from a severe case of repeated prediction, and SSD is unsuitable for small objects. Both YOLOV3 and YOLOV4 have instances of missing marijuana detection. The CenterNet makes inaccurate predictions of weeds and has poor trust in the accuracy of its predictions. The recently popular YOLOX and YOLOv7 algorithms have a decent detection performance, yet there have been instances were weeds were not detected.

5.4 Optimization for repeated prediction boxes

The prediction results before and after crop-first non-maximal suppression method are shown in Figures Figures 9A–F. (Figure 9A) illustrates a limited number of exceptions in prediction results, which produces repeated predicted bounding boxes for the same object under a confidence level of 0.5. This problem is mitigated by applying crop-first non-maximal suppression method as shown in Figure 9B). Here, the *IoU* threshold for crop-first non-maximal suppression is set to 0.5. As demonstrated in Table 6, this strategy enhances weed detection precision by 0.2%, but has no influence on the detection accuracy and recall of sugar beet. Consequently, the crop-first non-maximal suppression method accomplishes the goal of eliminating duplicate anticipated bounding boxes and minimizing the possibility of false crop removal.

TABLE 5 Ablation experiments results.

	performance(%)				
	RetinaNet				WeedNet-R (ours)
+context module ×5	–	√	–	√	–
+context module ×3	–	–	√	–	√
+untuned warmup	–	–	–	√	√
weed *AP*	77.69	81.40	81.14	85.19	85.70
sugar beet *AP*	97.61	98.24	98.15	98.94	98.89
mAP	87.65	89.82	89.64	92.06	92.30

context module×5: Add a context module after each of the P3~P7 layers of the FPN network output. context module×3: Add a context module after each of the P3~P5 layers of the FPN network output. untuned warmup: Set untuned exponential warmup schedule during model training. "-" indicates that the method does not contain the corresponding module, and "√" indicates that the method contains the corresponding module.

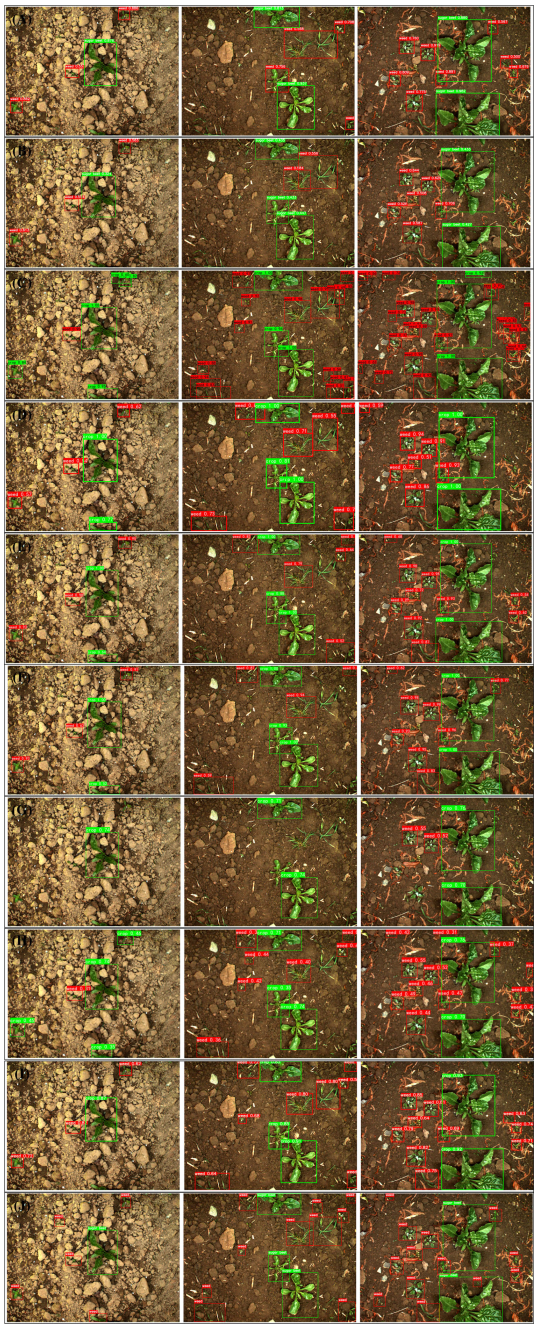


FIGURE 8
Detection results of different models: **(A)** WeedNet-R, **(B)** RetinaNet, **(C)** Faster R-CNN, **(D)** SSD, **(E)** YOLOv3, **(F)** YOLOv4 **(G)** CenterNet, **(H)**YOLOX, **(I)** YOLOv7, **(J)** Ground-truth.

5.5 Validation on other public weed dataset

To validate the effectiveness of our improved method, we conducted additional experiments using a another weed dataset (Ravirajsinh, 2020) containing while keeping the experimental configurations consistent. The results are presented in Table 7, showcasing the performance of our proposed approach on publicly available weed datasets. Notably, our WeedNet-R model achieved an *mAP* metric of up to 85.26%. It is important to note that the improvements in detection performance of WeedNet-R, compared to the original RetinaNet, were relatively modest on this new weed dataset, with a increase of 0.57% in the *mAP* metric.

5.6 Discussion

Experiments comparing our proposed approach to other sophisticated object detection algorithms demonstrate that the suggested algorithm has the highest average precision of individual categories and the highest total *mAP* metrics, as well as the highest detection accuracy. Despite the fact that WeedNet-model R's parameters are greater than those of the original RetinaNet, the average detection time is shorter. This is due to the fact that the less precise RetinaNet generates more inaccurate predictions during detection, which increases the time required for post-processing actions such as non-maximal suppression. Ablation studies confirmed the efficacy of our enhancements to RetinaNet. Experiments indicate that the optimal detection performance is achieved by adding a context module after each of P3 to P5 of the FPN outputs and configuring an untuned exponential warmup schedule during model training. Adding a context module to each of the P3 to P5 levels of the FPN reduces several parameters of the model and somewhat increases model recognition accuracy compared to adding a context module to each layer of the FPN outputs. This may be because the P6-P7 layers of the FPN are part of the high-level feature maps, which include sufficient deep context information to identify huge objects.

In addition, under natural light, the colour, morphology, and texture of sugar beet plants (especially early sugar beet seedlings) closely resemble those of field weeds. This resemblance exacerbates the difficulty of differentiating weeds from crops and is the primary cause of the model's misclassification of identified objects. In reality, this frequently shows as misclassification of objects within the projected bounding boxes or the generation of several predictions for the same object. We presented a crop-first non-maximum suppression strategy for a problem involving repeated predicted bounding boxes for the same object. Objects projected to be both weed and crop repeatedly at the same location are classed as crop in order to reduce the chance of crops being removed in error.

The model we proposed demonstrates effective performance on other publicly available weed datasets, albeit with a relatively modest increase of 0.57% in the mean average precision (*mAP*) metric compared to the baseline. We attribute this outcome, at least in part, to the limited number of images available in evaluated dataset. Building a robust model with a small number of images poses significant challenges. The scarcity of large-scale publicly available weed datasets remains a common obstacle in the domain of weed detection utilizing deep learning approaches. To overcome this challenge, future endeavors should focus on the acquisition and curation of larger and more diverse weed datasets. Therefore, we have made our annotated weed dataset based on Sugarbeet2016 publicly available to support the research community and facilitate future advancements in this field. Our annotated dataset comprises

10.3389/fpls.2023.1226329

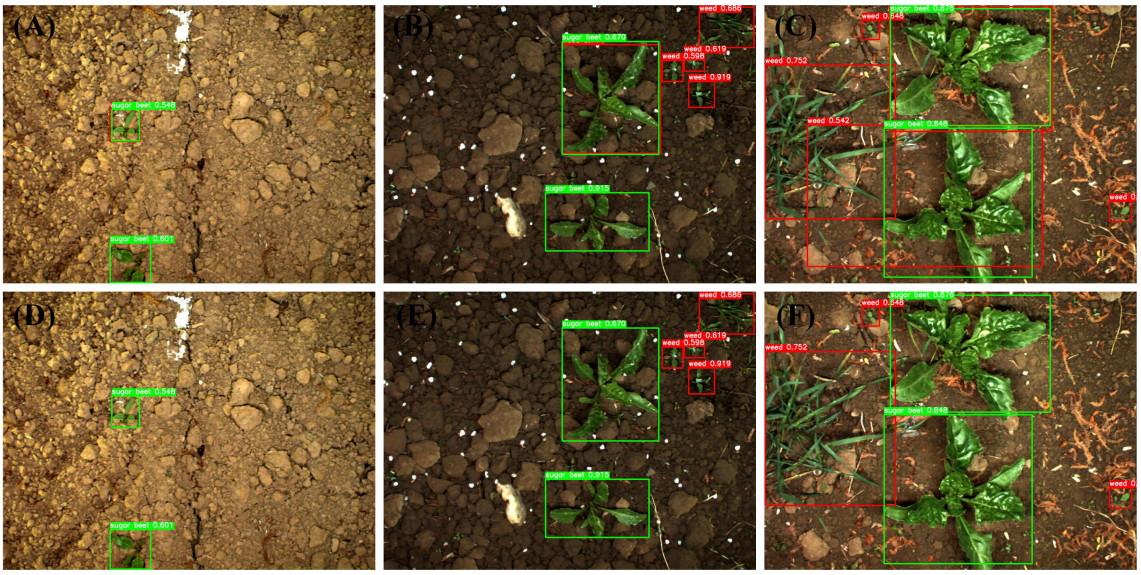


FIGURE 9
Comparison of prediction results before and after crop-first non-maximal suppression method: **(A-C)** are before, **(D-F)** are after.

TABLE 6 Comparison before and after using the crop-first non-maximal suppression method.

	weed		sugar beet	
	precision	recall	precision	recall
before	76.5%	82.5%	97.0%	96.5%
after	76.7%	81.9%	97.0%	96.5%

4,817 images and 18,768 annotations, making it one of the most extensive bounding box-based datasets for weed detection.

6 Conclusion

Detecting and identifying weeds in the field is a crucial step in attaining autonomous weed management. While the remarkable resemblance in color, morphology, texture, and other features between weeds and crops in the field under natural lighting conditions increases the complexity of machine vision-based weed detection. Theoretical and methodological developments in deep learning have produced new tools for visual identification problems, such as weed detection. Due to the complexity of weed detection tasks in the field, deep learning-based approaches for weed detection continue to be of significant scientific relevance. In this research, we present an enhanced detection model, WeedNet-R, which is based on RetinaNet and has greater detection accuracy than the original model and other sophisticated object detectors. WeedNet-R has the highest *mAP* for weed detection in sugar beet fields at 92.30%.

TABLE 7 Comparison between the RetinaNet and WeedNet-R on anther public weed dataset.

method	weed *AP*(%)	sesame *AP*(%)	*mAP*(%)
RetinaNet	86.21	83.16	84.69
WeedNet-R	$86.91 + ^{0.70}$	$83.61 + ^{0.45}$	$85.26 + ^{0.57}$

In this study, we relate the lack of detection accuracy of the baseline model to the insufficient size of its effective receptive field. In order to increase the effective receptive field of the feature extraction layers, context modules are added to the neck structure of RetinaNet. During model training, an untuned exponential warmup schedule is implemented in order to improve the optimal solution search capability. The *mAP* of WeedNet-R proposed in this article was enhanced by 4.65% as compared to the original RetinaNet as a result of the aforementioned enhancements. With only a little improvement in model parameters, the accuracy of weed detection increased by 8.01% to 85.70%, and the accuracy of sugar beet plant recognition increased by 1.2% to 98.89%. In addition, the crop-first non-maximal suppression strategy we presented reduces the few occurrences in which the same object is predicted many times by the model. The detection performance of the proposed approach is superior to that of other algorithms in the SugarBeet2016 dataset, but there is still a little room for improvement in weed detection. Therefore, continuing to optimize the structure of our model is our future efforts. And because larger image dateset would be beneficial for training of convolutional neural networks, the model's performances may be further optimized by obtaining more weed images. In addition, the size of model's parameters is key to the performance of model forward inference. Perhaps it is well worth considering to boost the model's detection speed by refining the model's backbone or implementing an anchor-free strategy for boosting the model's detectionsssss speed. In conclusion, pursuing a more precise and faster weed identification model to deal with the

complex farming environment will be the primary focus of our future work.

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

ZG: Software, Data curation, Resources, Visualization, Writing – original draft, Writing – review and editing. YL: Conceptualization, Methodology, Writing – original draft, Writing – review and editing. HG: Validation, Super-vision, Writing – review and editing. XL: Validation Formal analysis, Investigation. MZ: Validation, Investigation, Visualization. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation of China (31760342), the Science and Technology Major Project of Guangxi, China (Gui Ke 2018-266-Z01), and the Middle-aged and Young Teachers' Basic Ability Promotion Project of Guangxi (Grant No. 2021KY0015), Hubei Key Laboratory of Intelligent Robot (Grant No.HBIR202108).

Acknowledgments

We would like to thank the editor and reviewers for their valuable input, time, and suggestions to improve 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.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2023.1226329/full#supplementary-material

References

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1–1. doi: 10.1109/TPAMI.2016.2644615

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

Bosilj, P., Aptoula, E., Duckett, T., and Cielniak, G. (2020). Transfer learning between crop types for semantic segmentation of crops versus weeds in precision agriculture. *J. Field Robotics* 37, 7–19. doi: 10.1002/rob.21869

Champ, J., Mora-Fallas, A., Goëau, H., Mata-Montero, E., Bonnet, P., and Joly, A. (2020). Instance segmentation for the fine detection of crop and weed plants by precision agricultural robots. *Appl. Plant Sci.* 8. doi: 10.1002/aps3.11373

Chebrolu, N., Lottes, P., Schaefer, A., Winterhalter, W., Burgard, W., and Stachniss, C. (2017). Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields. *Int. J. Robotics Res.* 36, 1045–1052. doi: 10.1177/0278364917720510

Deng, J., Guo, J., Zhou, Y., Yu, J., Kotsia, I., and Zafeiriou, S. (2019). RetinaFace: single-stage dense face localisation in the wild. *arXiv preprint arXiv* 1905, 00641. doi: 10.48550/arXiv.1905.00641

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. *arXiv preprint arXiv* 2010, 11929. Available at: https://arxiv.org/abs/2010.11929.

Espejo-Garcia, B., Mylonas, N., Athanasakos, L., Fountas, S., and Vasilakoglou, I. (2020). Towards weeds identification assistance through transfer learning. *Comput. Electron. Agric.* 171, 105306. doi: 10.1016/j.compag.2020.105306

Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., and Nardi, D. (2019). Crop and Weeds Classification for Precision Agriculture Using Context-Independent Pixel-Wise Segmentation. in 2019 Third IEEE International Conference on Robotic Computing (IRC) (Naples, Italy: IEEE), 146–152. doi: 10.1109/IRC.2019.00029

Gao, J., French, A. P., Pound, M. P., He, Y., and Pieters, J. G. (2020). Deep convolutional neural networks for image-based convolvulus sepium detection in sugar beet fields. *Plant Methods* 16. doi: 10.1186/s13007-020-00570-z

Ge, Z., Liu, S., Wang, F., Li, Z., and Sun, J. (2021). YOLOX: exceeding YOLO series in 2021. *arXiv preprint arXiv* 2107, 08430. doi: 10.48550/arXiv.2107.08430

Hu, K., Coleman, G., Zeng, S., Wang, Z., and Walsh, M. (2020). Graph weeds net: a graph-based deep learning method for weed recognition. *Comput. Electron. Agric.* 174, 105520. doi: 10.1016/j.compag.2020.105520

Jiang, Y., Li, C., Paterson, A. H., and Robertson, J. S. (2019). DeepSeedling: deep convolutional network and kalman filter for plant seedling detection and counting in the field. *Plant Methods* 15, 141. doi: 10.1186/s13007-019-0528-3

Jin, X., Che, J., and Chen, Y. (2021). Weed identification using deep learning and image processing in vegetable plantation. *IEEE Access* 9, 10940–10950. doi: 10.1109/ACCESS.2021.3050296

Li, Y., Guo, Z., Shuang, F., Zhang, M., and Li, X. (2022). Key technologies of machine vision for weeding robots: a review and benchmark. *Comput. Electron. Agric.* 196, 106880. doi: 10.1016/j.compag.2022.106880

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollar, P. (2020). Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 42, 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. *arXiv:1512.02325 [cs]* 9905, 21–37. doi: 10.1007/978-3-319-46448-0_2

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., et al. (2020). "On the variance of the adaptive learning rate and beyond," in *Proceedings of the Eighth International Conference on Learning Representations (ICLR 2020)*.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang,, et al. (2021). "Swin transformer: hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*. 10012–10022.

Lottes, P., Hrferlin, M., Sander, S., and Stachniss, C. (2016). Effective vision-based classification for separating sugar beets and weeds for precision farming. *J. Field Robotics* 34. doi: 10.1002/rob.21675

Ma, J., and Yarats, D. (2021). On the adequacy of untuned warmup for adaptive optimization. *arXiv preprint arXiv* 1910, 04209. doi: 10.1609/aaai.v35i10.17069

Najibi, M., Samangouei, P., Chellappa, R., and Davis, L. S. (2017). "Ssh: single stage headless face detector," in *Proceedings of the IEEE international conference on computer vision*. 4875–4884.

Olsen, A., Konovalov, D. A., Philippa, B., Ridd, P., Wood, J. C., Johns, J., et al. (2019). DeepWeeds: a multiclass weed species image dataset for deep learning. *Sci. Rep.* 9, 2058. doi: 10.1038/s41598-018-38343-3

Ravirajsinh, D. (2020) *Crop and weed detection data with bounding boxes*. Available at: https://www.kaggle.com/datasets/ravirajsinh45/crop-and-weed-detection-data-with-bounding-boxes.

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. (2017). "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.

Sa, I., Chen, Z., Popovic, M., Khanna, R., and Siegwart, R. (2017). "Weednet: dense semantic weed classification using multispectral images and mav for smart farming," in *IEEE Robotics and Automation Letters*. 99.

Tang, X., Du, D. K., He, Z., and Liu, J. (2018). "Pyramidbox: a context-assisted single shot face detector," in *Proceedings of the European conference on computer vision (ECCV)*. 797–813.

Wang, A. (2019). A review on weed detection using ground-based machine vision and image processing techniques. *Comput. Electron. Agric.* 15. doi: 10.1016/j.compag.2019.02.005

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 e-prints*. doi: 10.48550/arXiv.2207.02696

Wang, A., Xu, Y., Wei, X., and Cui, B. (2020). Semantic segmentation of crop and weed using an encoder-decoder network and image enhancement method under uncontrolled outdoor illumination. *IEEE Access* 8, 81724–81734. doi: 10.1109/ACCESS.2020.2991354

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., et al. (2022a). Dino: detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv* 2203, 03605. doi: 10.48550/arXiv.2203.03605

Zhang, H., Wang, Z., Guo, Y., Ma, Y., Cao, W., Chen, D., et al. (2022b). Weed detection in peanut fields based on machine vision. *Agriculture* 12 (10), 1541. doi: 10.3390/agriculture12101541

Zhou, X., Wang, D., and Krhenbühl, P. (2019). Objects as points. *ArXiv Preprint ArXiv* 1904, 07850. doi: 10.48550/arXiv.1904.07850

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

Zhuang, J., Li, X., Bagavathiannan, M., Jin, X., Yang, J., Meng, W., et al. (2022). Evaluation of different deep convolutional neural networks for detection of broadleaf weed seedlings in wheat. *Pest Manage. Sci.* 78, 521–529. doi: 10.1002/ps.6656

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 09 August 2023
DOI 10.3389/fpls.2023.1234616

Check for updates

OPEN ACCESS

EDITED BY
Hao Lu,
Huazhong University of Science and Technology, China

REVIEWED BY
Yanan Li,
Wuhan Institute of Technology, China
Yinglun Li,
Nanjing Agricultural University, China

*CORRESPONDENCE
Hyongsuk Kim
hskim@jbnu.ac.kr

RECEIVED 05 June 2023
ACCEPTED 06 July 2023
PUBLISHED 09 August 2023

CITATION
Ilyas T, Lee J, Won O, Jeong Y and Kim H (2023) Overcoming field variability: unsupervised domain adaptation for enhanced crop-weed recognition in diverse farmlands.
Front. Plant Sci. 14:1234616.
doi: 10.3389/fpls.2023.1234616

COPYRIGHT
© 2023 Ilyas, Lee, Won, Jeong and Kim. 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.

Overcoming field variability: unsupervised domain adaptation for enhanced crop-weed recognition in diverse farmlands

Talha Ilyas[1,2], Jonghoon Lee[2], Okjae Won[3], Yongchae Jeong[2,4] and Hyongsuk Kim[2,4]*

[1]Division of Electronics and Information Engineering, Jeonbuk National University, Jeonju-si, Republic of Korea, [2]Core Research Institute of Intelligent Robots, Jeonbuk National University, Jeonju-si, Republic of Korea, [3]Production Technology Research Division, National Institute of Crop Science, Rural Development Administration, Miryang, Republic of Korea, [4]Division of Electronics Engineering, Jeonbuk National University, Jeonju-si, Republic of Korea

Recent developments in deep learning-based automatic weeding systems have shown promise for unmanned weed eradication. However, accurately distinguishing between crops and weeds in varying field conditions remains a challenge for these systems, as performance deteriorates when applied to new or different fields due to insignificant changes in low-level statistics and a significant gap between training and test data distributions. In this study, we propose an approach based on unsupervised domain adaptation to improve crop-weed recognition in new, unseen fields. Our system addresses this issue by learning to ignore insignificant changes in low-level statistics that cause a decline in performance when applied to new data. The proposed network includes a segmentation module that produces segmentation maps using labeled (training field) data while also minimizing entropy using unlabeled (test field) data simultaneously, and a discriminator module that maximizes the confusion between extracted features from the training and test farm samples. This module uses adversarial optimization to make the segmentation network invariant to changes in the field environment. We evaluated the proposed approach on four different unseen (test) fields and found consistent improvements in performance. These results suggest that the proposed approach can effectively handle changes in new field environments during real field inference.

KEYWORDS

crop-weed recognition, domain adaptation, precision agriculture, artificial intelligence, crop phenotyping, agricultural operations

1 Introduction

Deep Learning (DL) techniques have been successful in detecting and recognizing objects in images and videos. These techniques are now being applied to agriculture, particularly in the automatic detection and classification of weeds (Khan et al., 2020). This is a difficult problem because weeds and crops often have similar colors (green vs green),

shapes, and textures (Adhikari et al., 2019; Sarvini et al., 2019). Weeds are plants that negatively impact crop growth and yields by competing for resources such as water, sunlight, air, and nutrients. They can also interfere with crop growth through the release of chemicals (Patel and Kumbhar, 2016; Iqbal et al., 2019). Effective weed control is therefore necessary to support crop growth. In addition, what is considered a weed in one setting may be a crop in another. The increasing global population, expected to reach 9 billion by 2050, will require a 70% increase in agricultural production (Radoglou-Grammatikis et al., 2020). However, the agricultural industry will face challenges such as limited cultivation land and the need for more intensive production. Climate change and water scarcity will also impact productivity. Precision agriculture can help address these challenges (Lal, 1991; Seelan et al., 2003).

Farmers must use various strategies to control weeds, including preventative measures (manual weeding), cultural techniques like field hygiene (low weed seed bank), mechanical methods like mowing and tilling, biological methods like using natural enemies of weeds (insects or grazing animals), and chemical methods such as herbicide application (Tu et al., 2001; Melander et al., 2005). Automated weed control systems, which can reduce labor costs and minimize herbicide use, have become desirable as labor costs have increased and concerns about health and the environment have grown (Durmuş et al., 2015; Nicolopoulou-Stamati et al., 2016). Moreover, due to a lack of interest among younger people in joining the agriculture industry, there is a shortage of labor (Sarvini et al., 2019). This shortage, combined with the need for efficient and cost-effective weed control, has made automated weeding methods more necessary than ever before (Lameski et al., 2018).

On other hand automated weed detection systems follow a series of steps to identify and classify weeds in images. These steps include acquiring images, pre-processing them, extracting features, and detecting and classifying weeds (Pantazi et al., 2016; Parra et al., 2020). Deep learning approaches have been successful in achieving accurate results in recognizing crops and weeds in real-world conditions (Li and Tang, 2018). The key challenge in these systems is distinguishing between crops and weeds (Khan et al., 2020; Matloob et al., 2020). These systems typically use fully convolutional networks (FCNs) to perform semantic segmentation, which involves labeling each pixel in an image with a specific class (such as crop or weed) (Parra et al., 2020; Coleman et al., 2022).

One of the main challenges in developing an automatic weed management system is accurately detecting and recognizing weeds in crops. This can be difficult because weeds and crops often have similar colors, textures, and shapes, and may appear differently at different growth stages (Sarvini et al., 2019; Khan et al., 2020; Ilyas et al., 2022). Other challenges include occlusion, variations in color and texture due to lighting and illumination, and the presence of motion blur and noise in images (Sa et al., 2016). The species of weeds can also vary based on geographical location, crop variety, weather conditions, and soil type (Kriticos et al., 2006). All of these factors can make it difficult to classify plants accurately.

Several studies have shown advancements in this area. For example, Tavakoli et al. (2021) utilized marginal loss function in CNN training for better classification. Raja et al. (2019) developed crop signaling for improved detection, and Moazzam et al. (2022) used a CNN ensemble for high accuracy detection in sesame fields. Gao et al. (2020) explored DL-based object detectors for weed detection in sugar beet fields, while Picon et al. (2022) and Peng et al. (2022) investigated synthetic images and RetinaNet adaptations, respectively, for better crop-weed recognition. However, there remains a challenge with these DL models: they often produce confident predictions on the dataset from the source domain (original farm) but underperform on data from different domains (other farms) due to domain shift (Vu et al., 2019). This is further complicated by the high cost of acquiring labeled data for each new domain, especially for semantic segmentation where each pixel must be labeled (Tranheden et al., 2021).

Recent research has explored unsupervised domain adaptation (UDA) to improve the adaptability of crop-weed segmentation systems. Gogoll et al. (2020) devised a method utilizing cycle GANs to regenerate source data in the target domain style while maintaining semantic and structural object consistency. The result was a considerable enhancement in the generalization capabilities of fully convolutional networks (FCNs), resulting in around a 10% increase in the mIOU metric on two different source-target domain pairs.

Similarly, Kendler et al. (2022) tackled the issue of low-level variability in plant disease recognition training data. By dividing images into multiple patches, they increased dataset diversity and improved CNN generalizability without needing environmental modification. This resulted in a 20% improvement in classification accuracy over the baseline. For corn yield prediction across different regions, Ma et al. (2021) presented a CNN training strategy based on unsupervised adaptive domain adversarial training. Li et al. (2021) proposed an intermediate domain approach to decrease the domain gap in maize residue segmentation. However, the application of this approach may be limited as the intermediate domain is problem-specific.

Our approach is based on the idea that the classification of a plant as a crop or weed should not depend on the farm environment, soil type, the specific sensor (camera) used, or other low-level sources of variability. These sources of variability are uninformative for crop-weed recognition, but can significantly affect the predictions of CNNs.

In this paper our aim is to reduce the domain gap between the extracted features, from source and target domain, *via* adversarially optimized deep feature alignment and entropy minimization. Additionally, we introduce a novel regularization technique to improve the convergence of CNNs. In contrast to previous UDA works, we also explore the effectiveness of few-shot training strategy in the context of UDA, called few-shot supervised domain adaptation (SDA). Few-shot SDA involves fine-tuning the model on a small amount of labeled data from the target domain to improve its performance on that domain. The main advantage of

few-shot SDA is that it can be used to quickly adapt a model to a new domain with minimal labeled data.

Our main contributions can be summarized as follows:

- A deep adversarial optimized framework for UDA and few-shot SDA.
- Augmentation scheduling strategy for improved regularization and convergence.
- A versatile dataset for fine-grained crop-weed recognition collected from five different fields with different setups.

2 Materials

2.1 Dataset construction

Our proposed approach was tested on a bean field dataset collected over the past one and a half years at five different locations and farms in South Korea using different image acquisition platforms. The dataset includes a number of variations in real-field conditions such as the field seeding bed (Gebrekidan, 2003), environment, weed density, plant scales, and sizes. To evaluate the performance of our approach, we selected five farms with different conditions and data variations as shown in Figure 1 and Table 1.

Beans are a crop that help improve soil health through nitrogen fixation, adding nitrogen back into the soil. Because of this ability, beans are often included in crop rotation plans, as nitrogen is an essential nutrient for growing strong and productive plants (Aschi et al., 2017). In countries like South Korea, where only 20% of the land is suitable for cultivation, it is especially important to use crops that can improve soil health. The collected dataset includes a bean crop and various types of weeds, but for the purposes of the crop-weed recognition task, we have grouped all the weeds into a single category. Table 1 summarized the characteristics of the dataset.

2.2 Field data distribution

In order to make our dataset suitable for domain adaptation, such as the representation shown in Figure 2, we considered the case of data collected at five different locations and fields, designated as F_A, F_B, F_C, F_D, and F_S, as shown in Figure 1. This is a specific example of domain adaptation across various scenarios, in which we aim to build a more robust system by transferring the visual characteristics from one field to another. In this case, we assume that the conditions of each field are different, meaning that each field may have a different weed density, seeding system, image acquisition system, and crop size. The visual characteristics of the fields used for data collection are displayed in Figures 1, 2 illustrates the visual attributes of various seeding bed systems across different fields.

2.3 Source and target datasets

In order to create the source and target datasets for unsupervised domain adaptation (UDA) in our experiments, we designated the field with the largest number of data available i.e., F_S as the source field, and all the other fields (F_A, F_B, F_C, F_D) as the target fields. Based on this grouping, we consider the following combinations across the five fields for evaluation: S→S, S→A, S→B, S→C, S→D. We train the network using data from the source domain (F_S) and test it against all the target domain datasets (F_A, F_B, F_C, F_D).

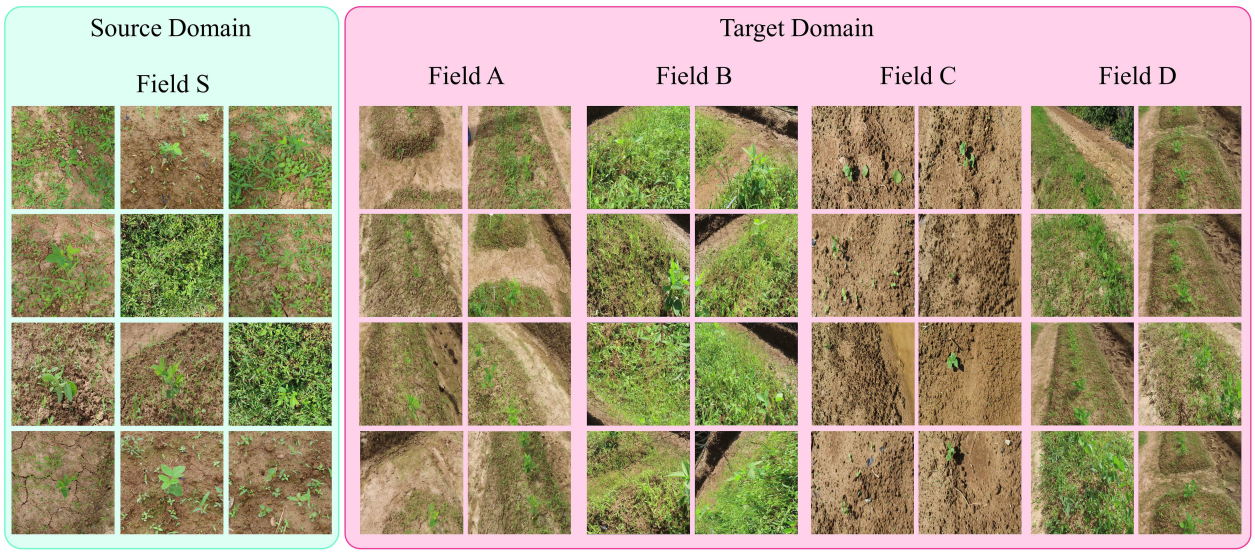


FIGURE 1
Representative images from different fields to collect data. The source domain (Field S) data is collected using handheld cameras in the form of images and are labelled by human annotators. Whereas the target domain data (Field A, B, C and D) is collected from various fields with a camera mounted on a moving platform in form of videos.

TABLE 1 Characteristics of Crop-Weed Recognition Datasets for unsupervised domain adaptation.

Field	No. of Images	Vegetation Density	Image Acquisition Platform	Seeding System
S	1893	Varying	Stationary	Flat Bed
A	202	Medium Dense	Moving	Ridge Bed
B	79	Dense	Moving	Ridge Bed
C	199	Sparce	Moving	Flat Bed
D	157	Medium Dense	Moving	Ridge Bed

3 Methodology

Here, we present our methodology for deep feature adaptation in context of UDA for crop-weed segmentation in unconstrained real-field environments. We also compare UDA approach with few-shot SDA for completeness. This section consists of the following sub-sections: (i) clearly defining the problem statement, (ii) introducing the architecture of the full framework, (iii) explaining the augmentation scheduling strategy which improves the performance of our framework, (iv) defining the learning objectives (loss functions), and (v) providing implementation details.

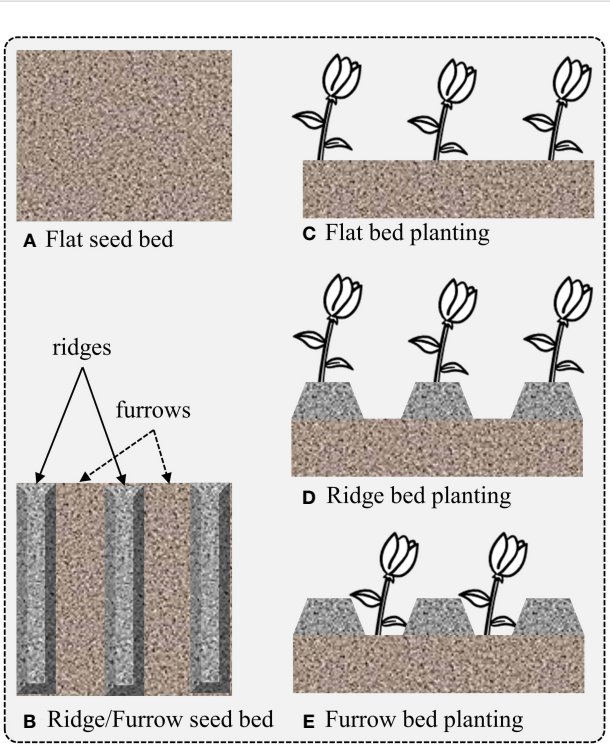


FIGURE 2
Representation of field seeding bed systems depending on the chosen planting method. Different planting methods can lead to varying crop yields for different crops. **(A)** Flat seed bed, **(B)** seeding bed with ridges and furrows, **(C)** plantation on flat seeding bed, **(D)** plantation on ridges and **(E)** plantation on furrows.

3.1 Problem definition

For better generalization we cast our problem as few-shot SDA because UDA can be simply defined as zero-shot SDA. Under these setting consider we are given a labelled soured dataset $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{N}$, where $D_s \subseteq \{F_s\}$ and N is the total number of images in D_S. Similarly, we have target domain datasets, $D_t = \{(x_i^t, y_i^t)\}_{i=1}^{M_t}$, from which we can only access j labelled images, here j ϵ {0,1,2,…, M_t} and M_t being the total number of images in *t-th* target domain dataset, and $D_s \subset \{F_A, F_B, F_C, F_D\}$. Here $x_i \in \mathbb{R}^{H\times W\times 3}$ is RGB-image and $y_i \in \mathbb{R}^{H\times W}$ is its corresponding label. We define j-shot SDA as randomly selecting j labelled images from each target domain datasets and using them for finetuning the network. The case of 0-shot SDA (j=0) is equivalent to unsupervised domain adaptation (UDA). For experiments we only consider $j = 0,1,3,M_t$. A graphical illustration that demonstrates the distinctions between UDA and few-shot SDA is displayed in Figure 3.

3.2 Augmentation scheduling

In conventional data augmentation strategies employed for training deep neural networks, a constant probability is applied for data augmentation, which often comprises a mixture of geometric and noise transformations. However, our proposed method diverges from this practice by progressively increasing the frequency of data augmentation as training advances, with each type of augmentation treated distinctly. The concept of increasing the augmentation probability finds parallels in the training of PA-GANs (Zhang and Khoreva, 2019), where both the generator and discriminator of a GAN grow progressively. Starting at low resolution, layers are incrementally added to enhance the resolution over time, thereby enabling the model to initially learn coarse-level structures, and then gradually learn fine-level details as training continues.

In contrast, the proposed technique involves adjusting the intensity or probability of data augmentation over time, but does not involve changing the architecture of the model itself over the course of training. In the augmentation scheduling of GANs, the emphasis is on enhancing the stability and efficacy of training through gradual growth of the model's structure. Conversely, augmentation scheduling focuses on presenting the model with an increasingly diverse and challenging array of training samples over time. While both techniques involve a form of progressive or scheduled change during training, they target different aspects of the training process. The augmentation scheduling technique is primarily about the model, while the augmentation scheduling technique is about the data.

Here we divide different augmentations into three categories depending on their characteristics:

- Geometric augmentations (G), augmentations which effect the entire image-label pair (x^s,y^s).
- Noise (distortion) augmentation (D), which only effect the original image (x^s) and labels (y^s) remain unchanged.

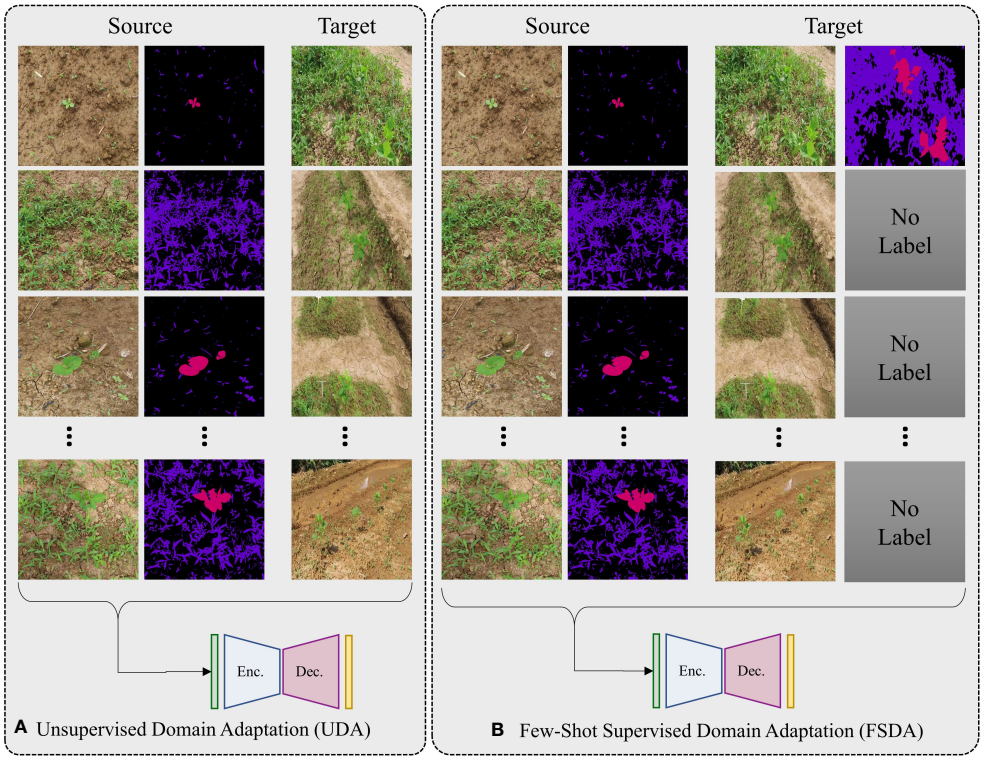


FIGURE 3
A graphical representation illustrating differences between UDA and few-shot SDA. In UDA **(A)**, there a relatively large number of unlabeled target domain data is available for use during training. In few-shot SDA **(B)**, only a small number of labeled samples (typically one or two) are available for training. The figure shows an example of 1-shot SDA as only one labelled sample is provided.

- Collage Augmentation (C) (Chiang et al., 2019), which generate a collage of multiple image-label pairs (x_i^s, y_i^s) in the dataset. Mathematically it can be expressed as,

$$(x_i^c, y_i^c) = \mathcal{C}(\{(x_i^s, y_i^s)\}_{i=1}^{M}, w_c, h_c, b_c) \quad (1)$$

where, C represents the function to generate a collage image-label pair (x_i^c, y_i^c) having width w_c and height h_c, of M images with b_c being the border width (in pixels) between images.

In the early training epochs, we only use the original images (identity augmentation, i.e., id = 1) so that the network can easily and quickly learn simple representations. We only augment the source domain images. Then, we gradually increase the probability of using the other augmentations, starting with geometric augmentations and eventually using all augmentations with specified probabilities (i.e., α, β, $\gamma > 0$). These stronger regularizations make learning more difficult for the CNN and improve its robustness. The probability weights for each type of augmentation can be considered as hyperparameters (i.e., α for G, β for D, and γ for C). The pseudo code for the augmentation scheduling process is shown in Figure 4, and Algorithm 1 Figure 5 summarizes the procedure for integrating augmentation scheduling into the proposed training loop of the framework. It is straightforward to adapt this to a standard training loop. Line graph in Figure 6 shows how the probability of each type of augmentation changes with training epochs for a specified set of hyperparameters. A few examples of data samples that have been augmented using the augmentation scheduling algorithm are presented in Figure 7.

3.3 Network architecture

The proposed framework for addressing the problem of domain shift between source and target domains is depicted in Figure 8. It consists of two subnetworks segmentation network and the discriminator network:

Segmentation Network - The segmentation network (φ), having learnable parameters w, consists of two main parts: an encoder and a decoder. The encoder is made up of a stem convolution block and four stages of feature extraction. The stem block consists of two 7x7 convolutions with a stride of 2. The subsequent four stages are composed of ConvNext blocks (Liu et al., 2022), with the number of channels in each block being $N_{ch} \in \{192, 384, 768, 1536\}$, in that order. Each block is repeated N_s times at each stage, with $N_s \in \{3, 3, 27, 3\}$.

The decoder also has four stages. The first stage uses an ASPP (Chen et al., 2018) module to extract multiscale features from the output of the encoder. The second stage is an upsampling module. In the third stage, the encoder's second stage features are concatenated with the output of the second stage of the decoder through a skip connection (Ronneberger et al., 2015; Badrinarayanan et al., 2017) and are then refined by a dense attention module (DAM) (Ilyas et al., 2021). To control the flow of useful information between the encoder and decoder, the encoder's feature maps are passed through a gating function (G), to reduce the number of feature maps and suppress low-level information, before being added *via* a skip connection. This can

Pseudo Code for Progressive augmentation

```
Transforms = {
    'Geometric' : ['Flip (L-R,U-D)', 'scale', 'translate X-Y', 'rotate', 'Shear X-Y', 'crop'],
    'Distortion' : ['Median-blur', 'gaussian-blur', 'contrast', 'brightness'],
    'Custom' : ['collage'],
    'Identity' : ['identity']
    }

def proaug(batch, w):
    ''' Progressively increasing augmentations.
    batch[X] : batch of images to be augmented
    batch[Y] : batch of labels to be augmented
    w : sequence of weights, selections are made according to these relative weights.
    '''
    Augmentations = [
                    (Transforms['Identity'],   (batch[X], batch[Y])),
                    (Transforms['Geometric'], (batch[X], batch[Y])),
                    (Transforms['Distortion'],(batch[X], batch[Y])),
                    (Transforms['Custom'],    (batch[X], batch[Y]))
                    ]

    aug_X, aug_Y =  random.choice(Augmentations, weights=w)
    return aug_X, aug_Y
```

FIGURE 4
Pseudo code of Augmentation scheduling algorithm.

be represented mathematically as, $G(x) = f_r^{1\times1}(x)$, where f is a 1x1 convolutional filter with r channels.

Discriminator Network - PatchGAN (Isola et al., 2017) is used as a fully convolutional discriminator (θ) to classify whether incoming image features are form source domain or target domain. By evaluating smaller patches of the output features rather than the full feature map as a whole allows the PatchGAN to capture fine-grained details in the original image and make more informed decisions. Our framework uses two discriminators for deep feature alignment between the source and target domain features, with one aligning the decoder features (θ^{v}, having learnable parameters v) and the other aligning the encoder features ($\theta^{v_{aux}}$, having learnable parameters v_{aux}). It was found to be more effective than using only one discriminator at the end of the decoder. Both discriminators (θ) consist of five layers having filter size of 4x4 and a stride of 2, with the number of channels in each layer being {64,128,256,512,1}. Each convolutional layer is followed by instance normalization and a LeakyReLU activation with a negative slope of 0.2.

3.4 Learning objective

Given the augmented source domain labelled pair (x_i^c, y_i^c) the segmentation network (φ^w) predicts a K-dimensional soft segmentation map $p_i = \varphi^w(x_i^c)$, where $p_i \in \mathbb{R}^{H\times W\times K}$ and K is the number of classes present in the dataset. Here each K-dimensional (pixel-wise) vector is a probability distribution over classes. The segmentation network is trained by minimizing the following cross-entropy loss between the ground truth (y_i^c) and the predicted probability map (p_i), given by equation 2.

$$\mathcal{L}_{seg}(x_i^c, y_i^c) = -\frac{1}{N}\sum_{i=0}^{N} \langle\, y_i^c \,.\, \log(\varphi^w(x_i^c))\,\rangle \tag{2}$$

Algorithm 1: Training loop for Bean-Weed Recognizer with progressive augmentation.

Input: Source Dataset → $\{(x_i^s, y_i^s)\}_{i=1}^{N_s}$
Input: Target Dataset → $\{(x_i^t, y_i^t)\}_{i=1}^{M_t}$
Input: total training steps → T; current training step → t
for t = 0 to T-1 do:
  $\mu = \left[\min(\alpha, \frac{\alpha t}{\lambda_{aug}}), \min(\beta, \frac{\beta t}{2\lambda_{aug}}), \min(\gamma, \frac{\gamma t}{3\lambda_{aug}})\right]$
  $id = 1 - \max(\mu)$
  $\omega = [\mathrm{id}, \mu]$
  for i=0 to $min(N_s, M_t)$ **do:**
    $(x_i^c, y_i^c) = proaug((x_i^s, y_i^s), \omega)$
    $train \rightarrow \varphi^w(\{(x_i^c, y_i^c), x_i^t\})$ with $\mathcal{L}_{total}$; freeze → $\{\theta^v, \theta^{v_{aux}}\}$.
    $train \rightarrow \theta^v(\{x_i^c, x_i^t\}) + \theta^{v_{aux}}(\{x_i^c, x_i^t\})$ with $\mathcal{L}_d$; freeze → $\{\varphi^w\}$
  end for
end for

FIGURE 5
Training algorithm for proposed framework.

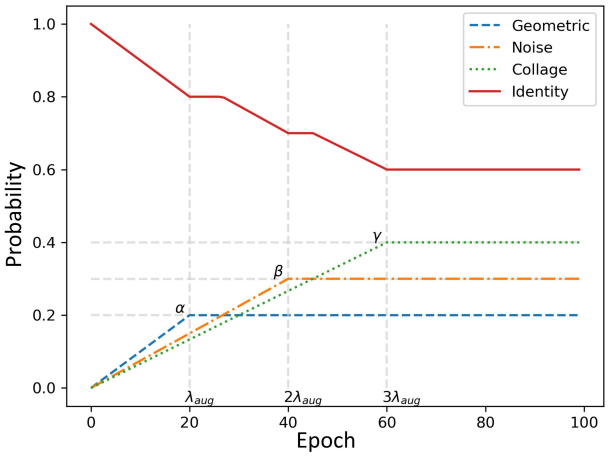


FIGURE 6
Line graph representing the changes in probabilities for each type of augmentation with training epochs for a specified set of hyperparameters, i.e., α=0.2, β=0.3, γ=0.4 and λ_{aug}=20.

For target domain samples (x_i^t) as annotation (y_i^t) are not available, hence these samples can't be used to learn the parameters (w) in same way as source domain samples can be used. So, following [28] here we use entropy minimization approach to maximize prediction certainty (lowering surprise) on target domain samples. Given a target domain input (x_i^t) we generate and entropy map (e_i), where $e_i \in \mathbb{R}^{H\times W}$ shows independent pixel-wise entropies of summation of network's predictions p_i (on target domain), normalized between [0,1] range. An example of entropy map is shown in Figure 8 and mathematically expressed by equation 3.

$$e_i = -\frac{1}{\log(K)}\sum_{k=0}^{K}\varphi^w(x_i^t).\log(\varphi^w(x_i^t)) \tag{3}$$

However, minimizing entropy directly is ineffective in low entropy regions (Yang and Soatto, 2020). So, we utilize robust entropy minimization, modified *via* carbonnier penalty function which penalizes high entropy predictions more than low entropy predictions when η > 0.5. Utilizing this modified entropy loss (L_{ent}) we update the network's parameters by equation 4.

$$\mathcal{L}_{ent}(x_i^t) = (\frac{1}{N}\sum_{i=0}^{N}e_i^2 + 0.0001^2)\eta \tag{4}$$

Given the class probability distributions generated from the features output by third stage of encoder and final stage of decoder, represented as $p_{i_{aux}}$ and p_i respectively. These distributions are then passed on to their corresponding discriminators, denoted as $\theta^{v_{aux}}$ and θ^v respectively. The goal of these discriminators is to produce domain classification outputs, with a value of 1 indicating the source domain and 0 indicating the target domain. Both discriminators are trained using the cross-entropy loss (L_{ce}). The overall objective for the final discriminator can be expressed as equation 5.

$$\mathcal{L}_d = \mathcal{L}_{ce}(\theta^v(x_i^c),\ 1) + \mathcal{L}_{ce}(\theta^v(x_i^t),\ 0) \tag{5}$$

Similarly, an equation can be written for the auxiliary discriminator ($\mathcal{L}_{d_{aux}}$), resulting in the total discriminator loss.

$$\mathcal{L}_D = \mathcal{L}_d + \lambda_{aux}\mathcal{L}_{d_{aux}} \tag{6}$$

Now, the adversarial objective for training segmentation network can be written as,

$$\mathcal{L}_{adv} = \mathcal{L}_{ce}(\theta^v(x_i^t),\ 1) \tag{7}$$

Both the segmentation and discriminator networks are jointly trained in each iteration. During training, the supervised segmentation loss for source domain samples and unsupervised entropy loss for target domain samples are jointly optimized. The adversarial loss trains the segmentation network to deceive the discriminator by maximizing the probability of target predictions being considered as source predictions. This is achieved by minimizing the cross-entropy loss between the discriminator's predictions for target images and the label of the source domain, which is 1. Therefore, the total loss becomes,

$$\mathcal{L}_{total} = \mathcal{L}_{seg} + \lambda_{ent}\mathcal{L}_{ent} + \lambda_{adv}\mathcal{L}_{adv} \tag{8}$$

In the few-shot SDA scenario, where we have j labelled images from the target domain, which are used to fine-tune our model. In addition to the entropy minimization loss described in equation 4,

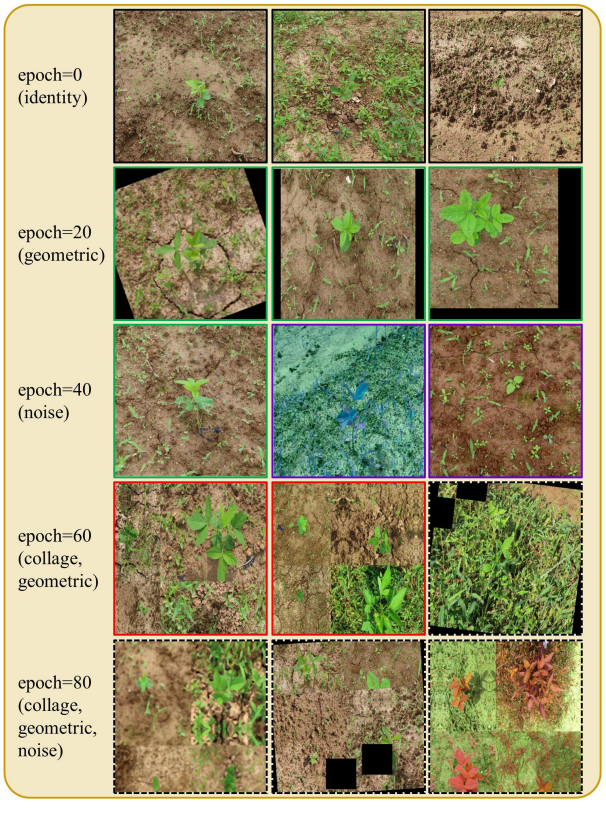


FIGURE 7
Augmentation scheduling in action: images are augmented using various combinations of augmentations at different training epochs. Black boxes depict identity augmentation, green boxes depict geometric (G) augmentation, purple boxes depict noise (D) augmentation, red boxes depict the collage (C) augmentation, and the dashed black boxes represent the application of all augmentations simultaneously.

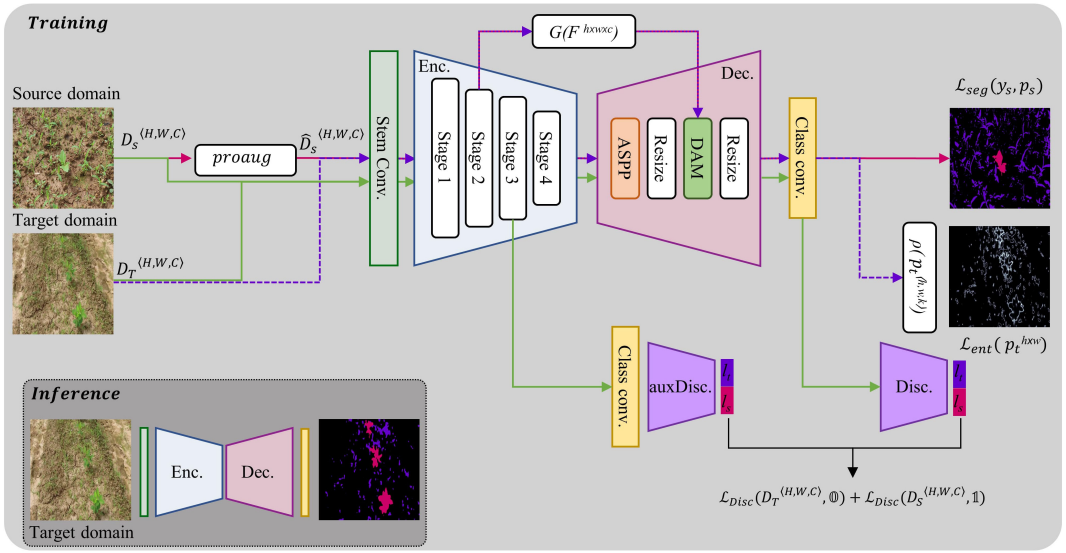


FIGURE 8
Architecture of proposed framework for UDA in crop-weed segmentation. D_S and $\hat{D}_s$ represent the source domain and augmented source domain datasets respectively. During encoder-decoder training, the pink arrows depict the flow of forward and backward gradients for input from the source domain, while the purple arrows represent input from the target domain. The discriminators are kept frozen during this training step. The green arrows show the flow while the discriminators are being trained. At this stage, the encoder-decoder network is kept frozen.

we also incorporate a cross-entropy loss similar to equation 2 for these j examples. Let's denote these labeled examples from the target domain as $D_t = \{(x_i^t, y_i^t)\}_{i=1}^{M_t}$ where i ranges from 1 to j. The additional cross-entropy loss for these samples can be expressed as:

$$\hat{\mathcal{L}}_{seg}(x_i^t, y_i^t) = -\frac{1}{j}\sum_{i=0}^{j} \langle\, y_i^t \,.\, \log(\varphi^w(x_i^t))\,\rangle \tag{9}$$

Therefore, in the case of j-shot SDA, the total loss would be updated to:

$$\mathcal{L}_{total} = \mathcal{L}_{seg} + \lambda_{seg}\hat{\mathcal{L}}_{seg} + \lambda_{ent}\mathcal{L}_{ent} + \lambda_{adv}\mathcal{L}_{adv} \tag{10}$$

where $\hat{\mathcal{L}}_{seg}$ corresponds to the supervised segmentation loss for the j labeled target domain samples, and λ_{seg} is a weight hyperparameter to balance this new term. The model is then jointly optimized for the supervised segmentation loss on both source domain and j labeled target domain samples, unsupervised entropy loss for the remaining unlabeled target domain samples, and adversarial loss.

In this way, we effectively use the limited labeled data available in the target domain to guide the model's adaptation process, while still leveraging the entropy minimization approach for the unlabeled target domain data.

3.5 Implementation details

In our implementation we used the PyTorch toolbox and a single NVIDIA RTX-3090 GPU, which has 24GB of memory. The source dataset, which contains a large number of images, was split into a 80% train-validation set and a 20% test set. The target datasets were split into a 70% training set (used only in the case of supervised training for comparison) and a 30% test set.

For training the segmentation network, we employed the SGD optimizer with a weight decay of $5x10^{-4}$. For training the discriminators, we used the Adam optimizer with a momentum value of 0.9 and 0.99. We used a cosine decay policy for the segmentation network, with a learning rate of 0.001 and warm start for the first 1000 iterations. For the discriminators, we used a polynomial decay policy with an initial learning rate of 10^{-4}. A detailed list of the hyperparameter settings for the augmentation scheduling and loss function weights can be found in Table 2.

4 Results and discussion

The performance of the proposed method for crop-weed recognition in bean fields was evaluated using the same field data distribution and source and target data splits described in Section 2. To thoroughly evaluate the proposed method, we employed widely used semantic segmentation frameworks, including DeepLab-v3+ and PSPNet (Zhao et al., 2017), with ResNet-101 (He et al., 2016), Xception-71 (Chollet, 2017), and ConvNext-L backbones as baselines. The results of our proposed method were compared with these baselines under the same operating conditions.

TABLE 2 Hyperparameter settings for proposed framework.

Hyperparameter	Value	Hyperparameter	Value
α	0.3	η	2.0
β	0.3	λ_{aux}	0.4
γ	0.3	λ_{ent}	0.5
λ_{aug}	20	λ_{adv}	0.4
λ_{seg}	2		

Firstly, we compared the performance of the proposed framework with traditional segmentation models and other recent unsupervised domain adaptation (UDA) methods. The results indicated that our proposed method performed competitively with these models. Furthermore, we demonstrated how the use of augmentation scheduling further improved the performance of our network. We also conducted ablation experiments to highlight the improvement in results achieved by using augmentation scheduling in comparison to vanilla augmentation.

Lastly, we compared the results of our proposed UDA method under both few-shot self-supervised domain adaptation (SDA) and fully supervised settings. The results showed that our proposed method performed well under both settings and yielded promising results. We evaluate the effectiveness of the proposed framework as well as compare it with other networks utilizing the Intersection-over-Union (IoU) metric, defined by equation 11.

$$mIoU = \frac{1}{N}\sum_{i=0}^{N}\left|\frac{y_i \cap p_i}{y_i \cup p_i}\right| \tag{11}$$

where y_i and p_i represent the ground-truth and predicted segments, respectively.

4.1 Source training only

In the first experiment, we trained semantic segmentation architectures in a simple supervised fashion on the source field (F_S) dataset and compared their performance. In this experiment, all models were trained on the source field dataset and results are reported on its test split (S→S), as shown in Table 3. PSPNet showed the worst performance among all other models when using the same backbone (ResNet-101), while DeepLab-v3+ with Xception-71 backbone performed better than PSPNet. Additionally, integrating the proposed modified decoder into the best-performing model (DeepLab-v3+ with ResNet-101) further boosted performance. It is worth noting that no data augmentation was used in these experiments.

Under the source training only (STO) setting, we also tested the segmentation performance of only source-trained models on other target domain fields (i.e., F_A, F_B, F_C, F_D). The results are reported in Table 3 under columns S→T, where T∈{A, B, C, D}. It can be seen from the table that, even though using better segmentation architectures resulted in considerably better performance on the F_S dataset, the results on the target domain fields did not improve and even got worse in some cases (e.g., the mIOU of field *A* and *C* decreased when using DeepLab-v3+ (ResNet-101) and proposed decoder). This demonstrates the need for unsupervised domain adaptation (UDA) approaches in the field of precision agriculture.

4.2 Unsupervised domain adaptation

In our unsupervised domain adaptation experiments, we used the same data pairs as in previous experiments. We applied the augmentation scheduling algorithm with the hyperparameter values listed in Table 2. The results of these experiments are shown in Tables 3, 4, with and without augmentation scheduling. Overall, we observed a significant improvement in the mIOU score for bean-weed recognition compared to STO methods (as seen in Table 3's top four rows). Our proposed deep feature alignment method without augmentation scheduling performed better on average than previous STO and UDA methods. As shown in Figure 9, using proposed deep feature alignment method resulted in a noticeable improvement in performance compared to using only STO. Additionally, incorporating augmentation scheduling further increased the performance of all models. Specifically, our proposed segmentation model that uses both deep feature alignment and augmentation scheduling outperformed previous best STO models by 8% and previous best UDA methods by 7%. The performance gap was even greater on target fields FA and FD, with improvements of 5.42% and 8.1% respectively.

4.3 Few-shot supervised domin adaptation

In this section, we compared our approach with other conventional few-shot SDA and fully supervised methods. The results are summarized in Table 4. All experiments were conducted under the same conditions. For the fully supervised training, all models were trained using training splits of both the target and source dataset as described in subsection 3.5 (Implementation Details). Under these conditions, our proposed segmentation network showed an improvement of 3% in the mIOU score compared to the DeepLabv3+ model, indicating its superior feature extraction ability. For the few-shot SDA experiments, the model's parameters were fine-tuned using a small number of labeled samples from the target domain. As shown in Table 5, using only one labeled sample (1-shot), our model achieved an accuracy that was almost similar to that of the fully supervised model (80.53% vs 83.6%). Additionally, our proposed method consistently outperformed other SDA methods throughout the few-shot experiments. As seen in Table 5, our method exceeded the best-performing few-shot SDA methods by 2.5% (0-shot), 3.0% (1-shot), and 2.2% (3-shot) for bean-weed recognition. Figure 10 compares the visualization results, demonstrating that our method showed significant improvements in recognizing crops and weeds.

4.4 Vanilla vs. scheduled augmentation

In these experiments, we verify the superior performance of the proposed augmentation scheduling over vanilla augmentation, and the results are summarized in Tables 4, 6. For these experiments we use proposed framework under UDA (0-shot SDA) settings. We experimented with different augmentation probabilities and found that augmenting 30% of all samples during each epoch produced the best results. Starting from the baseline (no augmentations), we first performed random geometric augmentations (G) and observed performance improvement. Then, we performed noise (D) and collage (C) augmentations one by one to see further improvements. A significant increase in performance, 55.36% (baseline) to 71.28%, can be seen when using collage augmentation (C), indicating that

TABLE 3 A comparison of the experimental results on a crop-weed segmentation dataset between traditional semantic segmentation and UDA methods with the use of Vanilla Augmentation.

Backbone	Framework	Method	mIOU (%)					
			S→S	S→A	S→B	S→C	S→D	Average
ResNet-101	PSPNet	–	74.05	66.34	56.83	38.62	52.88	53.66
Xception-71	DeepLab V3+	–	78.7	67.63	59.19	43.05	47.68	56.13
ResNet-101	DeepLab V3+	–	79.57	68.9	59.34	40.43	54.34	57.75
ResNet-101	Proposed	–	80.02	68.53	58.88	41.35	53.7	57.36
ResNet-101	DeepLab V3+	Vu et al., (2019)	**80.95**	63.66	55.62	52.31	47.62	54.80
ResNet-101	DeepLab V3+	Tsai et al. (2018)	75.94	75.57	65.23	45.65	**66.7**	63.28
ResNet-101	DeepLab V3+	Proposed	79.55	74.1	62.6	49.56	64.48	62.18
ConvNext-L	Proposed	Proposed	79.93	**75.67**	**68.54**	**52.36**	63.34	**64.97**

Bold and underlined values represent best and second-best values, respectively.

TABLE 4 A comparison of the experimental results on a crop-weed segmentation dataset between traditional semantic segmentation and UDA methods with the use of Augmentation scheduling.

Backbone	Framework	Method	mIOU (%)					
			S→S	S→A	S→B	S→C	S→D	Average
ResNet-101	PSPNet	–	78.31	77.07	67.36	47.39	62.33	63.53
Xception-71	DeepLab V3+	–	82.5	78.16	62.05	48.49	63.46	63.04
ResNet-101	DeepLab V3+	–	84.81	75.75	69.85	56.88	60.69	65.79
ResNet-101	DeepLab V3+	Vu et al. (2019)	83.99	80.6	73.4	60.86	71.7	71.39
ResNet-101	DeepLab V3+	Tsai et al. (2018)	79.8	78.75	70.8	50.67	79.98	70.05
ResNet-101	DeepLab V3+	Proposed	83.97	84.2	73.52	**61.71**	74.7	73.53
ConvNext-L	Proposed	Proposed	**88.5**	**91.02**	**83.11**	60.3	**81.1**	**78.88**

Bold and underlined values represent best and second-best values, respectively.

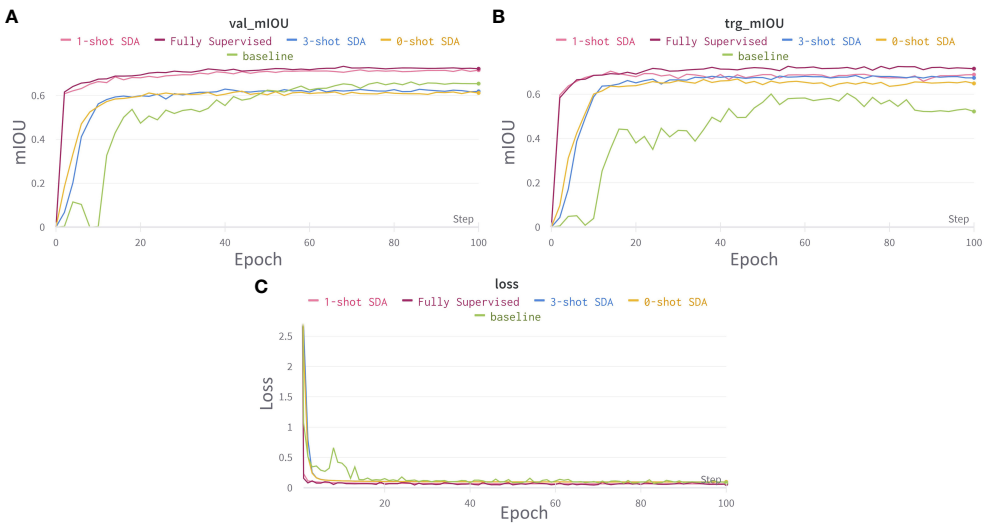


FIGURE 9
Training and loss curves for cross domain adaptation. **(A)** learning curve for mIOU score on source domains validation set. **(B)** learning curve for mIOU score averaged over all target domain test sets. **(C)** Segmentation network's loss curves. Best viewed in color.

TABLE 5 Comparison of mIOU scores for few-shot SDA models with varying values of *k* against fully supervised models.

Backbone	Framework	Method	Strategy	mIOU (%)					
				S→S	S→A	S→B	S→C	S→D	Average
ResNet-101	DeepLab V3+	Vu et al. (2019)	0-shot	83.99	80.6	73.4	60.86	71.7	71.39
ResNet-101	DeepLab V3+	Tsai et al. (2018)	(UDA)	79.8	78.75	70.8	50.67	79.98	70.05
ConvNext-L	Proposed	Proposed		88.5	91.02	83.11	60.3	81.1	78.88
ResNet-101	DeepLab V3+	Vu et al. (2019)	1-shot	86.13	88.71	83.59	58.89	75.05	76.56
ResNet-101	DeepLab V3+	Tsai et al. (2018)	(SDA)	76.49	85.01	73.02	61.59	74.9	73.63
ConvNext-L	Proposed	Proposed		88.7	92.53	84.25	62.5	80.87	80.53
ResNet-101	DeepLab V3+	Vu et al. (2019)	3-shot	88.57	89.66	84.17	60.69	78.62	78.28
ResNet-101	DeepLab V3+	Tsai et al. (2018)	(SDA)	77.36	86.07	75.49	62.99	76.31	75.19
ConvNext-L	Proposed	Proposed		89.01	92.67	85.6	62.9	82.43	81.4
ResNet-101	DeepLab V3+	–	Fully	90.56	93.25	83.5	61.56	83.42	80.43
ConvNext-L	Proposed	–	supervised	**91.62**	**95.24**	**85.48**	**68.67**	**85.39**	**83.69**

Bold and best results. Underlined values represent best results in their respective settings.

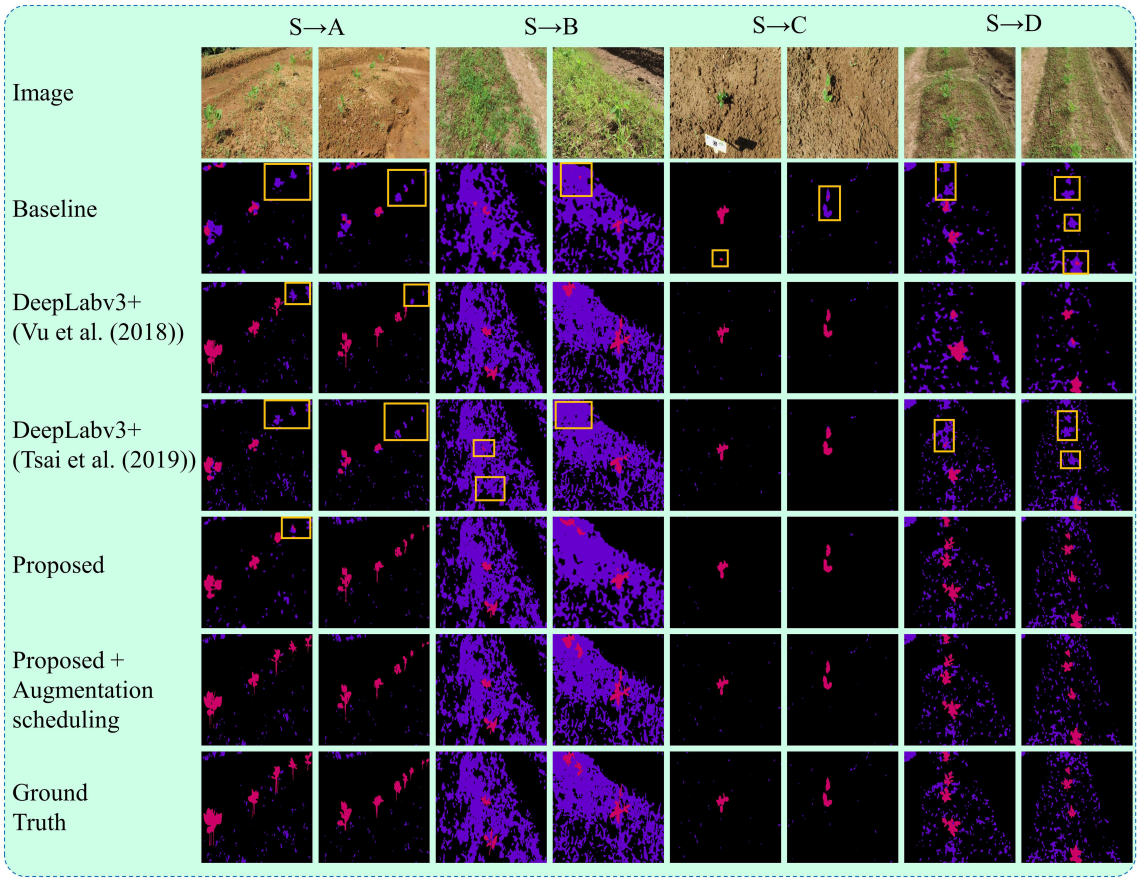


FIGURE 10
Segmentation results on datasets from the target domain under UDA setting. The results include the baseline method, DeepLabv3+ with the UDA algorithms from Vu et al., (2019) and Tsai et al. (2018), and the proposed method with and without augmentation scheduling. The ground truths are also displayed for comparison. Each target field **(A–D)** has two columns, with each column representing a different testing field with varying farm environments such as weed density, seeding bed types, plant sizes, and camera viewing angles. Boxes highlight the crops being misclassified as weeds.

TABLE 6 Effect of augmentation scheduling on performance of proposed UDA framework.

Strategy	Augmentation	mIOU (%)					
		S→S	S→A	S→B	S→C	S→D	Average
Baseline	–	73.9	68.36	56.86	42.33	53.91	55.36
Vanilla (0.3)	Geometric (G)	74.37	82.3	70.59	<u>55.98</u>	72.67	70.39
	Noise/Dist. (D)	77.59	78.49	<u>74.63</u>	55.83	73.42	70.59
	Collage (C)	80.35	<u>84.7</u>	71.33	53.91	<u>75.19</u>	<u>71.28</u>
	G+D	<u>81.03</u>	74.9	66.14	53.55	65.67	65.06
	G+D+C	79.93	75.67	68.54	52.36	63.34	64.97
Progressive (0.3)	G+D	82.59	83.57	75.85	57.75	78.36	73.88
	G+D+C	**88.5**	**91.02**	**83.11**	**60.3**	**81.1**	**78.88**

Bold and underlined values represent best and second-best values, respectively.

the collage augmentation improves the network's generalization on other domains as well. Next, we combined these augmentations at a constant probability (0.3) throughout the training process. It can be seen from Table 2 that performing all augmentations in combination considerably improved the framework's performance compared to the baseline.

However, when using all augmentations at once throughout the training process (i.e., G+D+C), the network's performance drops as compared to when only using G+D. We believe this is because the augmentations are quite strong from the start of training, making it difficult for the network to learn important distinguishing features. To overcome this, we deployed the proposed augmentation scheduling strategy, which fully activates each augmentation after a certain number of epochs (set by the user as a hyperparameter), so that the network can easily and quickly learn simple representations at the start of training. At the end of training, when all augmentations are fully activated, these stronger regularizations make learning more difficult for the CNN and improve its robustness.

As can be seen in Table 6, even without using collage augmentation, the augmentation scheduling algorithm improves the average mIOU by almost 9% compared to vanilla G+D+C. When using all three types of augmentation with progressive strategy, the results improvement is almost 14% as compared to the vanilla augmentation strategy and about a 22% increase when using no augmentation at all.

4.5 Training and loss curves across domains

In Figure 9, the graphs illustrate the training loss (source domain only) and accuracy curves for the proposed domain adaptation for the source domain and average of all the target domains. The system successfully adapted from one domain to the other and was able to effectively recognize both crops and weeds across various seeding bed systems.

5 Visual analysis

The qualitative results of the proposed method are illustrated in Figure 10. The figures present some examples of the system's qualitative performance on the testing dataset from the target and source domains. The system is capable of identifying crops and weeds effectively across different fields, even with varying densities of weeds and different seeding systems. Our approach is robust in addressing the recognition of crops (beans) and weeds, even in complex target (unseen) field environments used for domain adaptation. The underlying reason behind this performance is the utilization of deep feature alignment and augmentation scheduling algorithm which allows the system to incorporate more robust features and context information, leading to more stable and reliable segmentation results.

6 Conclusion

In this research, we presented an approach for unsupervised domain adaptation for crop-weed recognition in an unseen field environment. The main challenge in creating an automatic weed management system is the varying visual appearance of weeds based on factors such as lighting, weather, soil, and seeding bed type. We proposed to address this problem by minimizing the entropy of the network on target domain dataset and aligning the features of both domains through deep feature alignment. Our proposed framework, which is trained in an end-to-end fashion, consists of two main components: a segmentation network for feature extraction and robust entropy minimization and a discriminator network for adversarial training to generate target domain features as close as possible to the source domain. Additionally, we proposed the use of a augmentation scheduling strategy that starts with weak augmentations for quick adaptation to the source domain dataset and gradually increases to stronger augmentations for improved robustness and generalizability. We also demonstrated that the use of collage augmentation improves performance on target domains even further. Our extensive evaluation across four different fields

with various environments and plant seeding systems showed an overall performance gain of approximately 10% mIOU on average compared to the baseline. Furthermore, using just one image for fine-tuning in a few-shot SDA setting, our network achieved almost similar performance to that of a fully supervised network, i.e., 80.53% vs 83.6%. A potential direction for future research would be to explore the adaptation of the model for recognition of multiple crops and weeds.

Data availability statement

The original contributions presented in the study are publicly available. This data can be found here: https://github.com/Mr-TalhaIlyas/ARUFE.

Author contributions

TI and HK did conceptualization, investigation, methodology, validation, formal analysis. TI wrote the original draft and then revised. TI and HK supervised, reviewed and edited manuscript. HK, YJ and OW secured funding. TI and JL collected and labelled data, performed investigation and visualization. All authors discussed the results and contributed to the final manuscript. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by the Agricultural Science and Technology Development Cooperation Research Program (PJ015720) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A6A1A09031717 and NRF-2019R1A2C1011297).

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

Adhikari, S. P., Yang, H., and Kim, H. (2019). Learning semantic graphics using convolutional encoder–decoder network for autonomous weeding in paddy. *Front. Plant Sci.* 10, 1404. doi: 10.3389/fpls.2019.01404

Aschi, A., Aubert, M., Riah-Anglet, W., Nélieu, S., Dubois, C., Akpa-Vinceslas, M., et al. (2017). Introduction of Faba bean in crop rotation: Impacts on soil chemical and biological characteristics. *Appl. Soil Ecol.* 120, 219–228. doi: 10.1016/j.apsoil.2017.08.003

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 2481–2495. doi: 10.1109/TPAMI.2016.2644615

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))*, Munich, Germany. 801–818.

Chiang, H.-A., Chiang, C.-S., and Shih, C.-S. (2019). Using collaged data augmentation to train deep neural net with few data. In *Proceedings of Medical Imaging with Deep Learning*. London, England. OpenReview.net. doi: 10.1145/3400286.3418244

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251–1258).

Coleman, G., Salter, W., and Walsh, M. (2022). OpenWeedLocator (OWL): an open-source, low-cost device for fallow weed detection. *Sci. Rep.* 12, 1–12. doi: 10.1038/s41598-021-03858-9

Durmuş, H., Güneş, E. O., Kırcı, M., and Üstündağ, B. B. (2015). "The design of general purpose autonomous agricultural mobile-robot:" AGROBO," in *2015 Fourth International Conference on Agro-Geoinformatics (Agro-geoinformatics)*. Istanbul, Turkey. 49–53. IEEE.

Gao, J., French, A. P., Pound, M. P., He, Y., Pridmore, T. P., and Pieters, J. G. (2020). Deep convolutional neural networks for image-based Convolvulus sepium detection in sugar beet fields. *Plant Methods* 16, 1–12. doi: 10.1186/s13007-020-00570-z

Gebrekidan, H. (2003). Grain yield response of sorghum (Sorghum bicolor) to tied ridges and planting methods on Entisols and Vertisols of Alemaya area, Eastern Ethiopian highlands. *J. Agric. Rural Dev. Tropics Subtropics (JARTS)* 104, 113–128.

Gogoll, D., Lottes, P., Weyler, J., Petrinic, N., and Stachniss, C. (2020). "Unsupervised domain adaptation for transferring plant classification systems to new field environments, crops, and robots," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Las Vegas, NV, USA. 2636–2642. IEEE. doi: 10.1109/IROS45743.2020.9341277

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Ilyas, T., Jin, H., Siddique, M. I., Lee, S. J., Kim, H., and Chua, L. (2022). DIANA: A deep learning-based paprika plant disease and pest phenotyping system with disease severity analysis. *Front. Plant Sci.* 3862. doi: 10.3389/fpls.2022.983625

Ilyas, T., Umraiz, M., Khan, A., and Kim, H. (2021). DAM: Hierarchical adaptive feature selection using convolution encoder decoder network for strawberry segmentation. *Front. Plant Sci.* 12, 591333. doi: 10.3389/fpls.2021.591333

Iqbal, N., Manalil, S., Chauhan, B. S., and Adkins, S. W. (2019). Investigation of alternate herbicides for effective weed management in glyphosate-tolerant cotton. *Arch. Agron. Soil Sci.* 65 (13), 1885–1899. doi: 10.1080/03650340.2019.1579904

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Honolulu, HI, USA. 1125–1134. IEEE. doi: 10.1109/CVPR.2017.632

Kendler, S., Aharoni, R., Young, S., Sela, H., Kis-Papo, T., Fahima, T., et al. (2022). Detection of crop diseases using enhanced variability imagery data and convolutional neural networks. *Comput. Electron. Agric.* 193, 106732. doi: 10.1016/j.compag.2022.106732

Khan, A., Ilyas, T., Umraiz, M., Mannan, Z. I., and Kim, H. (2020). Ced-net: crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture. *Electronics* 9, 1602. doi: 10.3390/electronics9101602

Kriticos, D. J., Alexander, N. S., and Kolomeitz, S. M. (2006). Predicting the potential geographic distribution of weeds in 2080. *Weed Sci. Soc. Victoria Melbourne Aust.*, 27–34.

Lal, R. (1991). Soil structure and sustainability. *J. Sustain. Agric.* 1, 67–92. doi: 10.1300/J064v01n04_06

Lameski, P., Zdravevski, E., and Kulakov, A. (2018). Review of Automated Weed Control Approaches: An Environmental Impact Perspective. In: S. Kalajdziski and N. Ackovska (eds) ICT Innovations 2018. Engineering and Life Sciences. ICT 2018. Communications in Computer and Information Science. (Cham: Springer), vol 940. doi: 10.1007/978-3-030-00825-3_12

Li, L., Li, J., Lv, C., Yuan, Y., and Zhao, B. (2021). Maize residue segmentation using Siamese domain transfer network. *Comput. Electron. Agric.* 187, 106261. doi: 10.1016/j.compag.2021.106261

Li, J., and Tang, L. (2018). Crop recognition under weedy conditions based on 3D imaging for robotic weed control. *J. Field Robotics* 35, 596–611. doi: 10.1002/rob.21763

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). "A convnet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New Orleans, LA, USA. 11976–11986. IEEE. doi: 10.1109/CVPR52688.2022.01167

Ma, Y., Zhang, Z., Yang, H. L., and Yang, Z. (2021). An adaptive adversarial domain adaptation approach for corn yield prediction. *Comput. Electron. Agric.* 187, 106314. doi: 10.1016/j.compag.2021.106314

Matloob, A., Safdar, M. E., Abbas, T., Aslam, F., Khaliq, A., Tanveer, A., et al. (2020). Challenges and prospects for weed management in Pakistan: A review. *Crop Prot.* 134, 104724. doi: 10.1016/j.cropro.2019.01.030

Melander, B., Rasmussen, I. A., and Bàrberi, P. (2005). Integrating physical and cultural methods of weed control—examples from European research. *Weed Sci.* 53, 369–381. doi: 10.1614/WS-04-136R

Moazzam, S. I., Khan, U. S., Qureshi, W. S., Tiwana, M. I., Rashid, N., Hamza, A., et al. (2022). Patch-wise weed coarse segmentation mask from aerial imagery of sesame crop. *Comput. Electron. Agric.* 203, 107458. doi: 10.1016/j.compag.2022.107458

Nicolopoulou-Stamati, P., Maipas, S., Kotampasi, C., Stamatis, P., and Hens, L. (2016). Chemical pesticides and human health: the urgent need for a new concept in agriculture. *Front. Public Health* 4, 148. doi: 10.3389/fpubh.2016.00148

Pantazi, X.-E., Moshou, D., and Bravo, C. (2016). Active learning system for weed species recognition based on hyperspectral sensing. *Biosyst. Eng.* 146, 193–202. doi: 10.1016/j.biosystemseng.2016.01.014

Parra, L., Marin, J., Yousfi, S., Rincón, G., Mauri, P. V., and Lloret, J. (2020). Edge detection for weed recognition in lawns. *Comput. Electron. Agric.* 176, 105684. doi: 10.1016/j.compag.2020.105684

Patel, D., and Kumbhar, B. (2016). Weed and its management: A major threats to crop economy. *J. Pharm. Sci. Biosci. Res.* 6, 453–758.

Peng, H., Li, Z., Zhou, Z., and Shao, Y. (2022). Weed detection in paddy field using an improved RetinaNet network. *Comput. Electron. Agric.* 199, 107179. doi: 10.1016/j.compag.2022.107179

Picon, A., San-Emeterio, M. G., Bereciartua-Perez, A., Klukas, C., Eggers, T., and Navarra-Mestre, R. (2022). Deep learning-based segmentation of multiple species of weeds and corn crop using synthetic and real image datasets. *Comput. Electron. Agric.* 194, 106719. doi: 10.1016/j.compag.2022.106719

Radoglou-Grammatikis, P., Sarigiannidis, P., Lagkas, T., and Moscholios, I. (2020). A compilation of UAV applications for precision agriculture. *Comput. Networks* 172, 107148. doi: 10.1016/j.comnet.2020.107148

Raja, R., Slaughter, D. C., Fennimore, S. A., Nguyen, T. T., Vuong, V. L., Sinha, N., et al. (2019). Crop signalling: A novel crop recognition technique for robotic weed control. *Biosyst. Eng.* 187, 278–291. doi: 10.1016/j.biosystemseng.2019.09.011

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: N. Navab, J. Hornegger, W. Wells and A. Frangi (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science. (Cham: Springer), vol 9351. doi: 10.1007/978-3-319-24574-4_28

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., and Mccool, C. (2016). Deepfruits: A fruit detection system using deep neural networks. *sensors* 16, 1222. doi: 10.3390/s16081222

Sarvini, T., Sneha, T., Gs, S. G., Sushmitha, S., and Kumaraswamy, R. (2019). "Performance comparison of weed detection algorithms," in *2019 International Conference on Communication and Signal Processing (ICCSP)*. Chennai, India. IEEE. 0843–0847. doi: 10.1109/ICCSP.2019.8698094

Seelan, S. K., Laguette, S., Casady, G. M., and Seielstad, G. A. (2003). Remote sensing applications for precision agriculture: A learning community approach. *Remote Sens. Environ.* 88, 157–169. doi: 10.1016/j.rse.2003.04.007

Tavakoli, H., Alirezazadeh, P., Hedayatipour, A., Nasib, A. B., and Landwehr, N. (2021). Leaf image-based classification of some common bean cultivars using discriminative convolutional neural networks. *Comput. Electron. Agric.* 181, 105935. doi: 10.1016/j.compag.2020.105935

Tranheden, W., Olsson, V., Pinto, J., and Svensson, L. (2021). "Dacs: Domain adaptation *via* cross-domain mixed sampling," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, HI, USA. IEEE. 1379–1389. doi: 10.1109/WACV48630.2021.00142

Tsai, Y.-H., Hung, W.-C., Schulter, S., Sohn, K., Yang, M.-H., and Chandraker, M. (2018). "Learning to adapt structured output space for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Salt Lake City, UT, USA. 7472–7481. IEEE.

Tu, M., Hurd, C., and Randall, J. M. (2001). *Weed control methods handbook: tools & techniques for use in natural areas. All U.S. Government Documents (Utah Regional Depository)*. Paper 533. Available at: https://digitalcommons.usu.edu/govdocs/533

Vu, T.-H., Jain, H., Bucher, M., Cord, M., and Pérez, P. (2019). "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2517–2526. doi: 10.1109/CVPR.2019.00262

Yang, Y., and Soatto, S. (2020). "Fda: Fourier domain adaptation for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Seattle, WA, USA. IEEE. 4085–4095. doi: 10.1109/CVPR42600.2020.00414

Zhang, D., and Khoreva, A. (2019). Progressive Augmentation of GANs. In *33rd Conference on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2881–2890).

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 17 August 2023
DOI 10.3389/fpls.2023.1200501

Check for updates

OPEN ACCESS

EDITED BY
Changcai Yang,
Fujian Agriculture and Forestry University, China

REVIEWED BY
Karansher Singh Sandhu,
Bayer Crop Science, United States
Haikuan Feng,
Beijing Research Center for Information Technology in Agriculture, China

*CORRESPONDENCE
Aiwu Zhang
zhangaiwu@cnu.edu.cn

RECEIVED 05 April 2023
ACCEPTED 18 July 2023
PUBLISHED 17 August 2023

CITATION
Zhang J, Zhang A, Liu Z, He W and Yang S (2023) Multi-index fuzzy comprehensive evaluation model with information entropy of alfalfa salt tolerance based on LiDAR data and hyperspectral image data. *Front. Plant Sci.* 14:1200501. doi: 10.3389/fpls.2023.1200501

COPYRIGHT
© 2023 Zhang, Zhang, Liu, He 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.

Multi-index fuzzy comprehensive evaluation model with information entropy of alfalfa salt tolerance based on LiDAR data and hyperspectral image data

Jiaxin Zhang[1,2,3], Aiwu Zhang[1,2,3]*, Zixuan Liu[1,2,3], Wanting He[1,2,3] and Shengyuan Yang[1,2,3]

[1]Key Laboratory of 3D Information Acquisition and Application, Ministry of Education, Capital Normal University, Beijing, China, [2]Engineering Research Center of Spatial Information Technology, Ministry of Education, Capital Normal University, Beijing, China, [3]Center for Geographic Environment Research and Education, College of Resource Environment and Tourism, Capital Normal University, Beijing, China

Rapid, non-destructive and automated salt tolerance evaluation is particularly important for screening salt-tolerant germplasm of alfalfa. Traditional evaluation of salt tolerance is mostly based on phenotypic traits obtained by some broken ways, which is time-consuming and difficult to meet the needs of large-scale breeding screening. Therefore, this paper proposed a non-contact and non-destructive multi-index fuzzy comprehensive evaluation model for evaluating the salt tolerance of alfalfa from Light Detection and Ranging data (LiDAR) and HyperSpectral Image data (HSI). Firstly, the structural traits related to growth status were extracted from the LiDAR data of alfalfa, and the spectral traits representing the physical and chemical characteristics were extracted from HSI data. In this paper, these phenotypic traits obtained automatically by computation were called Computing Phenotypic Traits (CPT). Subsequently, the multi-index fuzzy evaluation system of alfalfa salt tolerance was constructed by CPT, and according to the fuzzy mathematics theory, a multi-index Fuzzy Comprehensive Evaluation model with information Entropy of alfalfa salt tolerance (FCE-E) was proposed, which comprehensively evaluated the salt tolerance of alfalfa from the aspects of growth structure, physiology and biochemistry. Finally, comparative experiments showed that: (1) The multi-index FCE-E model based on the CPT was proposed in this paper, which could find more salt-sensitive information than the evaluation method based on the measured Typical Phenotypic Traits (TPT) such as fresh weight, dry weight, water content and chlorophyll. The two evaluation results had 66.67% consistent results, indicating that the multi-index FCE-E model integrates more information about alfalfa and more comprehensive evaluation. (2) On the basis of the CPT, the results of the multi-index FCE-E method were basically consistent with those of Principal Component Analysis (PCA), indicating that the multi-index FCE-E model could accurately evaluate the salt tolerance of

alfalfa. Three highly salt-tolerant alfalfa varieties and two highly salt-susceptible alfalfa varieties were screened by the multi-index FCE-E method. The multi-index FCE-E method provides a new method for non-contact non-destructive evaluation of salt tolerance of alfalfa.

KEYWORDS

alfalfa, hyperspectral image, LiDAR data, phenotypic traits, the evaluation of salt tolerance, fuzzy

1 Introduction

Alfalfa is an important supplementary crop and plays an important role in animal husbandry. However, saline-alkali land has a serious impact on the growth and development of alfalfa (Munns and Tester, 2008). Therefore, it is of great practical significance to accurately evaluate the salt tolerance of alfalfa (Zhang et al., 2013; Reddy et al., 2022). However, it is a complex task to accurately evaluate the salt tolerance of alfalfa. Traditional phenotypic survey methods are limited by the number of samples and time costs, and cannot comprehensively evaluate the structural, physical and chemical indicators of alfalfa (Munns and Tester, 2008; Hanin et al., 2016; Ismail and Horie, 2017). Therefore, it is of great significance to comprehensively evaluate the salt tolerance of alfalfa by using modern plant phenotypic techniques to understand the salt tolerance mechanism of alfalfa, improve alfalfa varieties and increase agricultural production in saline-alkali areas (Hu and Schmidhalter, 2023).

Phenotypic traits of plant salt tolerance refer to all traits that reflect the physical, physiological and biochemical properties of plant salt tolerance influenced by genes or environment. Studies have been carried out in China and abroad to evaluate the salt tolerance of alfalfa using the phenotypic traits of indicator plant individuals or populations (Houle et al., 2010; Chunjiang, 2019; Al-Tamimi et al., 2022; Singh and Bhutia, 2022). However, considering the wide planting range of alfalfa, there are many drawbacks in the phenotype based on manual measurement. For example, large-scale monitoring will make the measurement cycle long and the timeliness of data low; point monitoring will make the data lack of overall representativeness, and it is difficult to meet the needs of regional breeding work. In addition, in breeding experiments, laboratory chemical analysis is often used to obtain physiological and biochemical parameters of alfalfa, which greatly increases the experimental cost. In order to promote the study of modern plant phenotypes, this paper proposes the concept of computing phenotypic traits. Computing phenotypic traits refer to the digital phenotypic traits that characterize the growth and development, physiology and biochemistry of plant individuals or populations extracted by comprehensive computer technology, image analysis technology and other modern science and technology. With the development of non-contact information collection methods such as computer vision, HSI technology, and 3D laser technology, more and more plant CPT have been extracted, which provides technical support for real-time and non-destructive evaluation of alfalfa salt tolerance (Tmušić et al., 2020; Brook et al., 2021; Singh et al., 2021; Li et al., 2022).

When alfalfa is subjected to salt stress, it first affects the physical and chemical parameters. The spectral absorption and reflection characteristics of plants can be used to characterize their physiological and biochemical characteristics (Tucker, 1977). Spectral indices can highlight phenotypic traits such as plant biomass, leaf water content, pigment content and salt stress index through band combination, and reduce the negative impact of spectral redundancy on trait extraction (El-Hendawy et al., 2022). The RGB and multispectral sensors have lower spectral resolution, fewer bands, and discontinuous spectral coverage, which results in a limited availability of spectral index features. HSI can describe the interaction between alfalfa physicochemical traits and the environment in more detail through its fine spectral superiority and spatial information, and shows a strong advantage in the extraction of alfalfa computational phenotypic traits (El-Hendawy et al., 2019a; Xiaofeng et al., 2020; Jin et al., 2021b). The hyperspectral narrow band vegetation index can more comprehensively characterize the content of crop stress resistance components through band combination (Post et al., 2007; Wu et al., 2008; Ullah et al., 2012; Thenkabail et al., 2013; El-Hendawy et al., 2019b), which provides an effective means for the study of alfalfa stress resistance phenotypic traits, breeding screening and implementation of precision agriculture (Hunt et al., 2013; Kasim et al., 2017). With the cumulative change of physical and chemical parameters, the growth, development and structural parameters of alfalfa were also affected and changed (Hanin et al., 2016). LiDAR technology measures the distance between the sensor and the target object by laser irradiation and is widely used in plant reconstruction and morphological structure extraction (M et al., 2021; Shaochen et al., 2023). Sun et al. used LiDAR technology to monitor the maximum canopy height, projected canopy area and plant volume of cotton plants with good results (Shangpeng et al., 2018). With the advancement of technology, the use of LiDAR point cloud data to obtain plant structural phenotypic traits has been widely recognized (Jimenez-Berni et al., 2018; Jin et al., 2021a). Secondly, LiDAR is an active remote sensing technology. Compared with spectral imaging technology, it is not affected by light and shooting angle, and can realize all-day monitoring. The true morphology of alfalfa was restored to the greatest extent by a high-precision three-dimensional point cloud, so as to extract its structural phenotypic

traits (Shaochen et al., 2023). These structural phenotypic traits are essential for evaluating the salt tolerance of alfalfa. In this paper, plant height, canopy leaf area and volume extracted from LiDAR data, and special indicators extracted from HSI data were called computing phenotypic traits.

The salt tolerance response mechanism of alfalfa is complex. The evaluation of salt tolerance is based on the overall performance of various physiological processes, and any single trait can't directly represent its salt tolerance level. It is more scientific and reasonable to use a multi-index system to comprehensively evaluate alfalfa (Shaohua et al., 2022; Sun et al., 2022). Some scholars have proposed that the method of a comprehensive evaluation of multiple indicators, such as cluster analysis, principal component analysis and the membership function value method, is the best method for screening salt-tolerant alfalfa (Hu et al., 2018; Xiangfeng et al., 2018). However, these evaluation methods are easily affected by subjective factors. Fuzzy Comprehension Evaluation (FCE) is a method based on the fuzzy transformation principle in fuzzy mathematics (Zadeh., 1965). Based on the principle of membership, the algorithm scientifically and objectively synthesizes a multi-index problem into a single-index result containing multi-index information, to realize multi-factor comprehensive evaluation in one-dimensional space (Robati and Rezaei, 2021). The basis of constructing a comprehensive evaluation model of salt tolerance by the FCE method is to establish a scientific, reasonable, comprehensive and objective evaluation index system. Xu et al. proposed a comprehensive yield evaluation index that reflected leaf area index, leaf biomass, leaf moisture content and leaf nitrogen content (Xu et al., 2021). However, the measurement of these indicators has low efficiency, serious subjectivity, large measurement error, and poor plant adaptability when repeated measurement of a single plant, which make it difficult to meet the requirements of modern agricultural production practice. In addition, these indicators focus on the estimation of yield and cannot represent the salt tolerance of alfalfa growth. Wu et al. discussed the contribution of morphological structure, physiological and biochemical phenotypic trait indicators to the screening of alfalfa germplasm. The results showed that the structural phenotypic traits associated with alfalfa growth and development and the phenotypic traits characterized by physicochemical characteristics were important parameters for evaluating the salt tolerance of alfalfa (Duan et al., 2008; Tilly et al., 2013; Tilly et al., 2015; Xinming et al., 2018; Duo et al., 2021; Penglei et al., 2021; Guiza et al., 2022). However, phenotypic traits based on manual measurement and chemical analysis are expensive and limited. Therefore, the application of remote sensing technology to the evaluation of salt tolerance can not only extract more phenotypic traits to characterize alfalfa salt tolerance more efficiently but also avoid irreversible damage to alfalfa caused by measurement in indicators.

Another focus of the FCE method is the weight distribution of the evaluation indicators. The traditional method of determining the weight mostly adopts the expert scoring method, which has strong subjective defects (Du et al., 2019). Therefore, it is necessary to improve the FCE method to more objectively reflect the salt tolerance level of alfalfa. In summary, the screening of salt-tolerant varieties is based on the comprehensive evaluation of their phenotypic traits. However, most of the current research on the salt tolerance evaluation of alfalfa relies on TPT (Roy et al., 2014; Ismail and Horie, 2017), and few studies have applied remote sensing technology to the screening of alfalfa salt tolerant varieties. Moreover, the construction of the comprehensive evaluation model of salt tolerance is mostly affected by subjective factors (Song et al., 2021). Therefore, the purpose of this study is to conduct a rapid and non-destructive comprehensive evaluation of alfalfa salt tolerance at flowering date based on HSI and LiDAR data, and to improve the FCE method to enhance the automation ability of the salt tolerance evaluation model. The specific research objectives of this study are as follows: (1) To construct a digital and non-destructive evaluation index system of alfalfa salt tolerance that characterizes the growth, development and physicochemical characteristics of alfalfa; (2) It is proven that the CPT can capture more salt sensitive information than the TPT and is more comprehensive and robust in evaluating the salt tolerance of alfalfa. (3) The FCE method was optimized to develop an automated comprehensive evaluation model for salt tolerance of alfalfa with strong portability. (4) Compared with the PCA method, the multi-index FCE-E model is more suitable for screening and breeding research. This study not only helps to enhance the understanding of the salt tolerance mechanism of alfalfa, but also provides new ideas and approaches for the evaluation of the salt tolerance of alfalfa, and provides valuable references for research and practice in related fields.

2 Materials and methods

2.1 Design of experiment

In this experiment, six alfalfa varieties of WL343HQ, Gibraltar, Gold Empress, Zhongmu No. 3, Aohan, and Cangzhou were studied at flowering date. To compare the effects of salt stress on the growth status of alfalfa, a salt stress (NaCl) group and a control (CK) group were set up on the material, and a repeat experiment was set up at the same time. In the pre-experiment, it was found that the differences between alfalfa varieties were not prominent in the salt stress treatment of 100 mmol/L; with 150 mmol/L of salt stress, there were obvious differences between alfalfa varieties. All varieties survived in two pre-experiments. Therefore, the NaCl group in this study was treated with salt stress of 125 mmol/L, and the CK group was treated with clean water.

To avoid the effect and damage caused by watering impulse on alfalfa roots, 9 plants of the same variety of alfalfa were evenly planted uniformly in a 20 cm × 20 cm porous pot, and 6 porous pots were placed in a 40 cm × 60 cm non-porous box (Figure 1). The irrigation treatment was in the non-porous box. Porous pots could enable alfalfa connections to be placed in a habitat, reducing the generation of variables. Except for different treatment methods, the other growth environments of the sample materials were consistent, which met the experimental conditions of the single variable method.

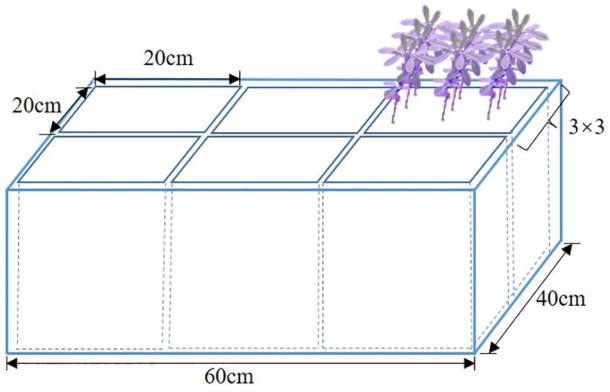


FIGURE 1
Schematic diagram of alfalfa planting.

To construct a comprehensive evaluation model of salt tolerance based on non-destructive monitoring, before measuring the measured phenotypic traits such as fresh weight and dry weight of alfalfa at flowering date, HSI data and LiDAR data were collected, and the spectral and structural phenotypic traits were analyzed and extracted.

2.2 Methods of research

To construct a salt tolerance evaluation model of alfalfa and screen the salt-tolerance alfalfa germplasm resources, it is necessary to establish a multi-index salt tolerance evaluation system for alfalfa. The experimental process is shown in Figure 2, which mainly consists of four main parts: (1) Data collection: To achieve rapid and non-destructive evaluation of alfalfa's salt tolerance, HSI data and LiDAR data of alfalfa at flowering date were collected. In addition, four

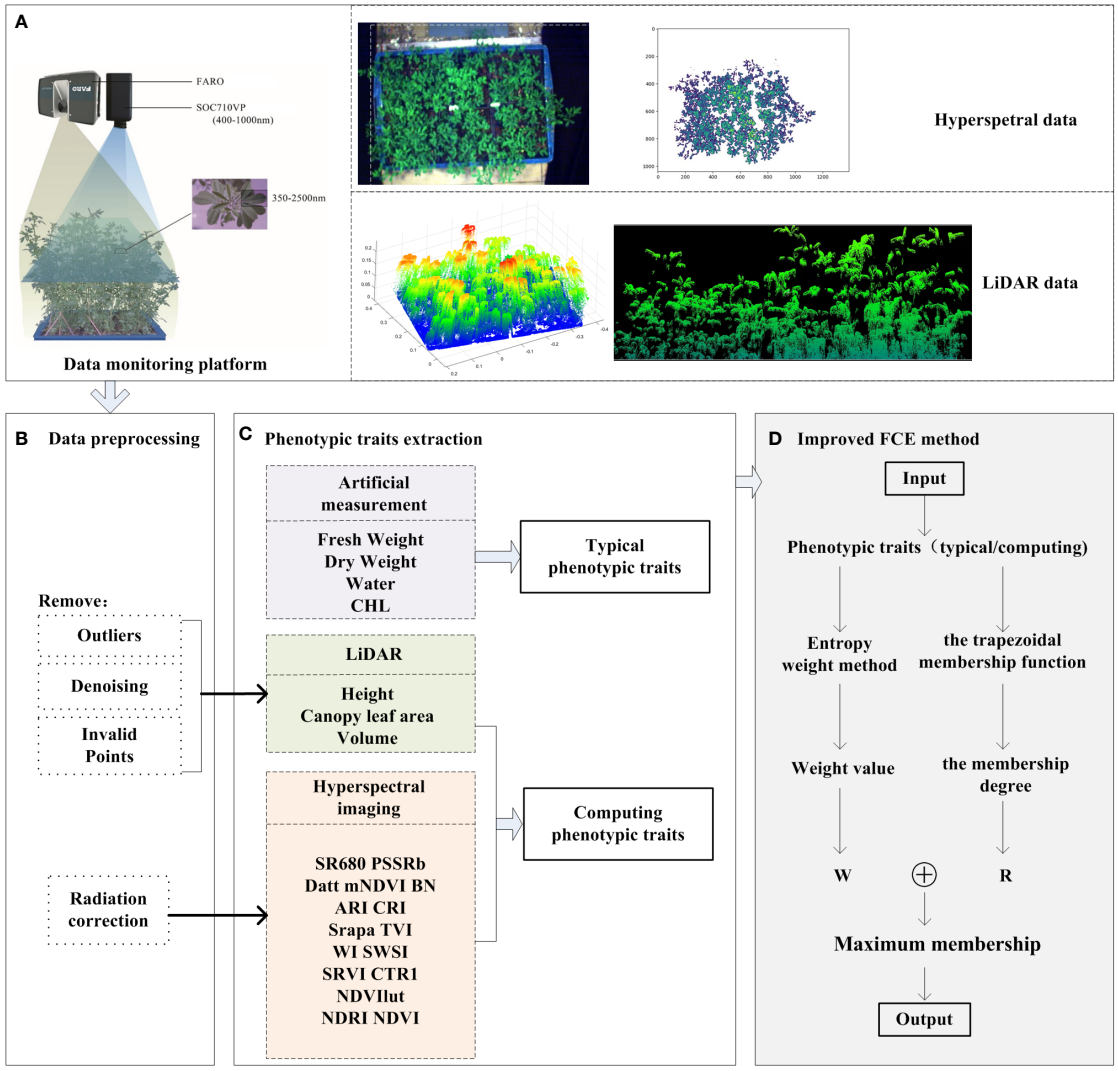


FIGURE 2
A workflow diagram of the experimental design, feature extraction, and modeling. **(A)** data collection; **(B)** data preprocessing for original data; **(C)** phenotypic traits extraction; **(D)** improved FCE method and model construction.

typical traits of alfalfa, including Fresh Weight (FW), Dry Weight (DW), Water Content (WC) and Chlorophyll (SPAD, CHL), were collected after mowing. (2) Data preprocessing for original data: In order to improve the accuracy of extracting structural phenotypic traits from the LiDAR data, the outliers, denoising and invalid points were removed from the LiDAR data, and then the structural phenotypic traits of alfalfa were extracted; The radiometric calibration of the HSI data was carried out by using the software of the hyperspectral imager, and then the spectral phenotypic traits of alfalfa were extracted (Figure 3). (3) Phenotypic traits extraction: Construct the CPT using the structural phenotypic traits extracted from the LiDAR data and the spectral phenotypic traits extracted from the HSI data. (4) Improved FCE method and model construction: Firstly, the subjectivity of the FCE method is improved by the entropy weight method and adaptive adjustment of the critical point of the fuzzy membership function. Then, on the basis of the CPT (step 3), comprehensive evaluation models for salt tolerance of alfalfa in the CK group and NaCl group were constructed by the multi-index FCE-E method, to obtain the salt tolerance rating of alfalfa and screen out the salt tolerance germplasm resources.

2.2.1 Acquisition of measured data

This experiment was based on the "Descriptors and Data Standard for Medic (Medicago Linn.) (Hongyan and Zongli, 2007)" investigated and extracted four typical phenotypic traits of alfalfa at flowering date, including FW, DW, WC, and CHL. First, the material was cut from 1cm above the ground in a non-porous box, and the alfalfa samples in each box were loaded into a mesh bag of the same specification. The FW was obtained by directly weighing each box of alfalfa plants and mesh bags using an electronic scale (± 0.01g). The DW was obtained by weighing the plant and drying it in a mesh bag at 120°C to a constant weight. The WC was the difference between FW and DW. The CHL was the average value of more than 15 measurements by the SPAD instrument.

2.2.2 Acquisition and preprocessing of hyperspectral data

In this study, the SOC710VP hyperspectral imager with a spectral range of 400-1000 nm was used to collect hyperspectral data with a spectral resolution of 4.6875 nm. HSI data were acquired by setting up a darkroom and illuminating it with a full-band lamp. At the same time of data collection, a standard gray plate as high as alfalfa was placed next to alfalfa for radiometric calibration of HSI data.

The HSI preprocessing process is shown in Figure 3A. The original image has radiation errors, so radiation calibration was performed before use. Because the HSI data was collected indoors, the influence of the atmosphere on the calibration did not need to be considered. The relationship between the calibration data of the hyperspectral imager and the actual measured radiance brightness of each band is established by the software of the instrument to achieve the purpose of radiation correction.

In order to extract the spectral phenotypic traits characterizing the physical and chemical characteristics of alfalfa, this paper first distinguishes alfalfa and non-alfalfa regions based on HSI data, and then the average value of the alfalfa imaging range was taken as the value of this trait. The experimental results have shown that the Triangular Vegetation Index (TVI, Table 1) could better distinguish between alfalfa and non-alfalfa in hyperspectral images (Figure 4). TVI is closely related to chlorophyll content. It is constructed according to the difference in light radiation energy of pigments in green, red and near-infrared radiation energy. A robust "triangle" spectral space is formed by the reflection peak of green light, the absorption valley near red light, and the red edge of the near-infrared. Vegetation and non-vegetation regions can be distinguished according to this spectral signature.

In the experiment, it was found that in the TVI image, the TVI values of alfalfa and non-alfalfa areas were quite different, while the TVI values in the alfalfa area were similar. The threshold

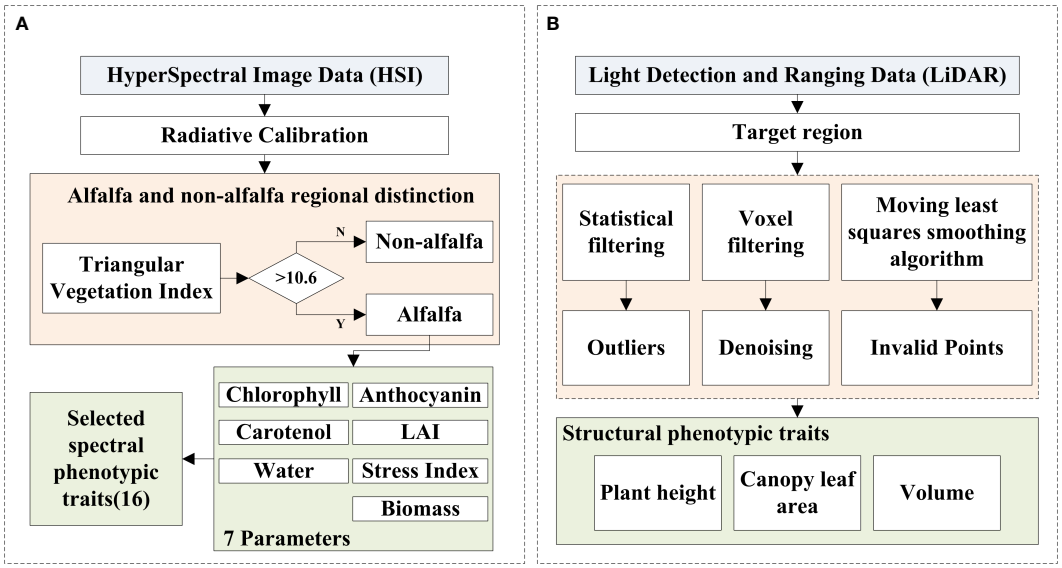


FIGURE 3
The processing flow chart of HSI data and LiDAR data. **(A)** is the HSI data, **(B)** is the LiDAR data.

TABLE 1 Selected spectral traits.

Parameters	VI	Formular	Reference
Chlorophyll	Simple Ratio Index	SR680=R800/R680	(Sims and Gamon, 2002)
	Pigment-specific Simple Ratio	PSSRb=R800/R635	(Blackburn, 2010)
	Datt Index	Datt=(R850-R710)/(R850-R680)	(Datt, 2010)
	Modified Normalized Difference Vegetation Index	mNDVI=(R750-R705)/(R750+R705-2*R445)	(Sims and Gamon, 2002)
	BN Index	BN=Log(R800/R550)	(Buschmann and Nagel, 2007)
Anthocyanin	Anth Reflectance Index	ARI=1/R550-1/R700	(Gitelson et al., 2001)
Carotenol	Carotenoid Reflectance Index	CRI=(1/R510)-(1/R550)	(Gitelson et al., 2002)
LAI	Simple Ratio Index	Srapa=R900/R680	(Aparicio et al., 2002)
	Triangular Vegetation Index	TVI=0.5*(120*(R750-R550)-200*(R670-R550))	(Broge and Leblanc, 2001)
Water	Water Index	WI=R900/R970	(Penuelas et al., 2010)
	Salinity and Water Stress Indices	SWSI=(R803-R681)/((R905+R972)**0.5)	(Hamzeh et al., 2013)
Stress Index	Simple Ratio Vegetation Index	SRVI=R830/R660	(Wang et al., 2010)
	Carter Indices	CTR=R695/R420	(Carter, 2007)
	Normalized Difference Vegetation Index	NDVIlut=(R913-R711)/(R913+R711)	(Luther and Carroll, 1999)
	Normalized Difference Red Edge Index	NDRI=(R790-R720)/(R790+R720)	(Barnes et al., 2000)
Biomass	Normalized Difference Vegetation Index	NDVI=(R800-R670)/(R800+R670)	(Tucker, 1979)

Where R * is the reflectivity of the corresponding wavelength.

segmentation method was the simplest and the most appropriate method for extracting alfalfa. Through the experimental method, 10.6 was set as the threshold of TVI to distinguish alfalfa and non-alfalfa areas. After alfalfa extraction, the average value of alfalfa imaging range after band calculation was used as the value of spectral traits to provide parameters for the multi-index salt tolerance evaluation model of this study.

2.2.3 Acquisition and preprocessing of LiDAR data

In this study, a FARO Focus M70 laser scanner with a wavelength of 1550nm was used to obtain LiDAR point cloud data, and the point cloud was preprocessed according to the process shown in Figure 3B. LiDAR uses scanning to obtain three-dimensional information such as the structure, position and shape of the target object, and inevitably produces some noise. In this experiment, the target object was cut first, and then the point cloud data of the target object was preprocessed by removing outliers, redundant points and mixed points. First of all, when the LiDAR scanned and recorded the information of the measured object, random noise would be generated, or some objects were not completely scanned. In order to make the point cloud structure characteristics of the alfalfa group clearer, this paper used the statistical filtering of the standard deviation multiple to eliminate outliers. Secondly, the LiDAR scanned the same area repeatedly during the scanning process, resulting in redundant point clouds, which require high data storage space and hardware equipment. According to the growth characteristics of the alfalfa population, this paper performed voxel filtering on the point cloud below the canopy, which reduced the number of point clouds on the basis of

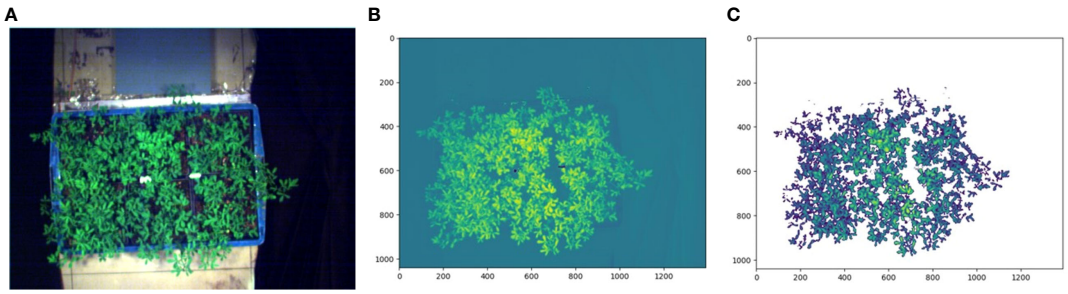


FIGURE 4
TVI Index distinguishing alfalfa and non-alfalfa. **(A)** is the RGB image synthesized by hyperspectral data, **(B)** is the TVI index image, **(C)** takes TVI = 10.6 as the threshold, the area with TVI greater than 10.6 is the alfalfa, and the area with TVI less than 10.6 is the other.

ensuring the geometric characteristics of point clouds, saved storage space and improved the efficiency of parameter extraction. Finally, in order to highlight the spatial characteristics of the alfalfa point cloud and ensure the smooth surface of leaves, aiming at the characteristics of high overlap, small leaves, and serious inter-species interleaving in the alfalfa population, this paper used the moving least squares smoothing algorithm to fit the surface by setting the radius range and projecting the mixed points to the surface to remove the mixed points. The above preprocessing work enabled the point cloud data to more accurately reflect the true information of alfalfa and provided a high-quality data basis for the subsequent extraction of structural phenotypic traits.

2.3 The construction of the multi-index fuzzy evaluation system

The effects of salt stress on alfalfa are mainly manifested in two aspects: (1) effects on the structural growth and development of alfalfa; and (2) effects on physiological and biochemical characteristics of alfalfa. Therefore, a multi-index fuzzy evaluation system was constructed by CPT combining spectral phenotypic traits based on HSI data and structural phenotypic traits based on LiDAR data.

The structural phenotypic traits related to the growth and development of alfalfa were extracted with three typical indexes: plant height, canopy leaf area and volume (Figure 3B). The change in plant height can reflect the change in alfalfa stem length, which is an important indicator of alfalfa growth; The change in canopy leaf area can reflect the speed of leaf growth. Alfalfa leaves provide nutrients for alfalfa growth through photosynthesis, which is very important for the growth and development, chlorophyll content and health status of alfalfa; Volume is the space occupancy of alfalfa growth, which is closely related to alfalfa biomass. In this paper, the bottom of alfalfa was used as the benchmark of point cloud data. Alfalfa was divided into 0.5cm grids, and the highest points in all grids were counted, and the average value was taken as the average plant height of alfalfa (Figure 5A). The sum of all grid areas was used as the canopy leaf area of the alfalfa population (Figure 5B). Alfalfa volume was calculated by constructing a convex hull (Figure 5C).

The spectral phenotypic traits were extracted by analyzing the spectral curve principle to characterize the phenotypic traits of alfalfa chlorophyll, anthocyanin, carotenol, leaf area index, water content, stress index and biomass (Figure 3A). Chlorophyll is involved in the absorption, transmission and transformation of light energy, which is an important basis for the growth of alfalfa. Water content is an important index in the evaluation of salt tolerance. Excessive salt accumulation in the soil will cause a decrease in soil water potential, making it difficult for alfalfa to absorb water, resulting in physiological drought and osmotic stress. The salt stress index is a comprehensive response to salt stress. Studies have shown that hyperspectral data can well reflect plant physiology and biochemistry information through a variety of linear or non-linear combination band operations (Saric et al., 2022). From the existing studies, a total of 16 vegetation indices were screened in this paper (Table 1, Figure 6), Among them, 5 indexes were extracted to characterize chlorophyll, 1 index was extracted to characterize anthocyanin, carotenol and biomass, 2 indexes were extracted to characterize leaf area index and leaf water content, and 4 indexes were extracted to characterize stress index. The vegetation index was calculated using the Python language GDAL library, and the alfalfa area extracted in the Section 2.2.2 was used as a mask, and the mean values of the alfalfa area of the vegetation indexes was used as the spectral phenotypic traits of this study.

2.4 The construction of fuzzy comprehensive evaluation model

Based on the multi-index fuzzy evaluation system (in section 2.3), the evaluation model of alfalfa salt tolerance was constructed by FCE-E method. This algorithm was completed on Matlab (Figure 7). The algorithm steps are as follows:

1. Input of the algorithm;

Let the evaluation index system be $X = [X_1\ X_2\ \cdots\ X_m]$, where X_m is the mth evaluation index. In this paper, the fuzzy evaluation index system for salt tolerance (in section 2.3) was used as the factor set, and the data from the factor set was used to construct a two-dimensional matrix A, where X_{nm} was the value of the mth index of the nth group of alfalfa.

$$A = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1m} \\ X_{21} & X_{22} & \cdots & X_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nm} \end{bmatrix} \quad (1)$$

Let the evaluation set be $U = [u_1\ \ u_2\ \ \cdots\ \ u_n]$, where u_n is the nth evaluation level. In this paper, the evaluation set was set $U = [\textit{Very tolerant}\quad \textit{Intermidiate}\quad \textit{Susceptible}\quad \textit{Very susceptible}]$

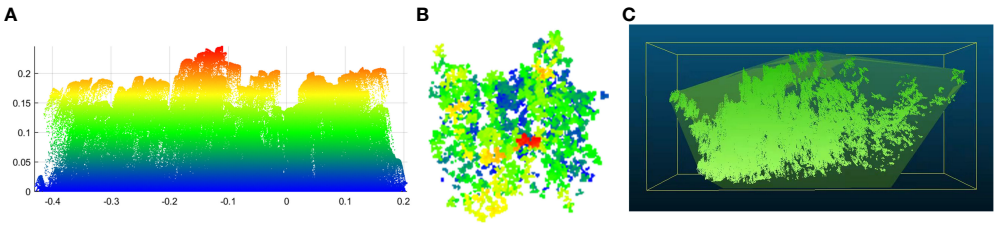


FIGURE 5
Structural phenotypic traits diagram. **(A)** is the plant height, **(B)** is the canopy leaf area, **(C)** is the volume.

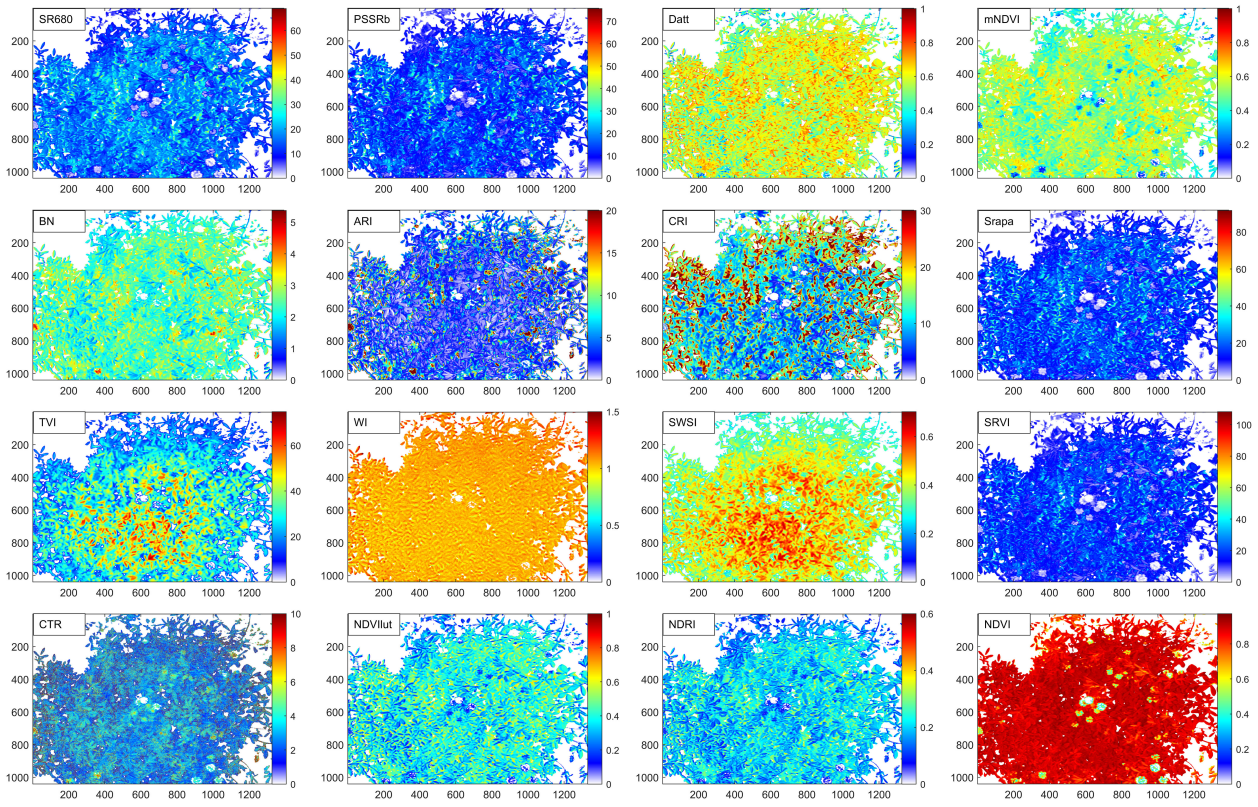


FIGURE 6
Spectral phenotypic traits diagram.

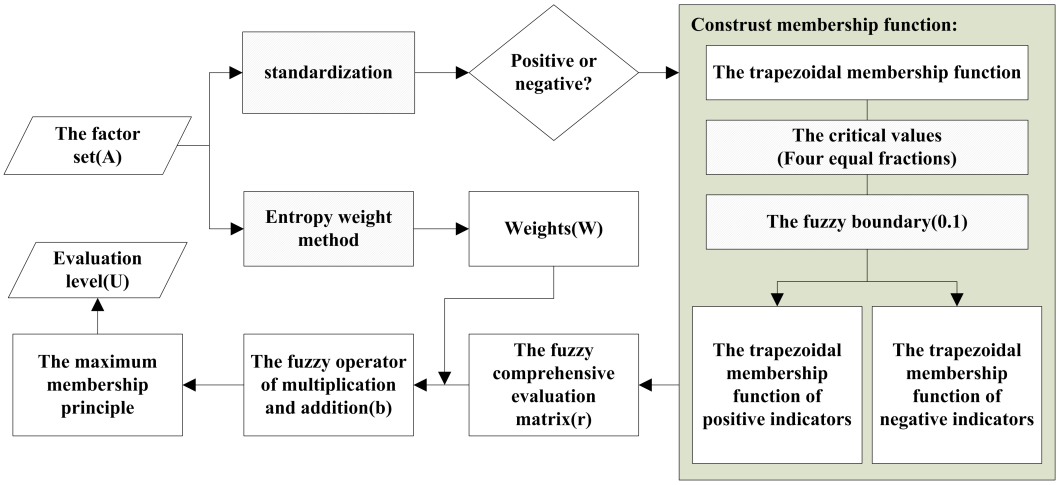


FIGURE 7
The construction of fuzzy comprehensive evaluation model.

according to the "Descriptors and Data Standard for Medic (Medicago Linn.) (Hongyan and Zongli, 2007)".

2. Decide whether the evaluation index is positive or negative;

With yield as the evaluation reference, the evaluation index was positively correlated with yield, and the index was defined as a positive index. On the contrary, the evaluation index was negatively correlated with the yield, so the evaluation index was defined as a negative index.

3. The entropy weight method is used to calculate the weight distribution of the evaluation index system;

Due to the different contributions of each evaluation index to salt tolerance evaluation, different weights should be assigned to

each factor involved in the evaluation. The entropy weight method is an objective weight method that determines the index weight according to the variation degree of each index value.

First, the index data was standardized and converted into a range between 0-1. The evaluation of salt tolerance focused on the difference in phenotypic traits among different alfalfa varieties under salt stress, and the weight of indexes was determined according to the different information contained in phenotypic traits. Therefore, the information entropy $E(X)$ was calculated to measure the amount of information, and then the weight $W(X)$ was determined according to the information entropy of each index, and finally the weight matrix $W$of the evaluation index system was constructed.

$$E(X) = -\Sigma(P(X_i) * \log_2(P(X_i))) \tag{2}$$

$$W(X) = \frac{1 - E(X)}{\Sigma(1 - E(X))} \tag{3}$$

$$\mathrm{W} = [\, w_1 \;\; w_2 \;\; \ldots \;\; w_\mathrm{m} \,] \tag{4}$$

Where, X is the index, X_i is the value of the index X of each sample, and $P(X_i)$ is the probability of each value of the index X_i. $W(X)$ is the weight of the index X.

4. Construct membership function;

Each evaluation index presents different rules for the evaluation level. This study used the most commonly used trapezoidal membership function to set membership rules, and the parameters of membership function are set by adaptive parameter adjustment method. Firstly, the positive and negative indicators were uniformly standardized to the interval [0,1], to avoid the influence of outliers and extreme values indirectly through centralization, and to be basis of adaptive parameter adjustment method. Secondly, the sample was divided into four ratings. When setting the upper and lower indices of the trapezoidal membership function, 0.25, 0.5 and 0.75 were taken as the critical values of four equal fractions. For example, there was a fuzzy evaluation of plant height at 0.24 and 0.26, so the fuzzy boundary was set as 0.1, that is, the fuzzy boundary of 0.25 was 0.2-0.3. The trapezoidal membership function of positive indicators of 'Very tolerant', 'Intermediate', 'Susceptible' and 'Very susceptible' are shown in formula 5-8 and Figure 8. x_1, x_2, x_3, x_4, x_5, x_6 are 0.2, 0.3, 0.45, 0.55, 0.7, and 0.8, respectively. The negative indicators are the opposite.

$$V(x) = \begin{cases} 0, & x < x_5 \\ \frac{x - x_5}{x_6 - x_5}, & x_5 \le x \le x_6 \\ 1, & x_6 < x \end{cases} \tag{5}$$

$$V(x) = \begin{cases} \frac{x - x_3}{x_4 - x_3}, & x_3 \le x < x_4 \\ 1, & x_4 \le x < x_5 \\ \frac{x_6 - x}{x_6 - x_5}, & x_5 \le x < x_6 \\ 0, & x < x_3 \; or \; x_6 < x \end{cases} \tag{6}$$

$$V(x) = \begin{cases} \frac{x - x_1}{x_2 - x_1}, & x_1 \le x < x_2 \\ 1, & x_2 \le x < x_3 \\ \frac{x_4 - x}{x_4 - x_3}, & x_3 \le x < x_4 \\ 0, & x < x_1 \; or \; x_4 < x \end{cases} \tag{7}$$

$$V(x) = \begin{cases} 1, & x < x_1 \\ \frac{x_2 - x}{x_2 - x_1}, & x_1 \le x \le x_2 \\ 0, & x_2 < x \end{cases} \tag{8}$$

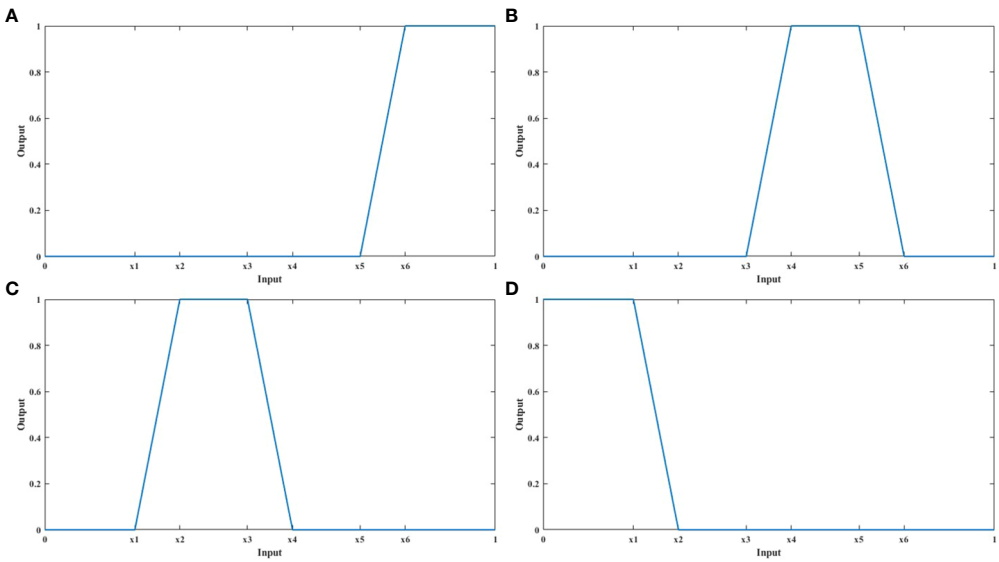


FIGURE 8
The fuzzy membership functions of the positive indicators were presented. **(A–D)** shows the fuzzy membership functions of four salt tolerance ratings: 'Very tolerant', 'Intermediate', 'Susceptible' and 'Very susceptible', respectively.

5. The fuzzy comprehensive evaluation matrix r_{Pi} of m indicators of each sample was calculated, and the membership degree $r = [r_1\ r_2\ r_3\ r_4]$ of each factor to the evaluation grade was calculated according to the membership function constructed by step 4.

$$r_{p_i} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ \vdots & \vdots & \vdots & \vdots \\ r_{m1} & r_{m2} & r_{m4} & r_{m4} \end{bmatrix} \quad (9)$$

6. For each sample, the fuzzy operator of multiplication and addition was used for fuzzy transformation, and the weight and fuzzy comprehensive evaluation matrix were synthesized into a fuzzy vector b. The fuzzy operator of multiplication and addition has a strong degree of synthesis, reflects the weight function obviously, and makes full use of the information of membership degree, which belongs to the weighted average type synthesis operator method.

$$b_{p_i} = \sum_{j=1}^{m} \left(w_j \cdot r_{j(p_i)} \right) \quad (10)$$

Where, w_j is the weight of the jth evaluation index of p_i sample, $r_{j(p_i)}$ is the fuzzy comprehensive evaluation matrix of the jth evaluation index of p_i sample, and b_{p_i} is the fuzzy vector of p_i sample.

7. The maximum membership principle was used to judge the salt tolerance of the samples.

$$U_k(p_i) = \bigvee_{k=1}^{4} b_k(p_i) \quad (11)$$

Where, $b_k(p_i)$ is the membership degree of p_i sample belonging to the kth evaluation level, $U_k(p_i)$ is the membership degree of p_i sample belonging to the kth evaluation level of the evaluation set, and $\vee$ is the operation of "taking large" in fuzzy mathematics. In this paper, the scores of 100, 75, 50 and 25 were set to represent four evaluation ratings of 'very tolerance', 'intermediate', 'susceptible' and 'very susceptible', respectively.

3 Results

3.1 Statistical analysis of phenotypic traits

Firstly, the phenotypic traits were statistically analyzed to explore the sensitivity of salt tolerance phenotypic traits to different statistical indicators, so as to determine weighting method of the FCE-E. Tables 2, 3 show the coefficient of variation, information entropy and variance contribution rate of TPT and CPT under CK and NaCl conditions, respectively. It could be seen from Table 2 that the variation coefficient and information entropy of FW, DW and WC were significantly higher than that of chlorophyll and the variation coefficient was above 0.3 after processing of two modes, which showed that different varieties of alfalfa in fresh weight, dry weight and water content of salt tolerance level had great differences and had a greater contribution in salt resistance evaluation. In addition, compared with the CK group, the coefficient of variation of chlorophyll increased, which also showed that chlorophyll had an indicative effect on alfalfa's salt tolerance, which could be used for salt tolerance identification of alfalfa. It could be seen from Table 3 that compared with the CK group, the coefficient of variation of the computing phenotypic traits in the NaCl group also changed to different degrees. Due to the influence of salt stress, most indexes increased and they were mainly concentrated in the indicators that represented structural growth, chlorophyll, stress index and biomass, and the information entropy of these indexes accounted for a larger proportion, indicating that these indexes could capture the salt tolerance of alfalfa more sensitively. In addition, the results showed that compared with the TPT, the CPT could find more salt tolerance phenotypes.

It was worth noting that in different treatment conditions or different evaluation index systems, the variance contribution rate was relatively average in all phenotypic traits, while the information entropy showed a differential distribution. Information entropy was more able to find the difference of different phenotypic characters of alfalfa under salt stress. Therefore, the FCE-E method in this paper allocated weight proportionally according to the size of information entropy.

3.2 Comprehensive evaluation of salt tolerance based on the typical phenotypic traits

In order to verify the evaluation ability of the multi-index FCE-E model based on the CPT in this paper, the four TPT of FW, DW,

TABLE 2 Statistical analysis of the typical phenotypic traits.

Index system	Traits	CK			NaCl		
		CV	H	Var	CV	H	Var
Typical phenotypic traits	FW	0.35	0.34	0.25	0.35	0.29	0.28
	DW	0.33	0.29	0.23	0.40	0.36	0.28
	WC	0.36	0.36	0.25	0.34	0.28	0.28
	CHL	0.05	0.01	0.27	0.17	0.07	0.16

CV is the coefficient of variation, H is the information entropy, and Var is the variance contribution rate.

TABLE 3 Statistical analysis of the computing phenotypic traits.

Index system	Traits	CK			NaCl		
		CV	H	Var	CV	H	Var
Computing phenotypic traits	Height	0.15	0.06	0.06	0.21	0.07	0.06
	Leaf-area	0.15	0.06	0.02	0.25	0.10	0.05
	Volume	0.18	0.08	0.05	0.25	0.10	0.06
	SR680	0.17	0.07	0.06	0.20	0.06	0.07
	PSSRb	0.19	0.10	0.06	0.23	0.08	0.06
	Datt	0.07	0.01	0.06	0.17	0.04	0.07
	mNDVI	0.07	0.01	0.05	0.20	0.06	0.07
	BN	0.09	0.02	0.06	0.16	0.04	0.07
	ARI	0.14	0.05	0.04	0.13	0.03	0.04
	CRI	0.17	0.07	0.05	0.16	0.04	0.06
	Srapa	0.17	0.07	0.06	0.21	0.07	0.07
	TVI	0.10	0.03	0.06	0.08	0.01	0.06
	WI	0.02	0.00	0.05	0.01	0.00	-0.02
	SWSI	0.07	0.01	0.06	0.10	0.01	0.06
	SRVI	0.18	0.09	0.06	0.22	0.07	0.06
	CTR	0.26	0.18	0.05	0.22	0.07	-0.03
	NDVIlut	0.11	0.03	0.06	0.23	0.08	0.07
	NDRI	0.13	0.05	0.06	0.23	0.08	0.06
	NDVI	0.03	0.00	0.06	0.06	0.01	0.06

WC and CHL were used as the evaluation index system, and the salt tolerance evaluation of alfalfa was carried out by the FCE-E method. The alfalfa was divided into four grades, and the results are shown in Figures 9A, B. Under NaCl condition, the growth of other alfalfa varieties was affected, except that the salt tolerance rating of Aohan, Gibraltar and Gold Empress were improved compared with the CK condition. Aohan had a poor rating under CK conditions, on the contrary, it was rated as intermediate salt tolerance under NaCl, indicating that the variety was not suitable for cultivation in undamaged land and was more suitable for cultivation and improvement promotion in saline-alkali land. Cangzhou had a poor rating under salt stress conditions, indicating that the variety of alfalfa was not suitable for cultivation on saline-alkali land.

3.3 Comprehensive evaluation of salt tolerance based on the computing phenotypic traits

Using the CPT as the multi-index evaluation system of salt tolerance, the FCE-E method was used to construct a non-destructive evaluation model of salt tolerance of alfalfa. Alfalfa was divided into four grades (Figures 9C, D) and compared with the evaluation result of salt tolerance based on TPT (in section 3.2). Under the condition of salt stress (Figure 9D), WL343HQ and Gibraltar were rated as salt-tolerant alfalfa varieties, Aohan and Gold Empress were rated as intermediate salt-tolerant alfalfa varieties, all Zhongmu No. 3 were rated as susceptible salt-sensitive alfalfa varieties, and Cangzhou were all highly salt susceptible varieties. Therefore, 3 highly tolerant, 4 intermediate, 3 susceptible and 2 highly susceptible materials were screened in this paper.

Compared to section 3.2, a total of 66.67% had consistent results. Among them, in the CK group, there were 75% consistent results; there were 58.33% consistent results in the NaCl group. Affected by salt stress, the physical and chemical characteristics of alfalfa would change in varying degrees. Compared with TPT, the CPT could find more salt-sensitive information. Therefore, the consistency rate of the two results in the NaCl group is reduced.

3.4 Comprehensive evaluation of salt tolerance by PCA method

To further explore the reliability of the multi-index FCE-E model, the PCA method was compared with the FCE-E method. The results are shown in Table 4. In the CK group, the first principal component accounted for 87.03% based on the PCA results of TPT, and 85.43% based on the CPT. In the NaCl group, the first principal

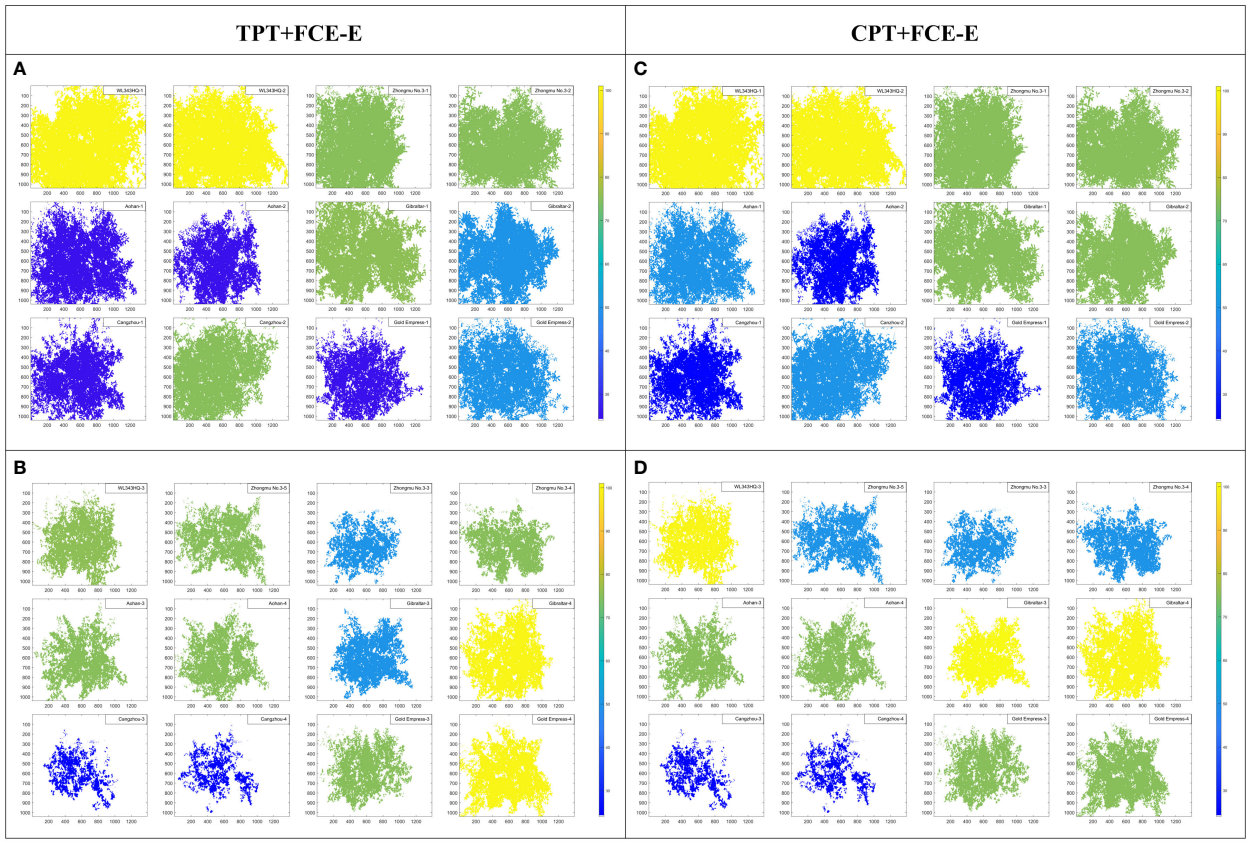


FIGURE 9
Salt tolerance evaluation results based on FCE-E. **(A, B)** are the results of salt tolerance evaluation were based on typical phenotypic traits. **(A)** for the CK groups, and **(B)** for the NaCl groups. **(C, D)** are the results of salt tolerance evaluation are based on computing phenotypic traits. **(C)** for the CK groups, and **(D)** for the NaCl groups. The scores of 100, 75, 50 and 25 represented the ratings of 'very tolerance', 'intermediate', 'susceptible' and 'very susceptible', respectively.

component based on TPT accounted for 88.19%, and the first principal component based on the CPT accounted for 79.43%. The first principal component had good explanatory power, so only the first principal component was extracted as the composite score. The results of Figures 9, 10 showed that the results of PCA were basically consistent with the results of the multi-index FCE-E, but the results of PCA could only rank the salt tolerance of the samples, and could not determine the salt tolerance level. Therefore, the results of the multi-index FCE-E model are more reliable and intuitive.

4 Discussion

The problem of soil salinization is becoming more and more serious worldwide, and screening and promoting salt-tolerant forage is the main way to improve and rationally use salinized soils (Singh et al., 2018). At present, the salt tolerance of maize (Fortmeier and Schubert, 1995), rice (Kumar et al., 2012) and other major crops has been deeply studied. Alfalfa, as the most widely planted and salt-tolerant forage, has high research value, so it has been studied in term of growth monitoring, salt tolerance mechanism and salt tolerance screening of alfalfa. However, for the non-destructive screening of alfalfa salt tolerance, there is still a lack of an accurate and systematic salt tolerance evaluation method. Miao et al. comprehensively evaluated the survival rate, plant height, number of green leaves, number of wilted leaves, number of branches and aboveground biomass of alfalfa seedlings by the membership function method (Lihong et al., 2016). Wu et al. explored the genetic diversity of alfalfa germplasm resources by using morphological indexes, agronomic traits and quality traits. The results showed that the agronomic traits of different alfalfa germplasm had the largest variation, followed by morphological traits and quality traits (Xinming et al., 2018). In this paper, based on previous studies, the CPT of alfalfa was constructed using HSI data and LiDAR data. The multi-index FCE-E model was used to evaluate the salt tolerance of 24 alfalfa materials, so as to obtain highly salt-tolerant and highly salt-susceptible varieties.

4.1 Advantages of computing phenotypic traits in comprehensive evaluation of salt tolerance

Combining multiple indicators to screen for salt-tolerant materials is the most reliable research method today (Hu et al., 2018).

10.3389/fpls.2023.1200501

TABLE 4 PCA salt tolerance rating results of CK and NaCl groups.

Treatment	Sample	TPT	Rank	CPT	Rank	Treatment	Sample	TPT	Rank	CPT	Rank
CK	WL343HQ-1	2.66	2	6.98	1	NaCl	WL343HQ-3	-1.45	3	-3.83	3
	WL343HQ-2	3.45	1	6.56	2		Zhongmu No.3-3	1.49	10	3.35	10
	Zhongmu No.3-1	1.26	3	2.95	3		Zhongmu No.3-4	0.06	8	0.53	8
	Zhongmu No.3-2	0.57	5	1.16	4		Zhongmu No.3-5	0.07	9	1.24	9
	Aohan-1	-0.64	8	-0.33	8		Aohan-3	-0.26	6	-0.61	6
	Aohan-2	-2.85	12	-5.40	11		Aohan-4	-0.66	4	-0.35	7
	Gibraltar-1	-0.01	6	0.25	6		Gibraltar-3	-0.10	7	-4.98	2
	Gibraltar-2	-0.09	7	0.88	5		Gibraltar-4	-3.21	1	-5.87	1
	Cangzhou-1	-2.23	10	-4.26	10		Cangzhou-3	3.44	12	6.91	12
	Cangzhou-2	1.01	4	-0.32	7		Cangzhou-4	3.18	11	6.44	11
	Gold Empress-1	-2.40	11	-6.30	12		Gold Empress-3	-0.30	5	-0.96	5
	Gold Empress-2	-0.73	9	-2.19	9		Gold Empress-4	-2.26	2	-1.87	4

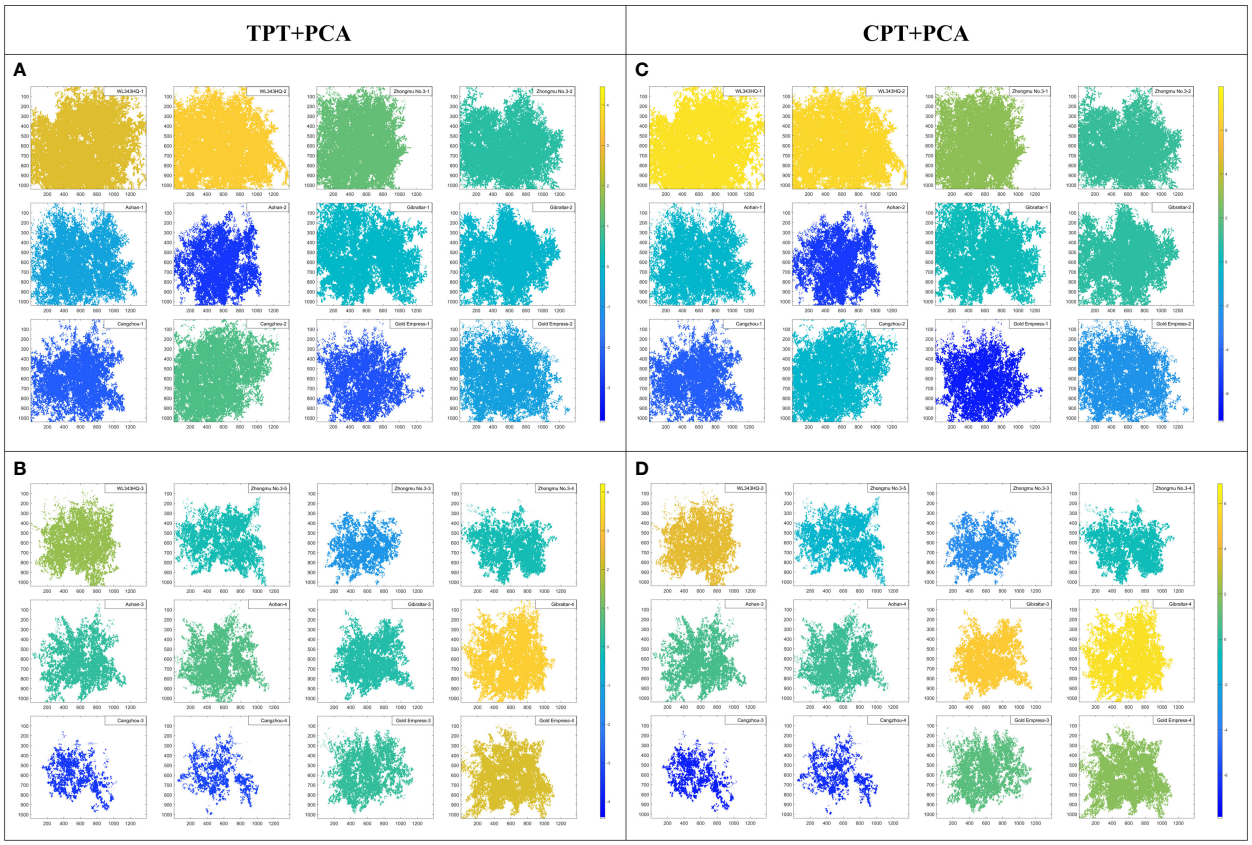


FIGURE 10
Salt tolerance evaluation results based on PCA. **(A, B)** are the results of salt tolerance evaluation were based on typical phenotypic traits. **(A)** for the CK groups, and **(B)** for the NaCl groups. **(C, D)** are the results of salt tolerance evaluation are based on computing phenotypic traits. **(C)** for the CK groups, and **(D)** for the NaCl groups.

However, the traditional method of obtaining phenotypic traits has the disadvantages of low measurement flux, time-consuming and labor-consuming, and data acquisition is difficult, especially for large-scale measurements. At present, the traditional phenotypic traits can no longer meet the needs of the rapidly developing plant stress resistance research, which seriously hinders the excavation of alfalfa salt-tolerant germplasm, so the high-throughput CPT have emerged (Wu et al., 2021). A series of changes occurred in its internal physiological components and external morphological structure after alfalfa was treated with salt stress.

Firstly, different spectral bands of HSI can capture the differences in various pigments, moisture content and biomass in alfalfa. Hyperspectral imaging technology was used to supplement and expand the TPT. For example, the contents of pigments such as anthocyanin and salt stress index of alfalfa were non-destructively extracted by using spectral phenotypic traits compared with the measured TPT. Taking the chlorophyll phenotypic trait SR680 vegetation index of alfalfa as an example, Figure 11 visually shows the difference between alfalfa in CK groups and NaCl groups. It could be seen from the figure that the chlorophyll content of alfalfa in the CK groups were significantly higher than that in the NaCl groups. In the single-basin canopy scale, the chlorophyll content in the leaf center was higher than that in the leaf edge.

Secondly, LiDAR technology can accurately obtain its 3D structure information. In order to evaluate the accuracy of the structural phenotypic traits extracted from the LiDAR data, we conducted a correlation analysis between the number of pixels representing the alfalfa area extracted from the HSI data and the canopy leaf area extracted from the LiDAR data. The correlation coefficient (R2) was found to be 0.7434, indicating a good quality of the canopy leaf area extracted from the LiDAR data.

Finally, it could be found from Figure 11 that the CPT combine spectral and spatial dimension information, which can sensitively capture the differences of alfalfa under salt stress, and contribute to more comprehensive and lower cost evaluation of salt tolerance and germplasm screening of alfalfa. The results of salt-tolerant germplasm screening of alfalfa based on the CPT were similar to

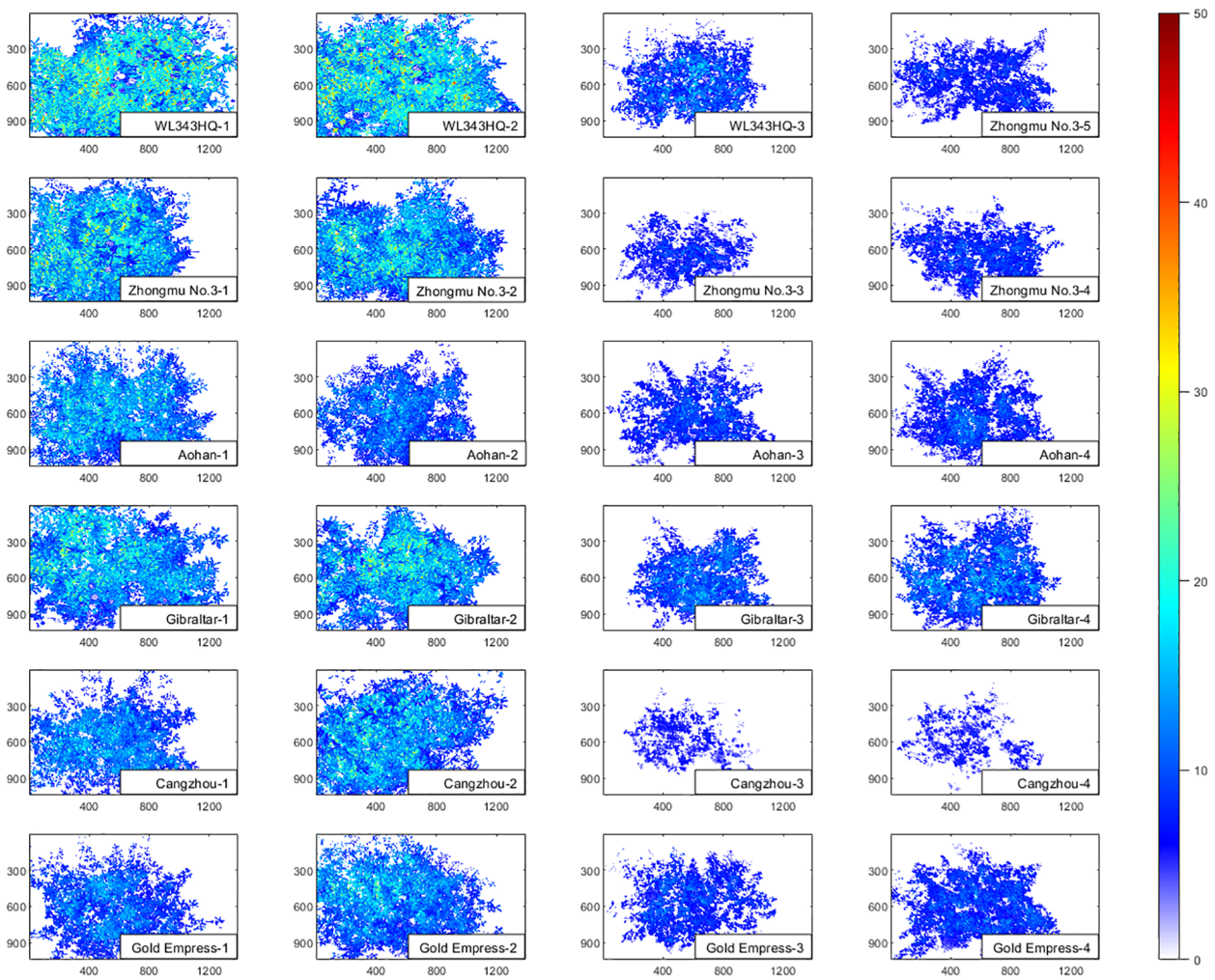


FIGURE 11
The schematic diagram of SR680 vegetation index of alfalfa in CK groups and NaCl groups. The left two columns are CK groups, and the right two columns are NaCl groups.

those of the TPT, and the results of the former were more consistent with those of manual screening, which provided a basis for automatic, low-difficulty and high-time-based salt-tolerance evaluation and breeding (Figure 9). Since more components were not detected in this material, they were not reflected in the TPT. More typical phenotypic traits should be added in future comprehensive rating experiments as an evaluation reference.

4.2 Portability and sensitivity of the multi-index FCE-E model

In the CK and NaCl groups, different phenotypic traits were affected by the growth environment to different degrees. Salt stress can hinder the growth of alfalfa, reduce the rate of leaf differentiation, slow down photosynthesis, and lead to different degrees of decrease or increase in phenotypic traits. The multi-index FCE-E model in this paper determined the weight of each trait in the evaluation of salt tolerance according to the information entropy of each factor. The smaller the information entropy of each trait was, the smaller the weight and the contribution to the salt tolerance rating would be. The PCA method determined the weight according to the variance contribution rate of each trait. The smaller the variance contribution rate was, the smaller the assigned weight and the smaller the contribution to salt tolerance rating would be. According to Tables 2, 3, the variance contribution rate of each phenotypic trait in the CK and NaCl groups were relatively balanced, and the information entropy presented a differential distribution. Therefore, compared with the variance contribution rate, the information entropy could better capture the response difference of phenotypic traits under salt stress. On the one hand, the entropy weight method could objectively assign weights to different phenotypic traits. On the other hand, it could adapt to the influence of different phenotypic traits on the salt tolerance evaluation model in different treatment conditions and increase the portability of the model. In addition, the FCE-E method in this paper adopted the method of adaptive adjustment when setting the membership function parameters, which also increases the portability of the model.

To test the sensitivity of the multi-index FCE-E model in evaluating the salt tolerance of alfalfa, the CK and NaCl groups were combined, and the multi-index FCE-E comprehensive ratings were performed on the same sample set. The results of salt tolerance evaluation of 24 samples (Figure 12) showed that the growth status of all varieties of the NaCl group was worse than that of the CK group, indicating that the multi-index FCE-E model based on the CPT could effectively characterize the salt tolerance of alfalfa. Therefore, according to section 3.3 (Figures 9C, D), the salt tolerance of 12 sample materials in the NaCl group was determined, and 3 highly salt-tolerant, 4 intermediate, 3 susceptible and 2 highly susceptible materials were obtained. In addition, as shown in Figure 9, WL343HQ had better growth performance in both the CK and NaCl groups, indicating that WL343HQ had more stable salt tolerance and could be cultivated and promoted in more regions.

4.3 Comparison of the multi-index FCE-E and the PCA

The results of the comprehensive evaluation of salt tolerance of alfalfa by the multi-index FCE-E method and the PCA method proposed in this paper (Figures 9, 10) showed that the two methods got relatively consistent results, and the multi-index FCE-E method could select salt-tolerant and highly salt-susceptible materials more intuitively, while the PCA method could only get the ranking of salt tolerance degree of alfalfa, and the selection of salt-tolerance materials could not be directly derived from the results. In addition, subjective factors are involved in the determination of the contribution rate of principal components in the comprehensive evaluation of salt tolerance using the PCA method.

The multi-index FCE-E method proposed in this paper was based on the probability and statistical theory of fuzzy mathematics, and the method could objectively evaluate the salt tolerance of alfalfa by using adaptive parameter settings. The results of multi-index FCE-E were the rating of salinity tolerance of alfalfa, the results of which intuitively gave the corresponding level in all materials. The better the result was, the better its growth status and tolerance to salt stress would be. Moreover, the results of salt tolerance evaluation based on the two evaluation index systems were similar, so the growth status of alfalfa could be evaluated and its salt tolerance could be judged by nondestructive CPT.

4.4 Application of the FCE-E method in large-scale breeding programs

For plant breeders, the FCE-E method can help them make many improvements in large-scale breeding programs. First, plant breeders can evaluate the salt tolerance of different alfalfa varieties based on this method. According to this method, the high-quality varieties with the highest salt tolerance score of alfalfa were determined, and the varieties for in-depth breeding or commercial promotion were preliminarily screened. Secondly, this method can help plant breeders determine more important phenotypic traits in different batches of alfalfa. By analyzing the weights assigned to each trait, breeders can explore potential breeding advantages and defects, and focus on the most critical traits, which can improve the accuracy and efficiency of breeding. Finally, the use of LiDAR and HSI data can shorten the breeding time and reduce the breeding cost. Through non-destructive measurement, breeders can simultaneously evaluate large areas of alfalfa, thereby quickly optimizing breeding strategies. In general, this method can help breeders select the best varieties and

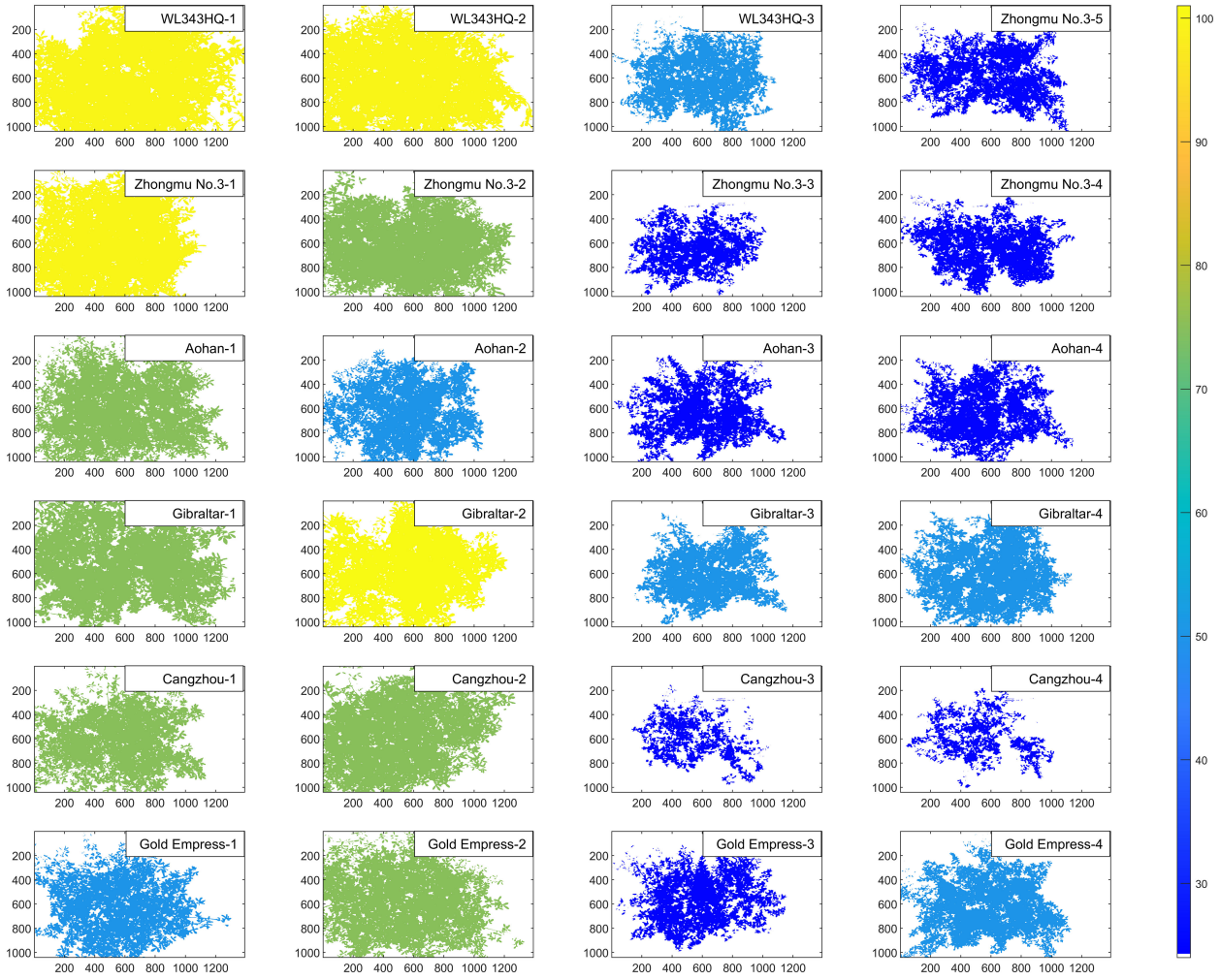


FIGURE 12
Results of salt tolerance evaluation of 24 samples using the FCE-E method. The left two columns are CK groups, and the right two columns are NaCl groups.

breeding strategies through systematic evaluation of physiology, biochemistry and structure, and improve the efficiency and effectiveness of large-scale breeding programs.

5 Conclusions

Through the improved fuzzy comprehensive evaluation method, the salt tolerance of 6 alfalfa varieties was comprehensively evaluated in this study, and the following main conclusions were obtained: Firstly, the multi-index FCE-E method proposed in this paper can accurately evaluate the response ability of alfalfa to salt stress. Secondly, this method combines entropy method and fuzzy comprehensive evaluation method to capture the differential performance of alfalfa phenotypic traits in response to salt stress in a more objective and sensitive way. In addition, the method uses the calculated phenotypic traits as the data source, which can more quickly and comprehensively find the differences and changes of alfalfa physical and chemical parameters and structural parameters.

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

AZ and JZ designed the research. All authors performed the experiments and analyzed the data. AZ and JZ initially drafted the manuscript with inputs from other authors. All the authors read and approved the final manuscript. All authors contributed to the article and approved the submitted version.

Funding

The research was funded by the Joint program of Beijing Municipal Education Commission and Beijing Municipal Natural Science Foundation, Grant Number KZ202110028044, and Science and Technology Program of Qinghai Province of China, Grant Number 2022-NK-136, and National Natural Science Foundation of China, Grant Number (42071303, 41571369).

Acknowledgments

In this paper, we would like to thank the team of Prof. Zhang Jingyu of the Institute of Botany, Chinese Academy of Sciences for their support and help in experimental data collection.

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

Al-Tamimi, N., Langan, P., Bernad, V., Walsh, J., Mangina, E., and Negrao, S. (2022). Capturing crop adaptation to abiotic stress using image-based technologies. *Open Biol.* 12 (6), 210353. doi: 10.1098/rsob.210353

Aparicio, N., Villegas, D., Araus, J. L., Casadesus, J., and Royo, C. (2002). Relationship between growth traits and spectral vegetation indices in durum wheat. *Crop Sci.* 42, 1547–1555. doi: 10.2135/cropsci2002.1547

Barnes, E., Clarke, T. R., Richards, S. E., Colaizzi, P., Haberland, J., Kostrzewski, M., et al. (2000). Coincident detection of crop water stress, nitrogen status, and canopy density using ground based multispectral data. *Environ. Sci.* 1619, 1–15.

Blackburn, G. A. (2010). Spectral indices for estimating photosynthetic pigment concentrations: A test using senescent tree leaves. *Int. J. Remote Sens.* 19 (4), 657–675. doi: 10.1080/014311698215919

Broge, N. H., and Leblanc, E. (2001). Comparing prediction power and stability of broadband and hyperspectral vegetation indices for estimation of green leaf area index and canopy chlorophyll density. *Remote Sens. Environ.* 76 (2), 156–172. doi: 10.1016/S0034-4257(00)00197-8

Brook, A., Tal, Y., Markovich, O., and Rybnikova, N. (2021). Canopy Volume as a Tool for Early Detection of Plant Drought and Fertilization Stress: Banana plant fine-phenotype. *bioRxiv* doi: 10.1101/2021.03.04.433868

Buschmann, C., and Nagel, E. (2007). *In vivo* spectroscopy and internal optics of leaves as basis for remote sensing of vegetation. *Int. J. Remote Sens.* 14 (4), 711–722. doi: 10.1080/01431169308904370

Carter, G. A. (2007). Ratios of leaf reflectances in narrow wavebands as indicators of plant stress. *Int. J. Remote Sens.* 15 (3), 697–703. doi: 10.1080/01431169408954109

Chunjiang, Z. (2019). Big data of plant phenomics and its research progress. *J. Agric. Big Data* 1 (02), 5–18. doi: 10.19788/j.issn.2096-6369.190201

Datt, B. (2010). Visible/near infrared reflectance and chlorophyll content in Eucalyptus leaves. *Int. J. Remote Sens.* 20 (14), 2741–2759. doi: 10.1080/014311699211778

Du, Y.-W., Wang, S.-S., and Wang, Y.-M. (2019). Group fuzzy comprehensive evaluation method under ignorance. *Expert Syst. Appl.* 126, 92–111. doi: 10.1016/j.eswa.2019.02.006

Duan, J., Li, J., Guo, S., and Kang, Y. (2008). Exogenous spermidine affects polyamine metabolism in salinity-stressed Cucumis sativus roots and enhances short-term salinity tolerance. *J. Plant Physiol.* 165 (15), 1620–1635. doi: 10.1016/j.jplph.2007.11.006

Duo, L., Shuang, B., Zhijie, L., Qingshan, Y., Xuebin, Q., Wei, G., et al. (2021). A review on the saline-alkaline tolerance of alfalfa (Medicago sativa L.). *J. OF Biol.* 38 (01), 98–101+105. doi: 10.3969/ j.issn.2095-1736.2021.01.098

El-Hendawy, S., Al-Suhaibani, N., Elsayed, S., Alotaibi, M., Hassan, W., and Schmidhalter, U. (2019a). Performance of optimized hyperspectral reflectance indices and partial least squares regression for estimating the chlorophyll fluorescence and grain yield of wheat grown in simulated saline field conditions. *Plant Physiol. Biochem.* 144, 300–311. doi: 10.1016/j.plaphy.2019.10.006

El-Hendawy, S. E., Al-Suhaibani, N. A., Hassan, W. M., Dewir, Y. H., Elsayed, S., Al-Ashkar, I., et al. (2019b). Evaluation of wavelengths and spectral reflectance indices for high-throughput assessment of growth, water relations and ion contents of wheat irrigated with saline water. *Agric. Water Manage.* 212, 358–377. doi: 10.1016/j.agwat.2018.09.009

El-Hendawy, S., Dewir, Y. H., Elsayed, S., Schmidhalter, U., Al-Gaadi, K., Tola, E., et al. (2022). Combining hyperspectral reflectance indices and multivariate analysis to estimate different units of chlorophyll content of spring wheat under salinity conditions. *Plants* 11 (3), 456. doi: 10.3390/plants11030456

Fortmeier, R., and Schubert, S. (1995). Salt tolerance of maize (Zea mays L.) : The role of sodium exclusion. *Plant Cell Environ.* 18 (9), 1041–1047. doi: 10.1111/j.1365-3040.1995.tb00615.x

Gitelson, A. A., Merzlyak, M. N., and Chivkunova, O. B. (2001). Optical properties and nondestructive estimation of anthocyanin content in plant leaves. *Photochem. Photobiol.* 74 (1), 38–45. doi: 10.1562/0031-8655(2001)0740038OPANEO2.0.CO2

Gitelson, A. A., Zur, Y., Chivkunova, O. B., and Merzlyak, M. N. (2002). Assessing carotenoid content in plant leaves with reflectance spectroscopy. *Photochem. Photobiol.* 75 (3), 272–281. doi: 10.1562/0031-8655(2002)0750272ACCIPL2.0.CO2

Guiza, M., Benabdelrahim, M. A., Brini, F., Haddad, M., and Saibi, W. (2022). Assessment of alfalfa (Medicago sativa L.) cultivars for salt tolerance based on yield, growth, physiological, and biochemical traits. *J. Plant Growth Regul.* 41 (8), 3117–3126. doi: 10.1007/s00344-021-10499-9

Hamzeh, S., Naseri, A. A., AlaviPanah, S. K., Mojaradi, B., Bartholomeus, H. M., Clevers, J. G. P. W., et al. (2013). Estimating salinity stress in sugarcane fields with spaceborne hyperspectral vegetation indices. *Int. J. Appl. Earth Observation Geoinformation* 21, 282–290. doi: 10.1016/j.jag.2012.07.002

Hanin, M., Ebel, C., Ngom, M., Laplaze, L., and Masmoudi, K. (2016). New insights on plant salt tolerance mechanisms and their potential use for breeding. *Front. Plant Sci.* 7. doi: 10.3389/fpls.2016.01787

Hongyan, L., and Zongli, W. (2007). *Descriptors and Data Standard for Medic (Medicago Linn.)* (China Agriculture Press).

Houle, D., Govindaraju, D., and Omholt, S. (2010). Phenomics: the next challenge. *Nat. Rev. Genet.* 11 (12), 855–866. doi: 10.1038/nrg2897

Hu, Y., and Schmidhalter, U. (2023). Opportunity and challenges of phenotyping plant salt tolerance. *Trends Plant Sci.* 28 (5), 552–566. doi: 10.1016/j.tplants.2022.12.010

Hu, D., Wu, D., You, J., He, Y., and Qian, W. (2018). Principal component analysis and comprehensive evaluation on salt tolerance related traits in Brassica napus L. *Botanical Res.* 07 (02), 101–112. doi: 10.12677/br.2018.72014

Hunt, E. R., Doraiswamy, P. C., McMurtrey, J. E., Daughtry, C. S. T., Perry, E. M., and Akhmedov, B. (2013). A visible band index for remote sensing leaf chlorophyll content at the canopy scale. *Int. J. Appl. Earth Observation Geoinformation* 21, 103–112. doi: 10.1016/j.jag.2012.07.020

Ismail, A. M., and Horie, T. (2017). Genomics, physiology, and molecular breeding approaches for improving salt tolerance. *Annu. Rev. Plant Biol.* 68 (1), 405–434. doi: 10.1146/annurev-arplant-042916-040936

Jimenez-Berni, J. A., Deery, D. M., Rozas-Larraondo, P., Condon, A. G., Rebetzke, G. J., James, R. A., et al. (2018). High throughput determination of plant height, ground cover, and above-ground biomass in wheat with LiDAR. *Front. Plant Sci.* 9. doi: 10.3389/fpls.2018.00237

Jin, S., Sun, X., Wu, F., Su, Y., Li, Y., Song, S., et al. (2021a). Lidar sheds new light on plant phenomics for plant breeding and management: Recent advances and future prospects. *ISPRS J. Photogrammetry Remote Sens.* 171, 202–223. doi: 10.1016/j.isprsjprs.2020.11.006

Jin, X., Zarco-Tejada, P. J., Schmidhalter, U., Reynolds, M. P., Hawkesford, M. J., Varshney, R. K., et al. (2021b). High-throughput estimation of crop traits: A review of ground and aerial phenotyping platforms. *IEEE Geosci. Remote Sens. Magazine* 9 (1), 200–231. doi: 10.1109/mgrs.2020.2998816

Kasim, N., Qingdong, S., Jingzhe, W., Sawut, R., Nurmemet, I., and Isak, G. (2017). Estimation of spring wheat chlorophyll content based on hyperspectral features and PLSR model. *Trans. Chin. Soc. Agric. Eng. (Transactions CSAE)* 33 (22), 208–216. doi: 10.11975/j.issn.1002-6819.2017.22.027

Kumar, R., Mustafiz, A., Sahoo, K. K., Sharma, V., Samanta, S., Sopory, S. K., et al. (2012). Functional screening of cDNA library from a salt tolerant rice genotype Pokkali identifies mannose-1-phosphate guanyl transferase gene (OsMPG1) as a key member of salinity stress response. *Plant Mol. Biol.* 79 (6), 555–568. doi: 10.1007/s11103-012-9928-8

Li, J., Schachtman, D. P., Creech, C. F., Wang, L., Ge, Y., and Shi, Y. (2022). Evaluation of UAV-derived multimodal remote sensing data for biomass prediction and drought tolerance assessment in bioenergy sorghum. *Crop J.* 10 (5), 1363–1375. doi: 10.1016/j.cj.2022.04.005

Lihong, M., Qinghua, Y., and Yu, W. (2016). Evaluation on salt tolerance of 41 alfalfa varieties at seedling stage. *Seed* 35 (04), 81–84. doi: 10.16590/j.cnki.1001-4705.2016.04.081

Luther, J. E., and Carroll, A. L. (1999). Development of an index of balsam fir vigor by foliar spectral reflectance. *Remote Sens. Environ.* 69 (3), 241–252. doi: 10.1016/S0034-4257(99)00016-4

M, L.Y., H, G.Q., B, W., N, Q.H., Z, W. D., X, X.K., et al. (2021). Current status and prospect of threedimensional dynamic monitoring of natural resources based on LiDAR. *Natl. Remote Sens. Bull.* 25 (1), 381–402. doi: 10.11834/jrs.20210351

Munns, R., and Tester, M. (2008). Mechanisms of salinity tolerance. *Annu. Rev. Plant Biol.* 59 (1), 651–681. doi: 10.1146/annurev.arplant.59.032607.092911

Penglei, L., Xiao, Z., Wenhui, W., Hengbiao, Z., Xia, Y., Yan, Z., et al. (2021). Assessment of terretrial laser scanning and hypersprctral remote sensing for the estimation of rice grain yield. *Scientia Agricultura Sin.* 54 (14), 2965–2976. doi: 10.3864/j.issn.0578-1752.2021.14.004

Penuelas, J., Pinol, J., Ogaya, R., and Filella, I. (2010). Estimation of plant water concentration by the reflectance Water Index WI (R900/R970). *Int. J. Remote Sens.* 18 (13), 2869–2875. doi: 10.1080/014311697217396

Post, C. J., DeGloria, S. D., Cherney, J. H., and Mikhailova, E. A. (2007). Spectral measurements of alfalfa/grass fields related to forage properties and species composition. *J. Plant Nutr.* 30 (11), 1779–1789. doi: 10.1080/01904160701626951

Reddy, P., Guthridge, K. M., Panozzo, J., Ludlow, E. J., Spangenberg, G. C., and Rochfort, S. J. (2022). Near-infrared hyperspectral imaging pipelines for pasture seed quality evaluation: an overview. *Sensors (Basel)* 22 (5), 1981. doi: 10.3390/s22051981

Robati, M., and Rezaei, F. (2021). Evaluation and ranking of urban sustainability based on sustainability assessment by fuzzy evaluation model. *Int. J. Environ. Sci. Technol.* 19 (1), 625–650. doi: 10.1007/s13762-021-03128-1

Roy, S. J., Negrao, S., and Tester, M. (2014). Salt resistant crop plants. *Curr. Opin. Biotechnol.* 26, 115–124. doi: 10.1016/j.copbio.2013.12.004

Saric, R., Nguyen, V. D., Burge, T., Berkowitz, O., Trtilek, M., Whelan, J., et al. (2022). Applications of hyperspectral imaging in plant phenotyping. *Trends Plant Sci.* 27 (3), 301–315. doi: 10.1016/j.tplants.2021.12.003

Shangpeng, S., Changying, L., H., P. A., Yu, J., Rui, X., S., R. J., et al. (2018). In-field high throughput phenotyping and cotton plant growth analysis using LiDAR. *Front. Plant Sci.* 9. doi: 10.3389/fpls.2018.00016

Shaochen, L., Aiwu, Z., Xizhen, Z., Zhiqiang, Y., and Mengnan, L. (2023). Phenotype leaf angles of corn seedlings using computational methods. *Laser Optoelectronics Prog.* 60 (02), 71–79.

Shaohua, Z., Jianzhao, D., Li, H., Yuhang, J., Schulthess, U., Lashkari, A., et al. (2022). Wheat yield estimation from UAV platform based on multi-modal remote sensing data fusion. *Acta Agronomica Sin.* 48 (07), 1746–1760.

Sims, D. A., and Gamon, J. A. (2002). Relationships between leaf pigment content and spectral reflectance across a wide range of species, leaf structures and developmental stages. *Remote Sens. Environ.* 81, 337–354. doi: 10.1016/S0034-4257(02)00010-X

Singh, A. P., and Bhutia, P. (2022). High throughput phenotyping in crop improvement programmes. *Readers Shelf* 18 (09), 29–32.

Singh, A., Jones, S., Ganapathysubramanian, B., Sarkar, S., Mueller, D., Sandhu, K., et al. (2021). Challenges and opportunities in machine-augmented plant stress phenotyping. *Trends Plant Sci.* 26 (1), 53–69. doi: 10.1016/j.tplants.2020.07.010

Singh, V., Singh, A. P., Bhadoria, J., Giri, J., Singh, J., T, V. V., et al. (2018). Differential expression of salt-responsive genes to salinity stress in salt-tolerant and salt-sensitive rice (Oryza sativa L.) at seedling stage. *Protoplasma* 255 (6), 1667–1681. doi: 10.1007/s00709-018-1257-6

Song, P., Wang, J., Guo, X., Yang, W., and Zhao, C. (2021). High-throughput phenotyping: Breaking through the bottleneck in future crop breeding. *Crop J.* 9 (3), 633–645. doi: 10.1016/j.cj.2021.03.015

Sun, Z., Li, Q., Jin, S., Song, Y., Xu, S., Wang, X., et al. (2022). Simultaneous prediction of wheat yield and grain protein content using multitask deep learning from time-series proximal sensing. *Plant Phenomics* 2022, 13. doi: 10.34133/2022/9757948

Thenkabail, P. S., Mariotto, I., Gumma, M. K., Middleton, E. M., Landis, D. R., and Huemmrich, K. F. (2013). Selection of hyperspectral narrowbands (HNBs) and composition of hyperspectral twoband vegetation indices (HVIs) for biophysical characterization and discrimination of crop types using field reflectance and Hyperion/EO-1 data. *IEEE J. Selected Topics Appl. Earth Observations Remote Sens.* 6 (2), 427–439. doi: 10.1109/jstars.2013.2252601

Tilly, N., Aasen, H., and Bareth, G. (2015). Fusion of plant height and vegetation indices for the estimation of barley biomass. *Remote Sens.* 7 (9), 11449–11480. doi: 10.3390/rs70911449

Tilly, N., Hoffmeister, D., Cao, Q., Lenz-Wiedemann, V., Miao, Y., and Bareth, G. (2013). Precise plant height monitoring and biomass estimation with Terrestrial Laser Scanning in paddy rice. *ISPRS Ann. Photogrammetry Remote Sens. Spatial Inf. Sci.* II-5/W2, 295–300. doi: 10.5194/isprsannals-II-5-W2-295-2013

Tmušić, G., Manfreda, S., Aasen, H., James, M. R., Gonçalves, G., Ben-Dor, E., et al. (2020). Current practices in UAS-based environmental monitoring. *Remote Sens.* 12 (6), 1001. doi: 10.3390/rs12061001

Tucker, C. J. (1977). Spectral estimation of grass canopy variables. *Remote Sens. Environ.* 6 (1), 11–26. doi: 10.1016/0034-4257(77)90016-5

Tucker, C. J. (1979). Red and photographic infrared l,lnear combinations for monitoring vegetation. *Remote Sens. Environ.* 8 (2), 127–150. doi: 10.1016/0034-4257(79)90013-0

Ullah, S., Schlerf, M., Skidmore, A. K., and Hecker, C. (2012). Identifying plant species using mid-wave infrared (2.5–6μm) and thermal infrared (8–14μm) emissivity spectra. *Remote Sens. Environ.* 118, 95–102. doi: 10.1016/j.rse.2011.11.008

Wang, D., Wilson, C., and Shannon, M. C. (2010). Interpretation of salinity and irrigation effects on soybean canopy reflectance in visible and near-infrared spectrum domain. *Int. J. Remote Sens.* 23 (5), 811–824. doi: 10.1080/01431160110070717

Wu, X., Feng, H., Wu, D., Yan, S., Zhang, P., Wang, W., et al. (2021). Using high-throughput multiple optical phenotyping to decipher the genetic architecture of maize drought tolerance. *Genome Biol.* 22 (1), 185. doi: 10.1186/s13059-021-02377-0

Wu, C., Niu, Z., Tang, Q., and Huang, W. (2008). Estimating chlorophyll content from hyperspectral vegetation indices: Modeling and validation. *Agric. For. Meteorology* 148 (8-9), 1230–1241. doi: 10.1016/j.agrformet.2008.03.005

Xiangfeng, Z., Xiaorong, Y., and Ziwei, J. (2018). Research progress of salt tolerance evaluation in plants and tolerance evaluation strategy. *J. Biol.* 35 (06), 91–94. doi: 10.3969/j.issn.2095-1736.2018.06.091

Xiaofeng, C., Keqiang, Y., Yanru, Z., and Haihui, Z. (2020). Current status of high-throughput plant phenotyping for abiotic stress by imaging spectroscopy: a review. *Spectrosc. Spectral Anal.* 40 (11), 3365–3372. doi: 10.3964/j.issn.1000-0593(2020)11-3365-08

Xinming, W., Pu, G., Huiwu, C., Zhihong, F., Yonghong, S., Yunqi, W., et al. (2018). Diversity Analysis of Phenotypic Traits and Quality Characteristics of Alfalfa(Medicago sativa) introducted from abroad Germplasm Resources. *J. Plant Genet. Resour.* 19 (01), 103–111. doi: 10.13430/j.cnki.jpgr.2018.01.012

Xu, X., Nie, C., Jin, X., Li, Z., Zhu, H., Xu, H., et al. (2021). A comprehensive yield evaluation indicator based on an improved fuzzy comprehensive evaluation method and hyperspectral data. *Field Crops Res.* 270. doi: 10.1016/j.fcr.2021.108204

Zadeh., L. A. (1965). Fuzzy sets. *Inf. AND CONTROL* 8 (4), 338–353. doi: 10.2307/2272014

Zhang, X., Liu, X., Qi, M., Liu, Y., and Kuai, J. (2013). Alfalfa seeding root characteristics under complex saline-alkali stress. *Chin. J. Eco-Agriculture* 21 (3), 340–346. doi: 10.3724/SP.J.1011.2013.00340

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 11 September 2023
DOI 10.3389/fpls.2023.1211490

Check for updates

OPEN ACCESS

EDITED BY
Changcai Yang,
Fujian Agriculture and Forestry University, China

REVIEWED BY
Esa Prakasa,
National Research and Innovation Agency (BRIN) of Indonesia, Indonesia
Chuan Lu,
Aberystwyth University, United Kingdom

*CORRESPONDENCE
Carola Figueroa-Flores
cfigueroa@ubiobio.cl

RECEIVED 24 April 2023
ACCEPTED 15 August 2023
PUBLISHED 11 September 2023

CITATION
Figueroa-Flores C and San-Martin P (2023) Deep learning for Chilean native flora classification: a comparative analysis. *Front. Plant Sci.* 14:1211490. doi: 10.3389/fpls.2023.1211490

COPYRIGHT
© 2023 Figueroa-Flores and San-Martin. 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.

Deep learning for Chilean native flora classification: a comparative analysis

Carola Figueroa-Flores[1*] and Pablo San-Martin[2]

[1]Department of Computer Science and Information Technology, Universidad del Bío Bío, Chillán, Chile, [2]School of Computer and Information Engineering, Universidad del Bío-Bío, Chillán, Chile

The limited availability of information on Chilean native flora has resulted in a lack of knowledge among the general public, and the classification of these plants poses challenges without extensive expertise. This study evaluates the performance of several Deep Learning (DL) models, namely InceptionV3, VGG19, ResNet152, and MobileNetV2, in classifying images representing Chilean native flora. The models are pre-trained on Imagenet. A dataset containing 500 images for each of the 10 classes of native flowers in Chile was curated, resulting in a total of 5000 images. The DL models were applied to this dataset, and their performance was compared based on accuracy and other relevant metrics. The findings highlight the potential of DL models to accurately classify images of Chilean native flora. The results contribute to enhancing the understanding of these plant species and fostering awareness among the general public. Further improvements and applications of DL in ecology and biodiversity research are discussed.

KEYWORDS

image classification, Chilean native flora, convolutional neural network, deep learning, transfer learning

1 Introduction

Chile's expansive geographical territory encompasses a wide array of flora, influenced by its diverse climatic conditions. However, information regarding native flora is often restricted to informative panels with minimal details, primarily found in specific physical locations within national parks and protected areas (Rodriguez et al., 2018). This limited availability of information hinders visitors from acquiring comprehensive knowledge about the country's native flora, resulting in a low societal appreciation of wildlife and an insufficient understanding of the significance of biodiversity conservation. To address this issue, it is crucial to comprehend people's attitudes and intentions towards wildlife and explore the relationships between humans and various species within local ecosystems. In this context, technological advancements, such as computers, the internet, television, and video games, are considered to have contributed to a decrease in personal experiences with nature, consequently diminishing sensitivity towards environmental issues. In order to effectively conserve biodiversity, it is crucial to consider the perceptions of individuals

towards their natural environment. Proposed models of environmental perception should acknowledge humans as information processors and organizers, capable of constructing a coherent representation of the world to address challenges. One approach that aligns with this perspective is the utilization of computational image classification methods, which employ Deep Learning (DL) techniques to accurately recognize and classify species depicted in images. DL is widely recognized for its exceptional performance in solving real-world problems, as well as its capacity to handle large volumes of data. The ease with which relevant features can be extracted during the learning process, coupled with the utilization of Graphic Processing Units (GPUs), further expedites the learning process. DL is extensively employed by companies to extract knowledge from data generated by electronic devices, ranging from computers to smartwatches or activity-tracking bracelets. By leveraging DL, valuable insights regarding native flora can be obtained, ultimately playing a pivotal role in the conservation of Chile's biodiversity (Rodriguez et al., 2018; Carranza et al., 2020).

Image classification algorithms can be classified into three main categories: supervised, unsupervised, and weakly supervised. In supervised classification, the user selects a representative sample of pixels from an image to train the algorithm. Unsupervised classification, on the other hand, groups pixels based on common characteristics without the need for userdefined sample classes. Weakly supervised classification algorithms utilize weaker forms of supervision and can employ complete, exact, or inexact supervision.

There are several well-known classification algorithms used in image analysis. Convolutional Neural Networks (CNNs) have gained significant attention due to their outstanding performance in image classification tasks. Artificial Neural Networks (ANNs) are also widely used, as they mimic the structure and function of the human brain. Support Vector Machines (SVMs) excel in both classification and regression tasks, aiming to find an optimal hyperplane to separate different classes. K-Nearest Neighbors (KNN) is a straightforward yet powerful algorithm that assigns a class label to a new data point based on the labels of its nearest neighbors. Na¨ıve Bayes classifiers rely on Bayes' theorem and assume independence among features given the class label. Finally, Random Forest is an ensemble learning method that combines multiple decision trees for making predictions.

These algorithms offer diverse approaches to image classification, each with its own strengths and weaknesses. The choice of algorithm depends on the specific task requirements, dataset characteristics, and desired performance. Researchers and practitioners should carefully evaluate and select the most suitable algorithm for their specific application to achieve accurate and reliable classification results (Sen et al., 2020).

When it comes to flora classification, deep learning using CNN algorithms emerges as the most effective method. Since 2012, CNNs have established themselves as the primary algorithm for image classification. They have demonstrated exceptional accuracy in various visual recognition tasks such as object detection, localization, and semantic segmentation.

To identify the most suitable CNN for classifying native Chilean flora, a comparative analysis of renowned CNNs will be conducted. This analysis will include Inception, VGG19, MobileNet, and ResNet 152, all of which will be pre-trained on ImageNet. The evaluation will be performed on a comprehensive dataset consisting of over 5000 images of native Chilean flora, incorporating transfer learning techniques to leverage pre-existing knowledge from the ImageNet dataset. By examining and comparing the performance of these CNN models, we aim to determine the optimal choice for accurate and efficient classification of native Chilean flora.

We briefly summarize below our main contributions:

- We create a dataset of Chilean native flora species by labeling and augmenting images to enhance the model's training,
- We conducted a comparative analysis of the most wellknown CNN models to determine which one provided better accuracy in the classification task,
- After conducting experiments and evaluations, we determined which CNN model delivers better results in the task of classifying native Chilean flora with an accuracy rate of 90% with transfer learning.

The paper is organized as follows. Section II provides a comprehensive review of the related work in image classification, with a specific focus on the classification of Chilean native flora. This section examines the existing literature, highlighting key studies and approaches in the field.

In Section III, we present our proposed approach for classifying native Chilean flora using CNN models. We discuss the selection and configuration of the CNN models, as well as the preprocessing steps and training procedures employed in our methodology.

Section IV presents the experimental results of our study. We provide a detailed analysis of the performance of the CNN models on the native Chilean flora dataset, including accuracy, precision, recall, sensitivity, specificity, F-Score, AUC and time in milliseconds. Additionally, we discuss any notable findings or insights obtained from the experiments.

Finally, in Section V, we draw our conclusions based on the outcomes of our research. We summarize the main findings, discuss their implications for the classification of native Chilean flora, and highlight potential avenues for future research and development in this field.

By following this organization, we aim to provide a clear and structured presentation of our study, allowing readers to easily navigate and comprehend the content of the paper.

2 Related work

2.1 Convolutional neural networks

Convolutional Neural Networks (CNNs) have garnered significant attention and have been extensively explored in diverse domains, including computer vision, natural language processing, and speech recognition. Within the realm of computer vision, CNNs have demonstrated remarkable accomplishments in a wide range of tasks, encompassing image classification, object detection,

semantic segmentation, and image generation (Sarigul et al., 2019; Zhou, 2020). Their inherent ability to effectively capture and extract meaningful features from images has contributed to their widespread adoption and success in various visual recognition tasks. The utilization of CNNs has propelled advancements in the field of computer vision, paving the way for enhanced capabilities and improved performance in tasks that require sophisticated understanding and interpretation of visual data (Maggiori et al., 2017; Ashraf et al., 2019; Hui et al., 2020; Chen et al., 2021; Zelenina et al., 2022).

In recent years, there has been an escalating interest in enhancing the performance and efficiency of Convolutional Neural Networks (CNNs). This has led to significant advancements in the field, with notable contributions encompassing the development of novel architectural designs, including ResNet, Inception, MobileNet, and VGG. These architectures have been specifically engineered to reduce the number of parameters while simultaneously preserving or even improving classification accuracy. In addition to architectural innovations, other approaches have been explored to optimize the training process of CNNs. These include leveraging transfer learning, data augmentation, and regularization techniques. Transfer learning enables the utilization of pre-trained models on largescale datasets to improve generalization and efficiency. Data augmentation techniques enhance model robustness by artificially expanding the training dataset through various transformations and perturbations. Regularization techniques, on the other hand, impose constraints on the model's parameters to mitigate overfitting and enhance generalization. Collectively, these research efforts aim to refine and optimize CNNs, leading to improved performance and more efficient utilization in various computer vision tasks (Zhu and Chang, 2019; Pattnaik et al., 2021; Ruchai et al., 2021; Wang and Lee, 2021; Bahmei et al., 2022; She et al., 2022).

CNNs have found wide-ranging applications in specific domains, including medical imaging, wherein they have demonstrated promising outcomes in critical tasks such as disease diagnosis, tumor detection, and lesion segmentation. Their ability to extract intricate visual features has enabled significant advancements in the field of medical diagnostics. Moreover, CNNs have made substantial contributions to areas like robotics, autonomous driving, and other domains that necessitate real-time visual processing. By leveraging CNNs, researchers and practitioners have been able to enhance the perception and decision-making capabilities of intelligent systems, enabling them to operate effectively and autonomously in dynamic environments. The utilization of CNNs in these domains showcases their versatility and efficacy in addressing complex visual challenges and underscores their potential to revolutionize various fields reliant on real-time visual analysis (Pham and Jeon, 2017; Aoki et al., 2018; Kocic et al., 2019; Lim et al., 2020; Lozhkin et al., 2021; Li et al., 2022; Roostaiyan et al., 2022).

Overall, Convolutional Neural Networks (CNNs) have emerged as a fundamental tool in the field of computer vision and remain an active area of ongoing research and development. Their exceptional performance in various visual tasks has solidified their importance and relevance.

In the context of this study, our objective is to conduct a comprehensive analysis of the four most renowned models utilized for image classification, particularly focusing on their applicability in classifying images of native Chilean flora. In the subsequent sections, we will provide detailed explanations and insights into these selected models.

2.1.1 ResNet152

ResNet152 is a convolutional neural network architecture with deep layers that was developed by researchers at Microsoft in 2015. The architecture introduces a novel concept called residual learning, wherein the network is trained to learn residual functions instead of directly mapping the input to the output. This innovative approach enables the network to become significantly deeper than previous architectures, while still achieving remarkable performance (He et al., 2016).

ResNet152 is a convolutional neural network architecture that consists of 152 layers. It utilizes skip connections, also known as residual connections, to facilitate the flow of gradients during backpropagation. These skip connections allow the network to simultaneously learn both low-level and highlevel features, making it highly effective for image recognition tasks.

Additionally, ResNet152 incorporates batch normalization, a technique that helps mitigate overfitting and accelerates the training process. By normalizing the activations within each batch, batch normalization enhances the network's stability and enables more efficient learning.

ResNet152 has demonstrated outstanding performance on prominent image recognition benchmarks, including ImageNet, which comprises over a million images across 1,000 classes. Its exceptional results have made it a state-of-the-art model in the field. Moreover, ResNet152 has found applications in diverse domains, such as object detection, image segmentation, and face recognition, showcasing its versatility and effectiveness across multiple tasks.

2.1.2 VGG19

VGG19 is a convolutional neural network (CNN) model proposed by the Visual Geometry Group (VGG) at the University of Oxford in 2014. It is characterized by its architecture, consisting of 19 layers comprising convolutional and pooling layers, followed by three fully connected layers.

One of the notable aspects of VGG19 is its extensive use of small 3x3 convolutional filters across the network. This design choice allows the model to effectively capture local features and their combinations, enhancing its ability to generalize well to new images. Moreover, the depth of the VGG19 architecture enables it to learn increasingly complex features as the network goes deeper.

VGG19 is commonly employed as a pre-trained model for transfer learning in various computer vision tasks. Pretrained models, such as VGG19, have already learned weights from large-scale datasets like ImageNet. Leveraging these pretrained weights as

a starting point can greatly benefit new tasks that involve smaller datasets, as it helps accelerate the learning process and improve performance.

By utilizing VGG19 as a pre-trained model, researchers and practitioners can leverage the knowledge and representations learned from vast image datasets, enabling them to tackle new visual recognition problems more effectively (Simonyan and Zisserman, 2014).

2.1.3 InceptionV3

InceptionV3 is a convolutional neural network (CNN) architecture that was introduced in 2015 by researchers at Google. It is a deep neural network with 48 layers, and it was designed specifically for image recognition and classification tasks.

InceptionV3 uses a unique module called an "Inception module" that is able to perform multiple convolutions and pooling operations at different scales in parallel. This allows the network to capture both local and global features in the image, making it more accurate at recognizing complex patterns.

The network was trained on the ImageNet dataset, which is a large dataset of over 14 million images. During training, the network learned to classify images into one of 1,000 different categories, such as "dog", "cat", or "car".

InceptionV3 has been used in many applications, including object recognition, facial recognition, and medical image analysis. Its high accuracy and ability to handle complex images make it a popular choice for deep learning practitioners (Szegedy et al., 2015).

2.1.4 MobileNetV2

MobileNetV2 is an architecture of convolutional neural network (CNN) introduced in 2018 as an enhancement to the original MobileNet model. It addresses the need for a lightweight network that can deliver high accuracy in image classification tasks while minimizing computational requirements.

The primary concept behind MobileNetV2 involves utilizing a combination of depthwise separable convolutions and linear bottlenecks. Depthwise separable convolutions involve breaking down the standard convolution into two distinct layers: a depthwise convolution and a pointwise convolution. The depthwise convolution applies individual filters to each input channel, while the pointwise convolution merges the outputs from the depthwise convolution through a linear transformation. By doing so, the computational complexity of the convolution operation is reduced while preserving accuracy.

MobileNetV2 also incorporates linear bottlenecks as a significant element. These bottlenecks serve to diminish the dimensionality of the feature maps while retaining the maximum amount of information. They accomplish this by applying a linear transformation to the feature maps, followed by passing them through an activation function.

Overall, MobileNetV2 stands as an extremely efficient and precise CNN architecture, ideally suited for scenarios with limited resources such as mobile devices and embedded systems. Its utilization of depthwise separable convolutions and linear bottlenecks enables it to strike a balance between computational efficiency and accuracy, making it a valuable choice in resource-constrained environments (Sandler et al., 2018).

2.2 Comparison convolutional neural network architecture

InceptionV3, ResNet152, VGG19, and MobileNetV2 are all popular convolutional neural network (CNN) models used in image classification tasks.

InceptionV3 was introduced by Google in 2015 and is a deep CNN with 48 layers. It uses a unique architecture of Inception modules, which are multi-branch convolutional blocks that allow the network to learn both spatial features and channel-wise correlations at different scales. InceptionV3 is known for its high accuracy and is often used in complex image recognition tasks.

ResNet152 is a residual network introduced by Microsoft in 2016. It is a very deep CNN model with 152 layers that uses residual connections to address the problem of vanishing gradients, which can occur in very deep networks. These connections allow the gradient to flow through the network more easily, which improves training and accuracy. ResNet152 has achieved state-of-the-art performance in many image recognition tasks.

VGG19 is a CNN model introduced by the Visual Geometry Group (VGG) at the University of Oxford in 2014. It has 19 layers and uses a simple architecture of repeated convolutional layers followed by max pooling and fully connected layers. VGG19 is known for its simplicity and ease of implementation, and it has achieved high accuracy in many image recognition tasks.

MobileNetV2 is a CNN model introduced by Google in 2018. It is designed for mobile and embedded devices and has a small footprint and low computational cost. It uses depthwise separable convolutions to reduce the number of parameters and computational complexity while maintaining high accuracy. MobileNetV2 is often used in real-time image recognition applications on mobile devices.

When comparing these four CNN models, it is important to consider the specific requirements of the image classification task at hand. In general, InceptionV3 and ResNet152 are more suitable for complex and high-accuracy tasks, while VGG19 and MobileNetV2 are more suitable for simpler tasks with less computational resources available.

2.3 Classification models for flora images

The classification of flora images has experienced notable advancements in recent years, primarily due to the progress made in deep learning techniques, notably convolutional neural networks (CNNs) (Lopez-Jimenez et al., 2019; Ibrahim et al., 2022). State-of-the-art models for flora image classification often employ pre-trained CNNs, which demonstrate the ability to accurately recognize intricate image patterns. Furthermore, transfer learning,

which involves finetuning pre-trained CNNs using new datasets, has proven to be an effective approach in enhancing the accuracy of flora image classification.

Several widely used CNN models have been successfully employed in flora image classification, including InceptionV3, ResNet152, VGG19, and MobileNetV2. Recent research studies have demonstrated the high accuracy rates achieved by these models in the classification of flora images (Kattenborn et al., 2020). For instance, a study focusing on the classification of Brazilian flora images using deep learning models reported classification accuracies reaching up to 93% for InceptionV3 and ResNet152 (Figueroa-Mata et al., 2022).

In order to overcome the challenges arising from limited labeled data in flora image classification, researchers have explored various techniques, including weakly-supervised and semi-supervised learning methods (Heredia, 2017). Moreover, recent studies have concentrated on enhancing accuracy by incorporating additional data sources, such as spectral and hyperspectral information, and leveraging more advanced CNN architectures (Lazarescu et al., 2004; Singh et al., 2009; Wang et al., 2009).

Overall, the state-of-the-art of classification models for flora images is constantly evolving, and we anticipate further advancements in the near future. The selection of an appropriate model may vary depending on the unique characteristics of the dataset and the specific requirements of the classification task at hand (Sulc and Matas, 2017; Hasan et al., 2020; Ball, 2021; Filgueiras, 2022).

3 Proposed method

The objective of this paper is to conduct a comparative analysis of various deep learning (DL) models for the classification of native Chilean flora images. The models under evaluation encompass InceptionV3, VGG19, ResNet152, and MobileNetV2, all of which have been pre-trained using the Imagenet dataset.

3.1 Data collection

In the process of data collection, the primary source of images was the internet, with careful consideration given to certain limitations and requirements to ensure the preservation of the classes and to obtain a high-quality dataset. To ensure accuracy and avoid interference from external factors such as other flowers, trees, animals, or plants, images were selected based on specific criteria. These criteria included minimal noise, a predominant focus on native species, and the absence of elements that could hinder species identification (refer to Figure 1). Furthermore, only real photographs were included, while illustrations of the species were intentionally excluded. Images that contained watermarks or copyright protection preventing their usage, even for non-commercial purposes, were also excluded from the dataset.

The creation of the dataset primarily relied on specialized websites dedicated to documenting flora, such as iNaturalist, Fundacion´ RA Philippi, and Chilebosque. These sources were selected for their comprehensive coverage of native Chilean flora. Additionally, websites



FIGURE 1
Some samples of chosen and rejected images.

focused on general photography, including Pinterest, Flickr, and Alamy, were utilized as secondary sources to enhance the dataset. In cases where the initial results from specialized websites were insufficient in terms of capturing the required flora species, the dataset was augmented with the best results obtained from Google Images while ensuring compliance with the aforementioned specifications and criteria.

The primary objective of this study is to compare the performance of different convolutional neural network (CNN) models, with a focus on analyzing their classification capabilities for Chilean native flora. As such, the resolution of each image used in the study did not significantly impact the CNN models' classification task. This can be attributed to the following reasons: (i) *Focus on model comparison*: The main objective of this study is to compare the performance of different convolutional neural network (CNN) models. The focus is on analyzing how these models perform in classifying Chilean native flora, regardless of the resolution of the images used. Therefore, the resolution of the images is not a critical factor for evaluating the models' ability to perform classification. (ii) *Adaptation of images to specific input values*: Regardless of the resolution of each image, all images need to be adapted to the specific input value required by each CNN model. This means that all images are resized to the desired input size, ensuring that all images are processed uniformly, regardless of their original resolution. (iii) *Variability in image quality on the web*: Images collected from different sources on the internet can have a wide range of resolutions and visual quality. Setting a specific resolution standard for all collected images would be impractical and could restrict the diversity and representativeness of the image sample used in the study.

Accordingly, all images are resized to the required input size, and the focus is on evaluating the models' ability to perform classification effectively. The details of the selected images are in the Table 1.

TABLE 1 The details of the selected images.

Website	# Selected Images	# Discarded	Total Images
iNaturalist	1000	320	1320
RA Philippi	500	250	750
Chilebosque	500	250	750
Pinterest	1500	850	2350
Flickr	1000	320	1320
Alamy	500	150	650
Total Images	5000	2140	7000

3.2 Native Chilean flora species selected

The chosen flora species are geographically distributed across the country, representing at least one species for each of the main regions (North, Central, and South). Although there are several variations within each species, the selected classes primarily consist of widely recognized varieties, excluding those with minor differences in color tones of the petals or other negligible variations (e.g., number of petals or small spots). Additional details regarding the selected images can be found in the provided Table 2.

Our selection of flora species was carefully chosen to represent a diverse range of regions in Chile see Figure 2 while also presenting specific shared traits to add complexity to the classification task. One notable common characteristic among these selected species is the presence of yellow or orange pollen stamens, accompanied by pointed petals rather than rounded ones. The Copihue, Ana˜ nuca,˜ and Chilco classes present a unique challenge in the classification process due to their vibrant red coloration and the potential for their shapes to appear remarkably similar from different angles. These characteristics add complexity to the task of accurately distinguishing and classifying these species (refer to Figure 3 for visual reference).

Finally, the dataset was divided into two distinct sets: the training set, which comprised 80% of the data, and the testing set, which encompassed the remaining 20%. The decision regarding this split was made based on a careful consideration of various factors,

TABLE 2 Detail our dataset of mages of various species of native Chilean flora.

# Clase	Specie Selected	# Total Images	Region
1	Copihue (Lapageria Rosea)	500	South
2	Chilco (Fuchsia magellanica)	500	South
3	Ana˜ nuca˜ de Fuego (Phycella cyrtanthoides)	500	Central
4	Azulillo (Pasithea caerulea)	500	North
5	Chagual (Puya alpestris)	500	Central
6	Maqui (Aristotelia chilensis)	500	Central/South
7	Lingue (Persea lingue)	500	Central/South
8	Canelo (Drimys winteri)	500	North
9	Quila (Chusquea quila)	500	South
10	Notro (Embothrium coccineum)	500	Central

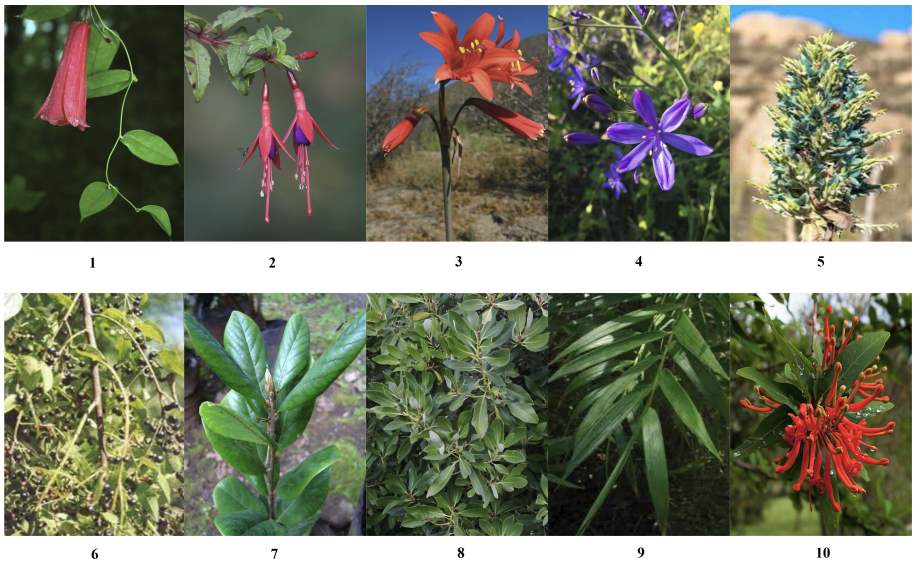


FIGURE 2
Sample of images depicting ten distinct species of Chilean native flora.

such as the size of the dataset, the desired balance between training and testing data, and the need to ensure a representative sample for evaluating the performance of the classification models (Mohanty et al., 2016).

3.3 Data training

During this phase, we initiated the training process by utilizing the ImageNet dataset (Deng et al., 2009), which consists of 1.2 million images distributed across 1000 categories. This served as a starting point to initialize the weights of our convolutional neural networks (CNNs) before fine-tuning them with our specific dataset of native Chilean flora.

To accomplish this, we employed Transfer Learning, a technique that allows the transfer of knowledge from one or more domains to a different domain with a distinct task. In our case, we fine-tuned the pre-trained models on our native Chilean flora dataset. This involved replacing the pre-trained output layer with a new layer that matched the number of classes in our dataset. Consequently, the last three layers of the pre-trained model, which included a fully-connected layer, a softmax layer, and a classification output layer, were substituted.

By utilizing pre-trained CNN models, we benefited from faster and more efficient training compared to starting with randomly initialized weights. Furthermore, pre-trained models exhibited lower training error rates in comparison to artificial neural networks (ANNs) that were not pre-trained. We thoroughly assessed the performance of various CNN architectures in addressing the classification task for native Chilean flora (Zizka et al., 2009).

CNN architectures are typically composed of specific elements that vary across different models. Figure 4 presents an overview of the general structure of a CNN, highlighting key components such as the input layer, convolutional layer, pooling layer, and flattening process. The output of the flattening process is then passed through a series of dense layers, culminating in the final output layer.

Therefore, the characteristics of the architectures used are described in the Table 3. It is important to highlight that (CNN).



FIGURE 3
Comparison of similarities between Copihue and Añañuca.

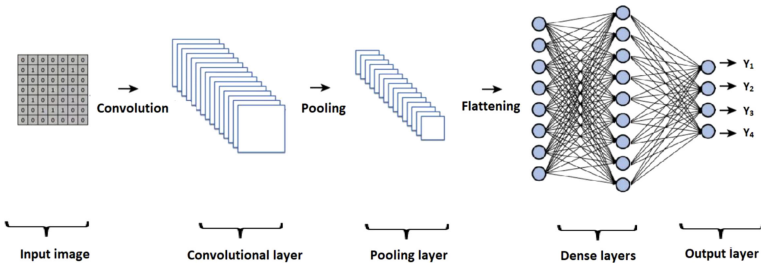


FIGURE 4
Representation of the architecture of a convolutional neural network (CNN).

The InceptionV3 model utilizes a convolutional neural network (CNN) architecture that requires a larger input image size compared to other CNN models, specifically 299x299 pixels. This distinctive image size for InceptionV3 is specifically optimized for the tasks and datasets on which it was trained. It represents a careful balance between maximizing performance on those specific tasks and efficiently managing computational resources.

The choice of a larger input image size, such as 299x299, in InceptionV3 offers several advantages. Firstly, it allows the model to capture more fine-grained details and intricate features within the input images, potentially enhancing its ability to recognize complex patterns. Secondly, the larger image size enables the network to effectively handle a wider range of object scales, accommodating both small and large objects within the same image.

It is essential to note that the specific image size of 299x299 for InceptionV3 is a deliberate design decision based on empirical evaluations and experimentation conducted during its development. This optimization aims to ensure that InceptionV3 performs optimally for the given tasks and datasets it was trained on, providing a balance between accuracy and computational efficiency.

In order to ensure a fair comparison between the experiments, we made an effort to standardize the hyperparameters across all the experiments. The specific hyperparameters used in our experiments are detailed in Table 4. The inclusion of these hyperparameters was essential for optimizing the performance of the deep learning models.

Hyperparameters play a crucial role in controlling different aspects of the training process, including the learning rate, momentum, batch size, and others. By carefully tuning these hyperparameters, our objective was to find the optimal configuration that would facilitate better convergence and improved accuracy of the models.

Standardizing the hyperparameters allowed us to establish a consistent framework for evaluating and comparing the performance of the different CNN architectures. It also ensured that any observed differences in performance were primarily attributed to the architectural variations rather than the hyperparameter settings.

We believe that by employing standardized hyperparameters, we have fostered a more reliable and meaningful comparison between the models, enabling us to draw robust conclusions regarding their relative performance in classifying the images of native Chilean flora.

Deep learning (DL) has revolutionized many research areas. Among optimization algorithms, Stochastic Gradient Descent with Momemtum (SGDM) has emerged as the most widely used due to its balance between accuracy and efficiency (Kleinberg et al., 2018). SGDM is simple and effective, but requires careful tuning of hyperparameters, particularly the initial learning rate, which determines the rate at which weights are adjusted to obtain a local or global minimum of the loss function. Momentum is used to accelerate SGDM in the appropriate direction and reduce oscillations (Ruder, 2016). Regularization is also important to prevent overfitting, with L2 Regularization being the most common type. Combined with SGDM, it results in weight decay, in which the weights are scaled by a factor slightly smaller than one at each update (Van Laarhoven, 2017). To train our models, we used 30 epochs, based on the findings of Mohanty et al. (2016), who reported consistently converging results after the first learning rate decrease. Finally, all CNNs were trained using a batch size of 32.

Training these CNN architectures is extremely computationally intensive. Therefore, all the experiments are carried out on a workstation, presenting the details summarized in Table 5.

TABLE 3 Summary of the utilized architectures.

Network	Depth	Parameters (Millions)	Image Input Size
VGG19	19	143	224 x 224
ResNET152	152	60	224 x 224
InceptionV3	48	23.9	299 x 299
MobileNetV2	53	3	224 x 224

TABLE 4 Hyper-parameters of the experiments.

Hyper-Parameters	Value
Optimization algorithm	SGDM
Momentum	0.9000
Initial learning rate	$1x10^{-3}$
L2 Regularization	$1x10^{-4}$
Epochs	30
Batch size	32

Stochastic gradient descent with momentum (SGDM).

3.4 Evaluation

The proposed method's performance is evaluated by comparing pre-trained models using different metrics. The quality of learning algorithms is commonly assessed by how well they perform on test data (Japkowicz and Shah, 2011). One of the metrics used is the Receiver Operating Characteristic Curve (ROC), which is also known as the Area Under the Curve (AUC). The AUC is a widely used performance measure for supervised classification tasks, and it is based on the relationship between sensitivity and specificity (Hanley and McNeil, 1982).

In this work, we used a generalized version of AUC for multiple classes, as defined by Hand and Till (2001). This function calculates the multiclass AUC by taking the mean of several AUC values 1. To use this function, a data frame is passed as a predictor, and the columns must be named according to the levels of the response.

$$AUC = \frac{1}{C(C-1)} \sum_{i=1}^{C} \sum_{j \neq i}^{C} AUC_{ij} \tag{1}$$

where C is the number of class in the multiclass problem, AUC_ij presents the binary AUC between i and class j.

Sensitivity or recall corresponds to the accuracy of positive examples and indicates how many positive class examples were correctly labeled. This can be calculated using Equation 2, where TP represents the true positives, which are the number of positive instances correctly identified, and FN represents the false negatives, which are the number of positive cases as negative.

$$Sensitivity(\text{Re}\, call) = \frac{TP}{TP + FN} \tag{2}$$

Specificity is a measure of the conditional probability of true negatives given a secondary class, which approximates the probability of the negative label being true. It can be calculated using Equation 3, where TN represents the number of true negatives, i.e., the negative cases that are correctly classified as negative, and FP represents the number of false positives, i.e., the negative instances that are incorrectly classified as positive cases.

$$Specificity = \frac{TN}{TN + FN} \tag{3}$$

To evaluate the overall classification performance, accuracy is the most commonly used metric. During the evaluation stage, accuracy was calculated every 20 iterations. This metric calculates the percentage of samples that are correctly classified, and it is represented by Equation 4:

TABLE 5 Hyper-parameters of the experiments.

Hardware and Software	Characteristics
Memory	16 Gb
Processor	Intel Core i7-7700 CPU @ 3.60 GHz
Graphics	GeForce GTX 1070 X 8 Gb
Operating Systema	Windows 10, 64 bits

Stochastic gradient descent with momentum (SGDM).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Precision is an important metric that evaluates the correctness of a model by measuring the number of true positives divided by the sum of true positives and false positives. In other words, precision measures how many of the predicted positive cases are actually positive, and it assesses the predictive power of the algorithm. The precision score is calculated using Equation 5

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

The F-score is a metric that combines precision and recall, and is defined as the harmonic mean of the two, as shown in Equation 6. It is a measure that focuses on the analysis of the positive class, and a high value of this metric indicates that the model performs better on the positive class.

$$F - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6}$$

Finally, the data was separated into two sets, containing 80% of the data in the training set and the remaining 20% in the testing set. The choice of the split is based on (Mohanty et al., 2016).

3.5 Results

In this study, we evaluated the performance of state-of-theart pre-trained models for the classification of Native Chilean flora. The main objective of this research was to compare the CNN models and assess their accuracy, precision, sensitivity, specificity, F-Score, and AUC through fine-tuning. The results of this evaluation are presented in Table 6.

All models demonstrated similar and statistically significant performance. In terms of AUC, VGG19 and MobileNetV2 yielded the lowest results at 90.81% and 92.89%, respectively, followed by InceptionV3 with 94.82%. The highest AUC result was achieved by ResNet152 with 96.02%, indicating excellent classification. Conversely, VGG19 exhibited the lowest precision metric result at 90.81%, with MobileNetV2 and InceptionV3 following at 92.73% and 95.01%, respectively.

The highest precision was achieved again by ResNet152 at 95.73%. In measures of sensitivity, specificity, and F-score, VGG19 showed poor performance at 90.48%, 91.07%, and 90.77%, respectively. In contrast, ResNet152 had the highest percentage in all previous metrics at 95.23%, 95.87%, and 95.85%, respectively. While all models had statistically significant performance, ResNet152 achieved the highest percentage. Additionally, considering the processing time required by each convolutional neural network (CNN) for the classification task, MobileNetv2 exhibited the best performance with the shortest processing time. This indicates a higher level of efficiency compared to the other CNN architectures. It is worth noting that although MobileNetv2 showed a slight disadvantage in terms of measurement statistics, the difference was not significant when compared to the results obtained by other CNNs, such as InceptionV3, which had the longest processing time. One possible

TABLE 6 Performance measures (%) for every pre-trained model.

Measures	VGG19	ResNet152	InceptionV3	MobileNetV2
Accuracy	90.29	95.87	94.98	92.45
Precision	90.81	95.73	95.01	92.74
Sensitivity	90.48	95.23	94.38	92.68
Specificity	91.02	95.87	94.85	93.01
F-Score	90.77	95.85	94.62	93.14
AUC	90.81	96.02	94.82	92.89
Time (milliseconds)	182.12	342.12	652.3	150.23

explanation for this observation is that InceptionV3 has a deeper and more complex architecture compared to the other models. Deeper and more complex architectures typically incur a higher computational load, resulting in longer inference times. Thus, it could be considered to accept a minimal decrease in precision in exchange for improved processing efficiency.

Additionally, Figure 5 presents the confusion matrix, which visually represents the performance of the classifiers and highlights the classes distinguished by all models used in this study. Each row corresponds to the predicted class, while each column corresponds to the true class. The cells on the diagonal represent correctly classified observations, while the off-diagonal cells indicate misclassifications.

The ResNet152 model has demonstrated superior performance in classifying native Chilean flora compared to the VGG19, InceptionV3, and MobileNet models. This can be attributed to its deep architecture, which enables it to effectively capture complex features and fine details in images.

The ResNet152 architecture utilizes a deep neural network structure with residual layers, allowing it to learn more intricate representations of visual characteristics in plants. This enhanced representation capability enables the ResNet152 model to accurately capture and distinguish the subtle variations and differences among species of native Chilean flora, resulting in higher classification accuracy compared to VGG19, InceptionV3, and MobileNet.

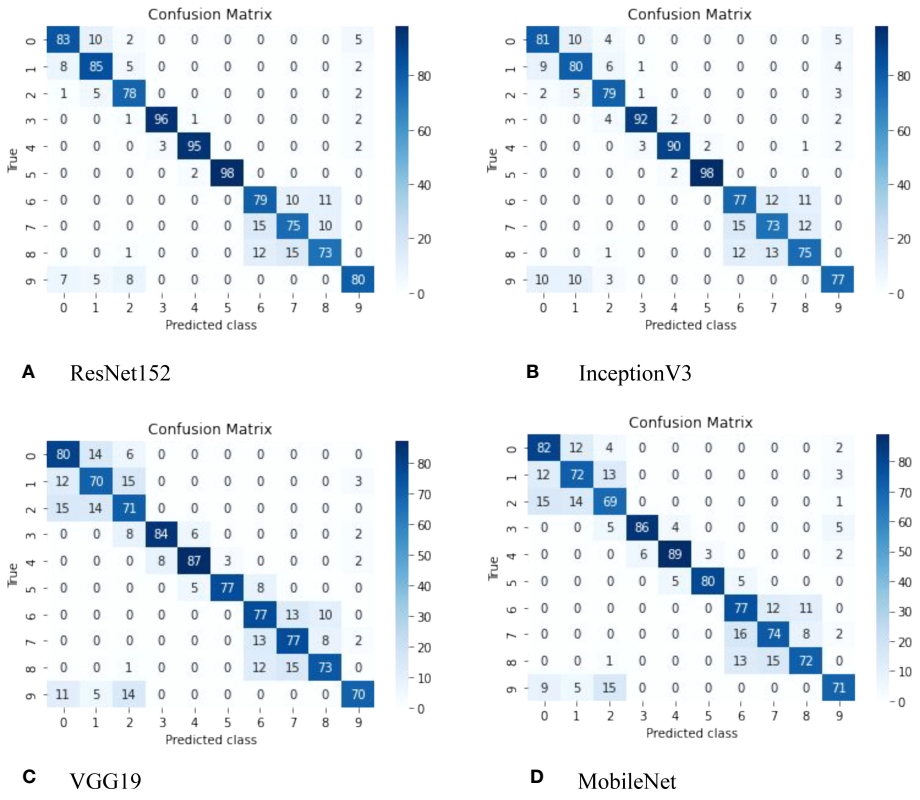


FIGURE 5
Confusion matrix derived from the ResNet152 model, featuring the following classes: (1) Copihue; (2) Chilco; (3) Añañuca de Fuego; (4) Azulillo; (5) Chagual; (6) Maqui; (7) Lingue; (8) Canelo; (9) Quila; and (10) Notro.

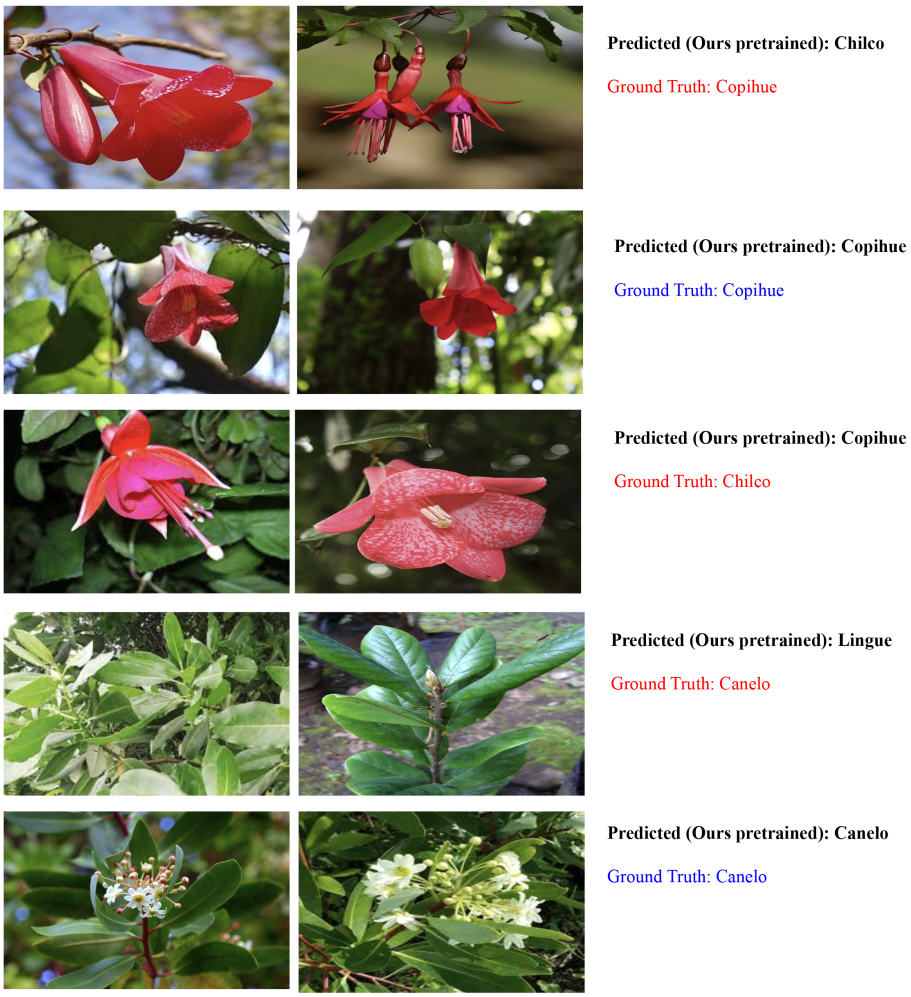


FIGURE 6
Examples of correct and incorrect predictions on our dataset based on ResNet152.

On the other hand, MobileNet stands out for its computational efficiency and processing speed. It achieves this through the use of lighter convolution operations and parameter reduction techniques, resulting in a more lightweight and faster architecture. Although MobileNet may have slightly lower accuracy compared to ResNet152, its processing speed is significantly faster.

The choice between ResNet152 and MobileNet depends on the specific requirements of the application scenario. If achieving the highest accuracy is of utmost importance, and processing time is of secondary concern, ResNet152 would be the preferred choice due to its superior performance in classifying native Chilean flora. However, if reducing processing time is critical, and a slight decrease in accuracy can be tolerated, MobileNet may be the more suitable option due to its computational efficiency and faster processing speed.

Finally, the ability of the ResNet152 model to accurately or inaccurately predict images of Copihue with Chilco and Canelo with Lingue can be attributed to several factors. The following are some possible explanations:(i) Visual similarity: The ResNet152 model has been trained to recognize and distinguish specific visual features of different flora classes. However, it is possible that images of Copihue and Chilco, as well as those of Canelo and Lingue, share similar visual characteristics. These similarities can lead to confusion in the model, resulting in both correct and incorrect predictions. (ii) Intraspecific variability: Within the same species, such as Copihue and Chilco, or Canelo and Lingue, there may be variations in the appearance and characteristics of individual plants. These variations can pose challenges for precise classification, as the model may encounter examples that exhibit atypical or unusual features within the species. In some cases, the model may adapt correctly to these variations and make accurate predictions, while in other cases, it may become confused and make erroneous predictions. (iii) Quality and diversity of the training dataset: The performance of the ResNet152 model heavily relies on the quality and diversity of the training dataset. If the dataset contains a wide variety of images of Copihue, Chilco, Canelo, and Lingue, capturing different variations and characteristics of each species, it is more likely that the model can make accurate predictions. However, if the dataset is limited in terms of species representativeness or does not adequately cover the intraspecific variability, the model may struggle to make precise predictions in all cases.

It is important to note that the performance of the model can be improved through additional techniques such as fine-tuning and

optimization of hyperparameters, collecting more representative training data, and including images that encompass a greater variety of species characteristics. These approaches can help reduce prediction errors and enhance the model's ability to accurately distinguish between Copihue and Chilco, as well as between Canelo and Lingue see Figure 6.

3.6 Conclusions

In conclusion, this study focused on the creation of a dataset consisting of images of native Chilean flora and the subsequent comparative analysis of different convolutional neural network (CNN) models. The dataset aimed to provide a comprehensive representation of the diverse flora found in Chile, capturing the variations and characteristics of different species.

Through the comparative study, we evaluated the performance of four CNN models: ResNet152, VGG19, InceptionV3, and MobileNet. Our findings indicate that the ResNet152 model exhibited superior performance in classifying native Chilean flora compared to the other models. This can be attributed to its deep architecture, which enabled the model to capture complex features and fine details in images more effectively. The ResNet152 model's ability to accurately distinguish between species contributed to its higher classification accuracy.

However, it is worth noting that the MobileNet model showcased exceptional computational efficiency and processing speed. While it may have slightly lower accuracy compared to ResNet152, MobileNet's faster processing speed makes it a suitable choice for scenarios where reducing processing time is crucial and a slight compromise in accuracy can be tolerated.

The study highlighted the importance of the quality and diversity of the training dataset in achieving accurate predictions. Additionally, factors such as visual similarity and intraspecific variability within species were identified as potential challenges in classification tasks.

Overall, this study provides valuable insights into the classification of native Chilean flora using CNN models. The findings can contribute to the development of more accurate and efficient systems for flora recognition and classification, with potential applications in biodiversity conservation, ecological research, and environmental monitoring. Further research can explore advanced techniques to enhance the performance of CNN models and expand the dataset to encompass a broader range of native plant species.

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

CF-F: Conceived and designed the study, performed data collection and analysis, and contributed to the writing of the paper, provided expertise on the subject matter, reviewed and provided feedback on the research methodology and findings, and contributed to the overall manuscript development. PS-M: Assisted with data analysis and interpretation, provided critical feedback on the manuscript. All authors contributed to the article and approved the submitted version.

Acknowledgments

We thank for the support from FOVI21001 "Fomento a la Vinculación Internacional para Instituciones de Investigación Regionales (ANID, Chile)" Agencia Nacional de Investigacion´ y Desarrollo and ALBA Research Group (Algorithms and Database) 2130591 GI/VC, "Ayudantes para el Fortalecimiento de Investigacin FACE 2022 " and "Proyecto de Reinsercion:´ DIUBB 2230508 IF/RS of the University of Bío Bío.

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

Aoki, R., Imamura, K., HIrano, A., and Matsuda, Y. (2018). High-performance super-resolution *via* patch-based deep neural network for real-time implementation. *IEICE Trans. Inf. Syst.* E101D (11), 2808–2817. doi: 10.1587/transinf.2018EDP7081

Ashraf, S., Kadery, I., Chowdhury, M. A. A., Mahbub, T. Z., and Rahman, R. M. (2019). Fruit image classification using convolutional neural networks. *Int. J. Softw. Innovation* 7 (4), 51–70. doi: 10.4018/IJSI.2019100103

Bahmei, B., Birmingham, E., and Arzanpour, S. (2022). Cnn-rnn and data augmentation using deep convolutional generative adversarial network for environmental sound classification. *IEEE Signal Process. Lett.* 29, 682–686. doi: 10.1109/LSP.2022.3150258

Ball, J. (2021). Few-shot learning for image classification of common flora. *arXiv:2105.03056*.

Carranza, D. M., Varas-Belemmi, K., De Veer, D., Iglesias-Muller, C., Coral-Santacruz, D., and Mendez, F. A. (2020). Socio-environmental conflicts: An underestimated threat to biodiversity conservation in Chile. *Environ. Sci. Policy* 110, 46–59. doi: 10.1016/j.envsci.2020.04.006

Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., and Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. *Remote Sens.* 13 (22), 4712. doi: 10.3390/rs13224712

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. 248–255 (IEEE).

Figueroa-Mata, G., Mata-Montero, E., Valverde-Otarola, J. C., AriasAguilar, D., and Zamora-Villalobos, N. (2022). Using deep learning to identify Costa Rican native tree species from wood cut images. *Front. Plant Sci.* 13, 211. doi: 10.3389/fpls.2022.789227

Filgueiras, A. C. S. (2022). *Floralens: a deep learning model for portuguese flora* [Master's Thesis]. da Universidade do Porto..

Hand, D. J., and Till, R. J. (2001). A simple generalisation of the area under the roc curve for multiple class classification problems. *Mach. Learn.* 45, 171–186. doi: 10.1023/A:1010920819831

Hanley, J. A., and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology* 143 (1), 29–36. doi: 10.1148/radiology.143.1.7063747

Hasan, R. I., Yusuf, S. M., and Alzubaidi, L. (2020). Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion. *Plants* 9 (10), 1302. doi: 10.3390/plants9101302

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

Heredia, I. (2017). "Large-scale plant classification with deep neural networks," in *Proceedings of the Computing Frontiers Conference*. 259–262.

Hui, R., Chen, J., Liu, Y., Shi, L., Fu, C., and Ishsay, O. (2020). Classification of deep convolutional neural network in thyroid ultrasound images. *J. Med. Imaging Health Inf.* 10 (8), 1943–1948. doi: 10.1166/jmihi.2020.3099

Ibrahim, N. M., Gabr, D. G. I., Rahman, A.-U., Dash, S., and Nayyar, A. (2022). A deep learning approach to intelligent fruit identification and family classification. *Multimedia Tools Appl.* 81 (19), 27783–27798. doi: 10.1007/s11042-022-12942-9

Japkowicz, N., and Shah, M. (2011). *Evaluating learning algorithms: a classification perspective* (Cambridge University Press).

Kattenborn, T., Eichel, J., Wiser, S., Burrows, L., Fassnacht, F. E., and Schmidtlein, S. (2020). Convolutional neural networks accurately predict cover fractions of plant species and communities in unmanned aerial vehicle imagery. *Remote Sens. Ecol. Conserv.* 6 (4), 472–486. doi: 10.1002/rse2.146

Kleinberg, B., Li, Y., and Yuan, Y. (2018). "An alternative view: When does sgd escape local minima?," in *International conference on machine learning*. 2698–2707 (PMLR).

Kocic, J., Jovicic, N., and Drndarevic, V. (2019). An end-to-end deep neural network for autonomous driving designed for embedded automotive platforms. *Sensors* 19 (9).

Lazarescu, M. M., Venkatesh, S., and Bui, H. H. (2004). Using multiple windows to track concept drift. *Intelligent Data Anal.* 8 (1), 29–59. doi: 10.3233/IDA-2004-8103

Li, Z., Zhang, J., Li, Y., Zhu, J., Long, S., Xue, D., et al. (2022). Learning feature channel weighting for real-time visual tracking. *IEEE Trans. Image Process.* 31, 2190–2200. doi: 10.1109/TIP.2022.3153170

Lim, Y., Bliesener, Y., Narayanan, S., and Nayak, K. S. (2020). Deblurring for spiral real-time mri using convolutional neural networks. *Magnetic Resonance Med.* 84 (6), 3438–3452. doi: 10.1002/mrm.28393

Lopez-Jimenez, E., Vasquez-Gomez, J. I., Sanchez-Acevedo, M. A., and Herrera-Lozada, J. C. (2019). Columnar cactus recognition in aerial images using a deep learning approach. *Ecol. Inf.* 52, 131–138. doi: 10.1016/j.ecoinf.2019.05.005

Lozhkin, A., Maiorov, K., and Bozek, P. (2021). Convolutional neural networks training for autonomous robotics. *Manage. Syst. Prod. Eng.* 29 (1), 75–79. doi: 10.2478/mspe-2021-0010

Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2017). Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Trans. Geosci. Remote Sens.* 55 (2), 645–657. doi: 10.1109/TGRS.2016.2612821

Mohanty, S. P., Hughes, D. P., and Salathe, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7, 1419. doi: 10.3389/fpls.2016.01419

Pattnaik, G., Shrivastava, V. K., and Parvathi, K. (2021). Tomato pest classification using deep convolutional neural network with transfer learning, fine tuning and scratch learning. *Intelligent Decision Technol.-Netherlands* 15 (3), 433–442. doi: 10.3233/IDT-200192

Pham, C. C., and Jeon, J. W. (2017). Robust object proposals re-ranking for object detection in autonomous driving using convolutional neural networks. *Signal Processing-Image Commun.* 53, 110–122. doi: 10.1016/j.image.2017.02.007

Rodriguez, R., Marticorena, C., Alarcon, D., Baeza, C., Cavieres, L., Finot, V. L., et al. (2018). Catalogo´ de las plantas vasculares de Chile. *Gayana. Botanica* 75 (1), 1–430.

Roostaiyan, S. M., Hosseini, M. M., Kashani, M. M., and Amiri, S. H. (2022). Toward real-time image annotation using marginalized coupled dictionary learning. *J. Real-Time Image Process.* 19 (3), 623–638. doi: 10.1007/s11554-022-01210-6

Ruchai, A. N., Kober, V., Dorofeev, K. A., and Karnaukhov, V. N. (2021). Classification of breast abnorMalities using a deep convolutional neural network and transfer learning. *J. Commun. Technol. Electron.* 66 (6), 778–783. doi: 10.1134/S1064226921060206

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv:1609.04747*.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. (pp. 4510–4520). doi: 10.1109/CVPR.2018.00474

Sarigul, M., Ozyildirim, B. M., and Avci, M. (2019). Differential convolutional neural network. *Neural Networks* 116, 279–287. doi: 10.1016/j.neunet.2019.04.025

Sen, P. C., Hajra, M., and Ghosh, M. (2020). "Supervised classification algorithms in machine learning: A survey and review," in *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018*. 99–111 (Springer).

She, L., Fan, Y., Xu, M., Wang, J., Xue, J., and Ou, J. (2022). Insulator breakage detection utilizing a convolutional neural network ensemble implemented with small sample data augmentation and transfer learning. *IEEE Trans. Power Deliv.* 37 (4), 2787–2796. doi: 10.1109/TPWRD.2021.3116600

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*

Singh, M., Curran, E., and Cunningham, P. (2009). *Active learning for multi-label image annotation* (Technical report, University College Dublin: School of Computer Science and Informatics).

Sulc, M., and Matas, J. (2017). Fine-grained recognition of plants from images. *Plant Methods* 13, 1–14. doi: 10.1186/s13007-017-0265-4

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition 1–9. doi: 10.1109/CVPR.2015.7298594

Van Laarhoven, T. (2017). L2 regularization versus batch and weight norMalization. *arXiv:1706.05350*.

Wang, H., Huang, H., and Ding, C. (2009). "Image annotation using multi-label correlated green's function," in *2009 IEEE 12th International Conference on Computer Vision*. 2029–2034 (IEEE).

Wang, J., and Lee, S. (2021). Data augmentation methods applying grayscale images for convolutional neural networks in machine vision. *Appl. Sci.-Basel* 11 (15). doi: 10.3390/app11156721

Zelenina, L., Khaimina, L., Khaimin, E., Khripunov, D., and Zashikhina, I. (2022). Convolutional neural networks in the task of image classification. *Mathematics Inf.* 65 (1), 19–29. doi: 10.53656/math2022-1-2-con

Zhou, D.-X. (2020). Theory of deep convolutional neural networks: Downsampling. *Neural Networks* 124, 319–327. doi: 10.1016/j.neunet.2020.01.018

Zhu, L., and Chang, W. (2019). Application of deep convolutional neural networks in attention-deficit/hyperactivity disorder classification: Data augmentation and convolutional neural network transfer learning. *J. Med. Imaging Health Inf.* 9 (8), 1717–1724. doi: 10.1166/jmihi.2019.2843

Zizka, G., Schmidt, M., Schulte, K., Novoa, P., Pinto, R., and Konig, K. (2009). Chilean bromeliaceae: diversity, distribution and evaluation of conservation status. *Biodiversity Conserv.* 18, 2449–2471. doi: 10.1007/s10531-009-9601-y

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 18 September 2023
DOI 10.3389/fpls.2023.1219584

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Jiong Mu,
Sichuan Agricultural University, China
Md Nashir Uddin,
The University of Tokyo, Japan

*CORRESPONDENCE
Lejun Yu
yulj@hainanu.edu.cn
Qian Liu
qliu@hainanu.edu.cn

†These authors have contributed equally to this work

RECEIVED 09 May 2023
ACCEPTED 28 August 2023
PUBLISHED 18 September 2023

CITATION
Lu Y, Wang J, Fu L, Yu L and Liu Q (2023) High-throughput and separating-free phenotyping method for on-panicle rice grains based on deep learning. *Front. Plant Sci.* 14:1219584. doi: 10.3389/fpls.2023.1219584

COPYRIGHT
© 2023 Lu, Wang, Fu, Yu 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.

High-throughput and separating-free phenotyping method for on-panicle rice grains based on deep learning

Yuwei Lu[1,2,3†], Jinhu Wang[1†], Ling Fu[2,3,4], Lejun Yu[1*] and Qian Liu[1*]

[1]Key Laboratory of Biomedical Engineering of Hainan Province, School of Biomedical Engineering, Hainan University, Haikou, China, [2]Britton Chance Center for Biomedical Photonics, Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan, Hubei, China, [3]MoE Key Laboratory for Biomedical Photonics, Huazhong University of Science and Technology, Wuhan, Hubei, China, [4]Department of Physics, School of Science, Hainan University, Haikou, China

Rice is a vital food crop that feeds most of the global population. Cultivating high-yielding and superior-quality rice varieties has always been a critical research direction. Rice grain-related traits can be used as crucial phenotypic evidence to assess yield potential and quality. However, the analysis of rice grain traits is still mainly based on manual counting or various seed evaluation devices, which incur high costs in time and money. This study proposed a high-precision phenotyping method for rice panicles based on visible light scanning imaging and deep learning technology, which can achieve high-throughput extraction of critical traits of rice panicles without separating and threshing rice panicles. The imaging of rice panicles was realized through visible light scanning. The grains were detected and segmented using the Faster R-CNN-based model, and an improved Pix2Pix model cascaded with it was used to compensate for the information loss caused by the natural occlusion between the rice grains. An image processing pipeline was designed to calculate fifteen phenotypic traits of the on-panicle rice grains. Eight varieties of rice were used to verify the reliability of this method. The R^2 values between the extraction by the method and manual measurements of the grain number, grain length, grain width, grain length/width ratio and grain perimeter were 0.99, 0.96, 0.83, 0.90 and 0.84, respectively. Their mean absolute percentage error (MAPE) values were 1.65%, 7.15%, 5.76%, 9.13% and 6.51%. The average imaging time of each rice panicle was about 60 seconds, and the total time of data processing and phenotyping traits extraction was less than 10 seconds. By randomly selecting one thousand grains from each of the eight varieties and analyzing traits, it was found that there were certain differences between varieties in the number distribution of thousand-grain length, thousand-grain width, and thousand-grain length/width ratio. The results show that this method is suitable for high-throughput, non-destructive, and high-precision extraction of on-panicle grains traits without separating. Low cost and robust performance make it easy to popularize. The research results will provide new ideas and methods for extracting panicle traits of rice and other crops.

KEYWORDS

rice, rice panicle traits, high-throughput phenotyping, visible light scanning, deep learning

1 Introduction

Rice (*Oryza sativa*) is one of the most important food crops in the world. Ensuring its yield and quality is crucial for food security and social and economic stability in the world (Tester and Langridge, 2010; Yang et al., 2014). At the same time, with the continuous development of the world's socio-economic situation, people's demand for food quality will not decline. Cultivating high-quality rice varieties with high yields has always been an important research direction of rice breeding (Zhang, 2007). The grain trait of rice is one of the most basic and essential rice breeding indexes, which directly reflects the grain yield and quality, including the total grain number, grain length, grain width, length-width ratio, the 1000-grain weight of a rice plant or a panicle and other traits (Li et al., 2019). However, unlike the highly developed genomic tools, the current phenotyping method of rice panicle and grain traits mainly relies on manual counting, which limits the efficiency and accuracy of panicle and grain trait statistics (Crossa et al., 2017; Watt et al., 2020; Sun et al., 2022).

Over the past few decades, researchers commonly obtained traits such as rice grain number and size by threshing and manually measuring (Crossa et al., 2017). This method is inefficient and straightforward to introduce the subjective error of the operator, and the destructive threshing operation will also affect the accuracy of the results (Huang et al., 2013). With the rapid development of computer vision and machine learning technology, automated and high-throughput crop phenotypic techniques based on various imaging techniques and image processing algorithms are gradually becoming essential for obtaining key phenotypic traits (Yang et al., 2020). Many researchers have made a series of beneficial explorations in automatically extracting rice grain-related traits.

The current research on extracting rice panicle-related traits can be divided into two categories from the perspective of pretreatment methods: requiring threshing and not requiring threshing. The method that requires threshing involves the destructive processing of rice panicles with the help of specialized threshing and conveying equipment, ultimately flattening the grains on an imaging platform and automatically calculating the quantity and size characteristics of the grains using optical imaging methods and digital image processing technology (Duan et al., 2011a; Duan et al., 2011b; Huang et al., 2013; Huang et al., 2022). Some works combine scanning tiled rice grains with image processing techniques to analyze rice grains' morphological and color traits (Whan et al., 2014; Wu et al., 2019). An automated analysis software for rice panicle traits based on traditional digital image processing methods has been developed, which can estimate the number of grains on rice panicles with high throughput (Al-Tam et al., 2013). However, those methods require specialized threshing, transmission, and imaging environment, resulting in high image acquisition costs. The threshing process is prone to damage the rice and affect the accuracy of the final results.

In contrast, methods that do not require threshing have higher efficiency and stronger generalization ability. Many studies directly attempt to extract grains and related traits from rice panicle images. The two-dimensional image information of rice panicles can also be used for modeling, a correction-model-referred on-panicle grain counting method was proposed based on the area of the rice panicle and its edge contour wavelet analysis and achieves an average accuracy of 94% compared to the results of manual counting (Gong et al., 2018). The area of the panicle was also used to directly predict yield (Zhao et al., 2019). Deep learning technology has also been widely used in rice counting, positioning, and segmentation (Wu et al., 2019; Deng et al., 2022). Due to the frequent adhesion and occurrence of natural shielding on rice panicles, methods that do not require threshing can usually only obtain quantitative traits of rice but cannot obtain morphological traits. Some researchers have reduced shielding by separating several branches of a single rice panicle and extracting quantitative and morphological traits of rice grains (Gong et al., 2018; Wang et al., 2022). However, separating rice branches is time-consuming and fragile, and it is still impossible to avoid the impact of rice adhesion on the results. Some advanced imaging systems, such as the X-ray imaging system, have also been used to extract rice traits, but high costs and low efficiency limit the promotion of such methods (Su and Chen, 2019; Hu et al., 2020; Yu et al., 2021). Therefore, developing a low-cost phenotyping method for analyzing comprehensive on-panicle rice grain traits with high throughput, high accuracy, and without complex pretreatment is necessary.

This study proposed a high-throughput phenotyping method for extracting on-panicle rice grain traits without grain threshing and branch separating. Color images of individual rice panicles are efficiently obtained based on visible light scanning imaging technology. The cascaded Faster R-CNN model and an improved Pix2Pix model were used to detect, segment, and restore every on-panicle rice grain. Based on the processing results, fifteen rice grain traits are automatically calculated in the designed image processing pipeline.

2 Materials and methods

2.1 Collection of rice panicle

This study randomly selected rice panicles from eight varieties, including three japonica and five indica varieties. Six were planted in Wuhan, Hubei province (30.27°N,114.2°E) and harvested in mid-July 2021. The other two were planted in Sanya, Hainan Province (18.24°E,109.50°E) and harvested in early July 2022. Whether in Wuhan or Sanya, all materials were planted in the same experimental field, using different plots to distinguish different varieties. In terms of field management, both experimental fields followed the conventional field method of maintaining a certain water level throughout the entire growth period. The conditions were strictly identically controlled, except for differences in varieties. Field maintenance, including weeding and pest control, was performed by professionals throughout the growth period. Twenty panicle samples from each variety were randomly selected for imaging and analysis. Specific information for each variety is shown in Table 1.

TABLE 1 Variety information.

Rice variety	Subspecies	Growing area	Harvested time
Jiujiuxinxiang	indica	Sanya	2022.07
Hongxiangyou3	japonica	Sanya	2022.07
Kendao1867	japonica	Wuhan	2021.07
Kenyan1803	japonica	Wuhan	2021.07
Wangdao2	indica	Wuhan	2021.07
Wendao21	indica	Wuhan	2021.07
Z98-308	indica	Wuhan	2021.07
Ganzhi	indica	Wuhan	2021.07

2.2 Visible light scanning imaging method

All images in the experiment were collected by a visible light scanner (Uniscan M1, Tsinghua Unigroup, China). The rice panicles were placed on the scanning panel. A computer with a 64-bit Windows operating system was connected to set the scanning parameters and control the scanner. The scanning mode is a charge-coupled device (CCD). The scanning resolution was 600 dpi, and the image size was 7200 * 10200 pixels. Then crop the image to 5700 * 6800 pixels to remove some background area (Figure 1). According to scanner parameters, each pixel corresponds to an actual size of 0.0423 mm. Under this parameter, the time for single imaging and storing the result is approximately 60 seconds.

2.3 On-panicle rice grain traits extracting algorithm based on deep learning

The on-panicle rice grain extracting algorithm based on visible light scanning imaging results comprises three cascade modules (Figure 2). Firstly, the rice grain detection model detects each grain on the panicle and outputs a region of interest (ROI) local image of the target area. Secondly, the grain occlusion restoration model is used to restore each output result in the upper part to compensate for possible information loss caused by occlusion. Thirdly, the grain trait extraction pipeline is used to calculate rice grain-related phenotyping traits, including one quantitative trait, eight size traits, and six morphological traits.

2.3.1 Rice grain detection model based on Faster R-CNN

Faster R-CNN (Ren et al., 2015) is a convolutional neural network model for target detection tasks proposed by Ren Shaoqing and He Kaiming based on R-CNN (Girshick et al., 2014) and Fast R-CNN (Girshick, 2015). This network skillfully solves the problem of slow training and prediction speed for R-CNN and Fast R-CNN by simultaneously training classification and regression tasks. It proposes a Regional Proposal Network (RPN), which enables the network to conduct end-to-end training. Since its introduction, Faster R-CNN has attracted the attention of many researchers and has been successfully applied in many fields.

The Faster R-CNN mainly includes four parts: feature extraction network, region proposal network (RPN), ROI Pooling module, and classification/regression module. This architecture has good performance in general target detection tasks. Still, in this study, the research targets are small and densely distributed, making it difficult for the original Faster R-CNN network structure to detect grains accurately. Feature pyramid networks (FPN) significantly improve the detection effect of models for small targets by fusing feature maps of different depths (Lin et al., 2017). Therefore, to improve the model's accuracy for detecting grains in the ear, this study incorporated the FPN module into the Faster R-CNN. The overall structure of the designed rice grain detection model is shown in Figure 3.

2.3.2 Rice grain occlusion restoration model based on improved Pix2Pix

Pix2Pix (Isola et al., 2017) is an image translation model based on a conditional generative adversarial network (CGAN) (Mirza

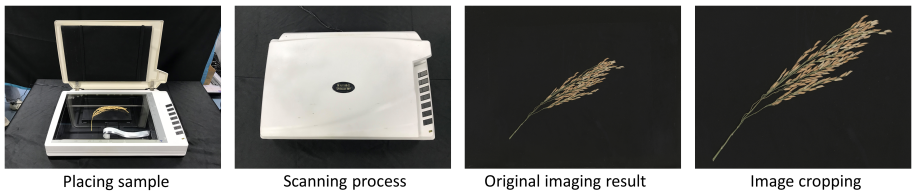


FIGURE 1
Visible light scanning imaging process.

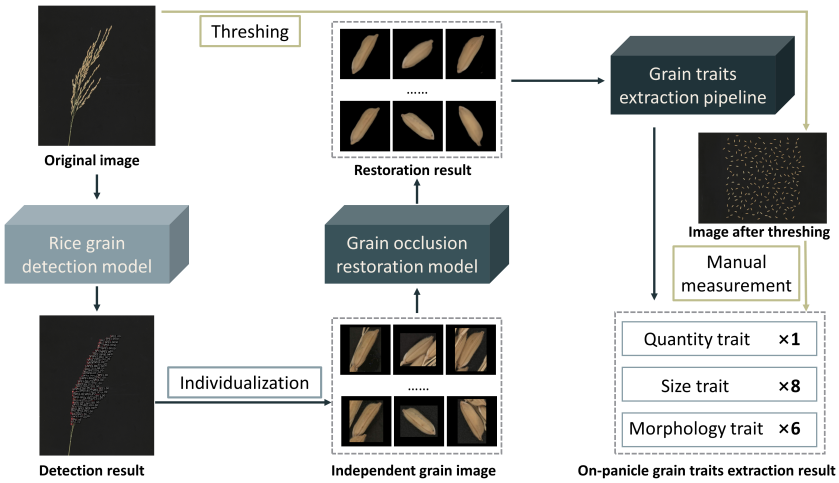


FIGURE 2
Workflow of on-panicle rice grain traits extracting algorithm.

and Osindero, 2014). Pix2Pix learns a mapping between the input and output images by conditioning the input image to obtain the specified output image. The U-Net structure will be adopted as the generator in Pix2Pix (Figure 4B). Different from the traditional encoder-decoder structure, U-Net (Ronneberger et al., 2015) uses skip connections between corresponding encoder and decoder layers to preserve low-level features that may be lost during downsampling. These skip connections concatenate the feature maps from the encoder with those of the corresponding decoder layer, significantly improving image details' reconstruction. The key for Pix2Pix training is the discriminator, which is named PathchGAN. Unlike traditional GlobalGAN discriminator, the output of PatchGAN is not a scalar but an N×N two-dimensional matrix, and each element of this matrix corresponds to a patch in the original image. By discriminating each patch, PatchGAN can provide better feedback to the generator about the local consistency of the generated images.

In the process of grain restoration, the restoration effect of global and local image details will affect the extraction of traits. In order to obtain more accurate grain traits, we need to comprehensively consider the global and local details of the restored grain image. Therefore, GlobalGAN and PatchGAN are fused as the discriminator of the grain occlusion restoration model (Figure 4C).

In summary, the complete grain occlusion restoration model (Figure 4A) mainly includes the following three parts: the generator of the U-Net structure, the discriminator of the fusion of GlobalGAN and PatchGAN, and the CGAN architecture to train the network. The overall structure of the grain occlusion restoration model is shown in Figure 4.

2.3.3 Image-based automatic extraction pipeline for rice grain traits

An automatic extraction pipeline for grain traits was designed to process each rice grain image obtained in the previous step. The automatic extraction pipeline, as shown in Figure 5, was developed with Python language and OpenCV (Bradski, 2000), an open-source image processing toolkit. Firstly, the RGB image of each independent rice grain (Figure 5A) was used as input to the pipeline. Secondly, the red channel (Figure 5B) was extracted

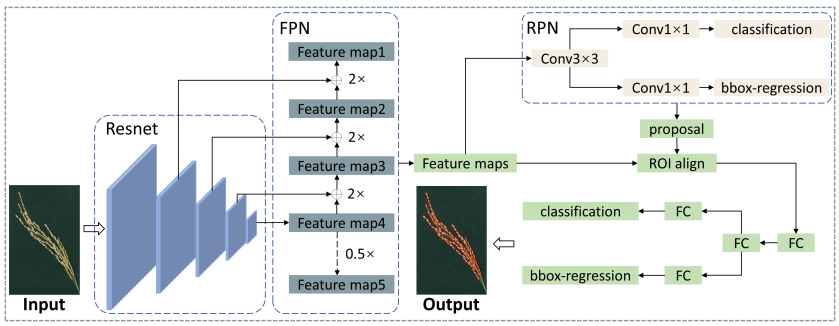


FIGURE 3
Designed rice grain detection model based on Faster R-CNN.

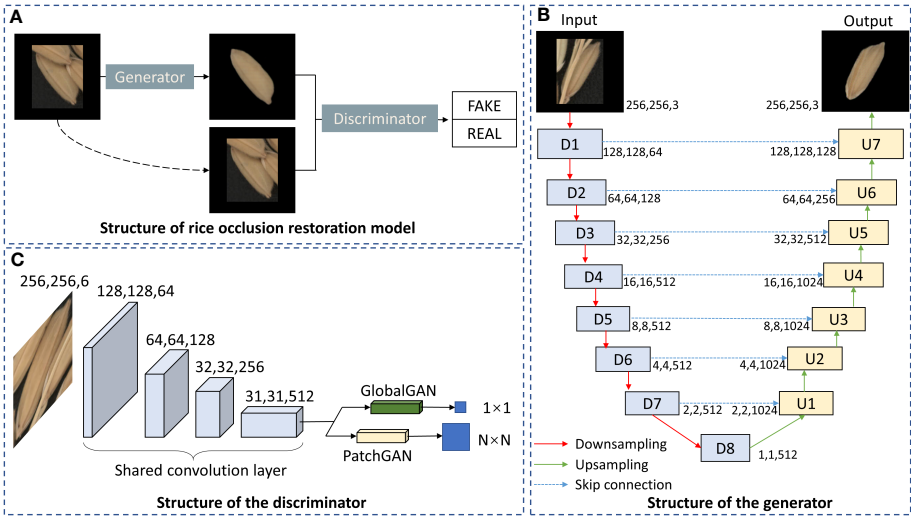


FIGURE 4
Structure of rice occlusion restoration model based on improved Pix2Pix. **(A)** Structure of rice occlusion restoration model. **(B)** Structure of the generator. **(C)** Structure of the discriminator.

from the RGB image of the grain. Compared with other channels, the contrast of the red channel was more obvious, which could better separate the grain from the background. Thirdly, the OTSU algorithm was used to automatically generate the optimal segmentation threshold and binarize the gray-scale image (Figure 5C). Fourthly, extracting the outer contour (Figure 5D) based on binary images was the basis for further trait calculation. Fifthly, the projection area, perimeter, and length of the rice grain are obtained by measuring the area of the grain, the outer contour length, and the distance between the farthest two points on the contour (Figure 5E). Obtain the intersection point of lines perpendicular to the major axis and the contour and use the maximum value of the distance as the grain width. Grain length/width ratio, perimeter/area ratio, equivalent ellipse and circularity could be further calculated by the previous traits. Table 2 shows the total fifteen on-panicle rice grain-related traits that can be extracted.

3 Result

3.1 Accuracy evaluation of on-panicle rice grain detection model

This experiment was run on a Dell Precision3650 server with Intel core i7-11700k CPU (32 GB memory) and NVIDIA GeForce RTX 3090 GPU (24 GB graphic memory). The software environment for deep-learning model training uses Python language under an Ubuntu operation system with Pytorch deep-learning framework.

Since the selection of model parameters will directly affect the final performance of the model, all the relevant parameters were adjusted before the training of the Faster R-CNN model for the situation where the number of on-panicle rice grains panicle is large, the size is small, and there is a certain degree of mutual

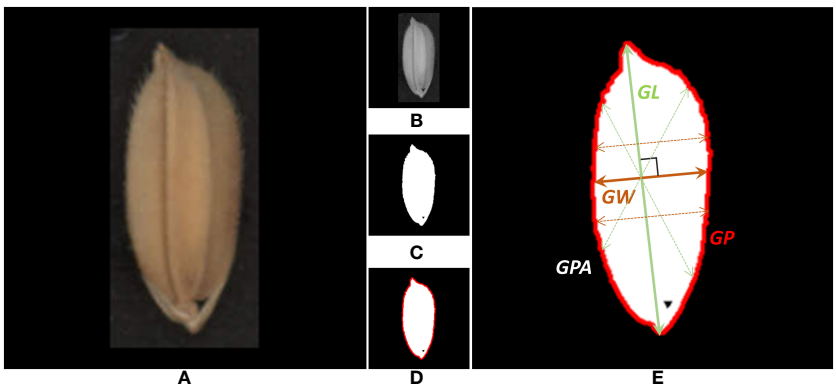


FIGURE 5
Automatic extraction pipeline for rice grain trait. **(A)** Single-grain scanning image. **(B)** Grayscale of the red channel. **(C)** Binary image after OTSU segmentation. **(D)** Outside contour of grain. **(E)** Morphological traits of grain.

TABLE 2 On-panicle rice grain phenotyping traits evaluated in this study.

Type	Traits	Abbreviation	Mean ± SD	Unit
Quantity trait	Grain number	GN	109.450 ± 48.782	–
Size trait	Mean value of grain length	MGL	8.607 ± 0.697	mm
	Standard deviation of grain length	SGL	0.589 ± 0.156	mm
	Mean value of grain width	MGW	2.770 ± 0.139	mm
	Standard deviation of grain width	SGW	0.248 ± 0.052	mm
	Mean value of grain projection area	MGPA	18.140 ± 1.307	mm^2
	Standard deviation of grain projection area	SGPA	1.875 ± 0.332	mm^2
	Mean value of grain perimeter	MGP	21.313 ± 1.481	mm
	Standard deviation of grain perimeter	SGP	1.386 ± 0.412	mm
Morphology trait	Mean value of grain area/perimeter ratio	MGAPR	0.851 ± 0.037	–
	Standard deviation of grain area/perimeter ratio	SGAPR	0.070 ± 0.011	–
	Mean value of grain circularity	MGC	0.507 ± 0.048	–
	Standard deviation of grain circularity	SGC	0.048 ± 0.007	–
	Mean value of grain length/width ratio	MGLWR	3.148 ± 0.340	–
	Standard deviation of grain length/width ratio	SGLWR	0.377 ± 0.100	–

Symbol "-" indicates that the parameter has no units.

occlusion to be detected in this study. The main parameters are shown in Table 3.

This study obtained 160 images of rice panicles in natural form. After manual labeling, all the images were divided into training, verification, and test sets in a 2:1:1 ratio. Based on the number of grains detected by the grain detection model and the actual number of grains in the panicle, the R^2 coefficient, mean absolute percentage error (MAPE) and root mean square error (RMSE) were used to measure the accuracy of the grain detection model. Mean average precision (mAP) was used to evaluate the accuracy of on-panicle rice grain location.

The R^2, MAPE and RMSE were calculated by the following equation:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (1)$$

TABLE 3 Main hyperparameter settings of rice grain detection model and occlusion restoration model.

Model	Hyperparameter	Setting
Rice grain detection model based on Faster R-CNN	Img_size	1425 x 1700 pixels
	NMS threshold	0.74
	Batch size	1
	Optimizer	SGD
	Learning rate	0.02
	Momentum	0.9
	Weight decay	0.0001
	RPN proposal number	4000
	Maximum epoch number	100
Rice grain occlusion restoration model based on improved Pix2Pix	Img_size	256x256 pixels
	Batch size	64
	Optimizer	Adam
	Learning rate	0.0002
	Maximum epoch number	500

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{y_i} \times 100\,\% \quad (2)$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \quad (3)$$

$\hat{y}_i$ is the trait parameters extracted from the grain image after restoration,

y_i is the true trait parameter,

$\bar{y}$ represents the average value of grain trait in samples

The two most commonly used feature extraction networks (He et al., 2016), based on Resnet50 and Resnet101, were used for performance comparison to select the network depth appropriate for rice grain detection (Figure 6A). At the beginning of training, the losses of both models decreased rapidly. After training for 20 epochs, the speed of loss reduction slowed down and converged after 80 epochs, and the model achieved the optimal state. Faster R-CNN using Resnet50 as a feature extraction network showed faster and better convergence. Figure 6A shows the AP variation curve of the Faster R-CNN model on the validation set. The changing trend of AP was opposite to the changing trend of loss. Finally, the Faster R-CNN model using Resnet50 as a feature extraction network achieved a higher AP, proving that Resnet50 was more suitable for detecting grains in the panicle than Resnet101. In addition, it was found that the AP reached 0.965, 0.933 and 0.601 when IoU was 0.50, 0.75 and 0.95, respectively.

The Faster R-CNN model marks the detected grains on the original map as detection frames, so the number of detection frames on the resulting map is the number of detected grains. Two types of image data, including natural morphology and separating the branch, were used to verify the accuracy of the network. The counting results are shown in Figure 6B. In all cases, the Faster R-CNN model can accurately count the on-panicle rice grains, with R^2 reaching 0.99. In addition, the counting accuracy of the Faster R-CNN model using Resnet50 as a feature extraction network was slightly higher than that of Resnet101, both in natural morphology and after artificially separating branches of the panicle. Compared with the case of separate branches, the model showed a slight decrease in accuracy in the case of the natural panicle. Specifically, the MAPE value increased by 0.69% and the RMSE value increased by 1.26. In conclusion, for natural rice panicles, the on-panicle rice grain detection model proposed in this study achieved 1.65% on MAPE and 4.39 on RMSE.

3.2 Performance evaluation of on-panicle rice grain occlusion restoration model

The dataset of the on-panicle rice grain occlusion restoration model must be paired. Various situations that may occur under natural conditions can be simulated by manually adjusting the grain occlusion ratio. Finally, 2000 pairs of images, set at an 8:2 ratio, were used in the model's training set and verification. The hyperparameter settings for the model are shown in Table 3.

Fifty pairs of images with varying degrees of occlusion were used to test the performance of the restoration model. The MAPE value was used to verify the model's restoration performance from the perspective of trait calculation (Figure 7). Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) were used to evaluate the model's ability in image feature restoration (Figure 8). To prove the excellent performance of the proposed model (Improved Pix2Pix, ImpP2P), this paper used the same hyperparameters to

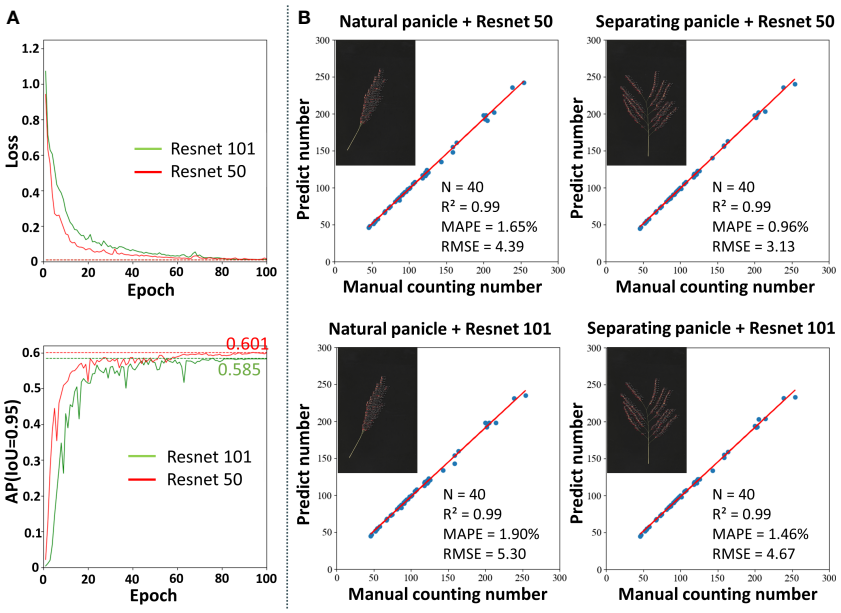


FIGURE 6
Performance of the rice grain detection model. **(A)** The training loss and AP curves with Resnet50 and Resnet101. **(B)** Comparison between the number of grains prediction by the model and manual counting.

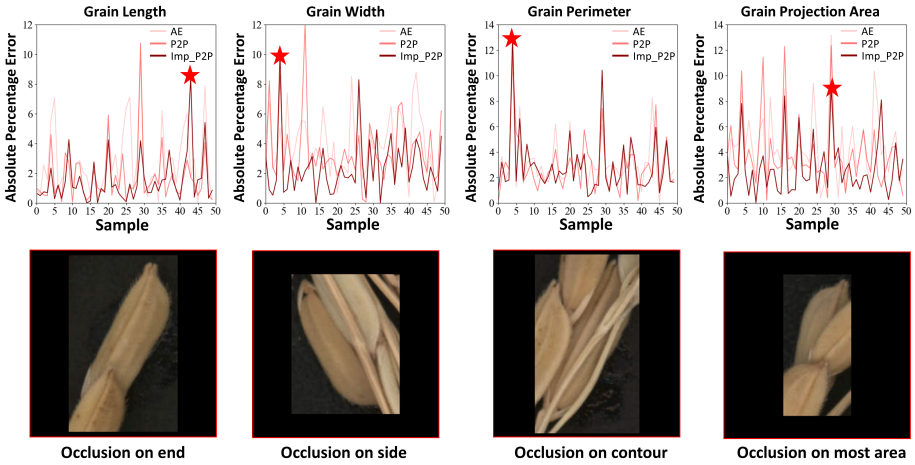


FIGURE 7
The restoration performance of each model in grain traits.

train an auto-encoder-based generating model (AE) and the original Pix2Pix model (P2P).

PSNR and SSIM have been commonly used evaluation metrics in image restoration. PSNR is based on the error of corresponding pixels with the dB unit. The higher the PSNR, the smaller the image distortion. SSIM is a full-reference image quality evaluation metric that measures the similarity of images from three aspects: luminance, contrast, and structure. Its value range is from 0 to 1. The larger the SSIM, the smaller the image distortion. Their calculation formulas are shown in equations 4 and 5, respectively.

$$PSNR = 10\star \log_{10}\left(\frac{MAX_I^2}{\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}X(i,j)-Y(i,j)^2}\right) \quad (4)$$

Where $X(i,j)$ and $Y(i,j)$ represent the pixel values of the real grain image and the grain image after restoration at the coordinate (i,j). m and n represent the height and width of the image. In this paper, both m and n are 256. MAX_I represents the maximum possible pixel value for the image, which is 255 in the case of an 8-bit binary grayscale image.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (5)$$

Where x and y represent the unoccluded grain image and the grain image after restoration, respectively. μ_x and μ_y represent the mean value of image respectively, and σ_x and σ_y represent the variance of image, respectively. σ_{xy} represents the covariance of the images. c_1 and c_2 are constants to avoid division by zero.

As shown in Figure 7, the improved Pix2Pix model performs best on all four grain traits. Regarding grain length, the MAPE of AE and P2P is 2.76% and 1.68%. As for the improved Pix2Pix model, it achieves 1.50%. They are 4.01%,3.48% and 2.41% in grain width for

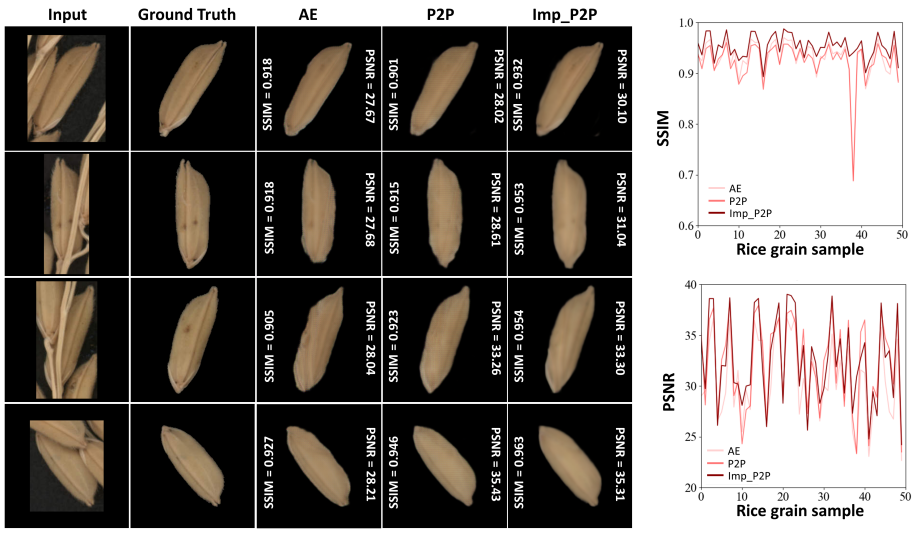


FIGURE 8
The restoration performance of each model in SSIM and PSNR.

AE, P2P and ImpP2P. In grain perimeter, they are 3.21%, 2.92% and 2.85% for AE, P2P and ImpP2P. In grain projection area, they are 4.42%, 3.84% and 2.74% for AE, P2P and ImpP2P. For individual rice grains, when occlusion occurs on the end, side, surround, and large area of the rice grain, it will cause significant errors in the length, width, perimeter, and area measurement, respectively.

Figure 8 shows four representative occlusion situations. All models can restore the approximate shape of the grain, and the ImpP2P model performs best in terms of overall structure and signal-to-noise ratio. For 50 rice grains with various degrees of occlusion, the average SSIM of ImpP2P achieves 0.953, better than AE (0.929) and P2P (0.924). The average PSNR of ImpP2P, AE and P2P are 32.49, 30.59 and 32.22. The visualization of four typical convolutional layers is shown in Figure 9, they are feature maps of the 1st encoding layer, the 2nd encoding layer, the 6th decoding layer and the 7th decoding layer. It can be found that the network has learned some features in different abstractive layers. In the encoding layers, the edge information of rice grains and significant internal areas are more concerned by the network. Correspondingly, in the decoding layers, the edge and internal center regions of the target are first restored and eventually extended to the entire rice grain.

3.3 Reliability verification of on-panicle rice grain phenotyping traits

Forty rice panicles selected randomly from eight varieties were used as samples to verify the reliability and robustness of the proposed on-panicle rice grain phenotyping method. The grain number of all rice panicles is distributed between 45 and 250. The result of the fifteen traits is shown in Table 2, and the average time-consuming for the processing and trait calculation of each rice panicle image is about 10 seconds. The time-consuming will inevitably increase with the growth of on-panicle grain numbers. Four morphological traits directly related to rice quality, mean grain length, mean grain width, mean grain length-width ratio and mean grain perimeter were used to compare the result by the method with the ground truth, which was obtained by manually measuring rice grains after threshing (Figure 2). For all grains on each panicle, mean length, mean width, mean length/width ratio and mean perimeter extracted by the method proposed in this paper are compared with the results of manual extraction (Figure 10), and the R^2 values between them reach 0.96, 0.83, 0.90 and 0.84. The MAPE values of the method versus the manual measurement for the four traits are 7.15%, 5.76%, 9.13% and 6.51%. Their RMSE values are 0.68mm, 0.18mm, 0.37 and 1.64mm, respectively. According to the conclusion in Figure 6, the R^2, MAPE, and RMSE values for grain number counting are 0.99, 1.65% and 4.39.

Referring to the concept of thousand-grain weight commonly used in rice seed evaluation, the thousand-grain length, thousand-grain width, and thousand-grain length/width ratio of eight varieties were obtained. The histograms of the distribution quantities on three traits of the samples are shown in Figure 11. Overall, most rice grains are 5-11mm long for all varieties, the grain width of samples from subspecies japonica is larger, appearing thicker and shorter compared to subspecies indica, while samples from subspecies indica are slender. As shown in Figure 11, rice grains of Kenyan1803 are generally short, most of them in the range of 6-8mm. Grains of Ganzi have a prominent length; most are larger than 8mm, and a considerable part is higher than 10mm. As for grain width, most of the grain widths of all varieties are between 2 and 4mm, and some varieties, such as Wangdao2 and Ganzhi, have very few grains with a grain width of less than 2mm. For Kenyan1803, almost all the grains are wider than 2.5mm. The length/width ratio is one of rice grains' most crucial reference traits, and the eight varieties are mostly distributed between 2 and 5. The grain distribution range of Kenyan1803 is the most concentrated, and most are between 2 and 3, indicating that the grains of this variety have little difference in shape. On the contrary, the grain length-width ratio distribution of Jiujiuxinxiang, Wendao21, Z98-

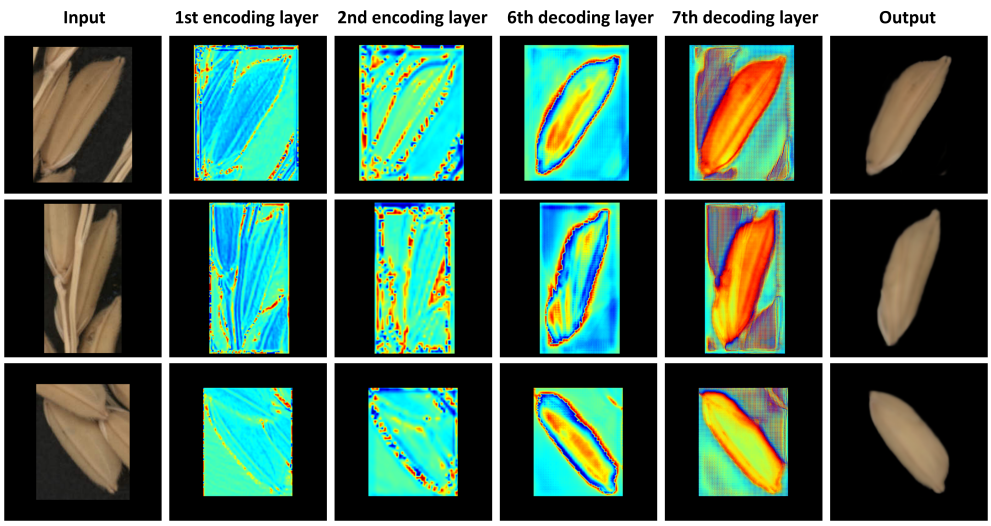


FIGURE 9
The visualization of typical convolutional layers in occlusion restoration model.

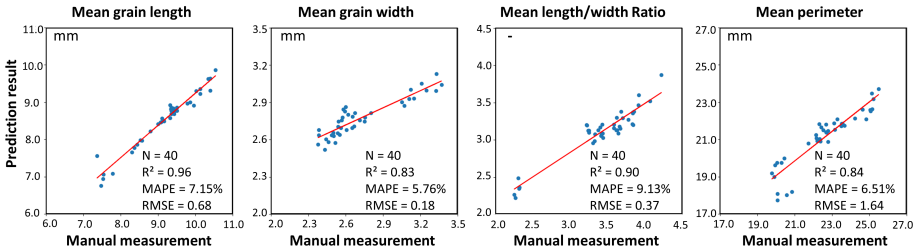


FIGURE 10
Analysis of extraction results of four morphological traits.

308 and Ganzhi is relatively dispersed, indicating that the grains of these varieties have significant differences in shape.

4 Discussion

The number of grains per panicle and grain morphological-related traits of rice varieties are essential reference data for rice breeding and functional identification of crucial genes. High throughput, convenient, and economical phenotypic trait evaluation methods are crucial. In previous work, the accuracy and efficiency of measuring grain-related traits were often contradictory. High-accuracy methods often rely on complex mechanical equipment and post-processing algorithms. In contrast, simple and efficient imaging and processing methods are challenging to obtain accurate and comprehensive phenotyping traits. These problems limit the promotion and development of these phenotyping methods. This study proposed a method based on visible light scanning imaging and deep learning technology for on-panicle rice grain traits, which balanced measurement efficiency and accuracy. Visible light scanning equipment is inexpensive (The price is less than one percent or even lower than that of large seed testing equipment and X-ray imaging equipment), readily available, and can provide stable imaging results in laboratory and field environments. The method in this study does not require complex sorting and threshing of samples, which often takes one minute or even longer for a single rice panicle. The method in this study could potentially expand to the detection of rice panicle and grain traits at multiple growth stages of rice.

Due to the natural occlusion of on-panicle rice grains, previous measurement methods can only estimate the number of rice grains and other morphological traits after separating the branches. Separating and fixing branches is time-consuming and fragile, and the accuracy of trait extraction is easily affected by the degree of separation. Even so, avoiding the possible occlusion between adjacent grains is impossible. Deep learning technology provides a way to solve this problem. This study proposed a cascade model based on the Faster R-CNN model and improved the Pix2Pix model to achieve accurate counting and occlusion restoration of the on-panicle grains. From the restoration results, morphological traits of grains can be extracted without panicle separation. For rice panicles with grain numbers between 45 and 250, the grain detection will be completed in about 1 second, and the extraction of traits will take about 10 seconds. The sufficient experimental result proves the high accuracy and reliability of the method.

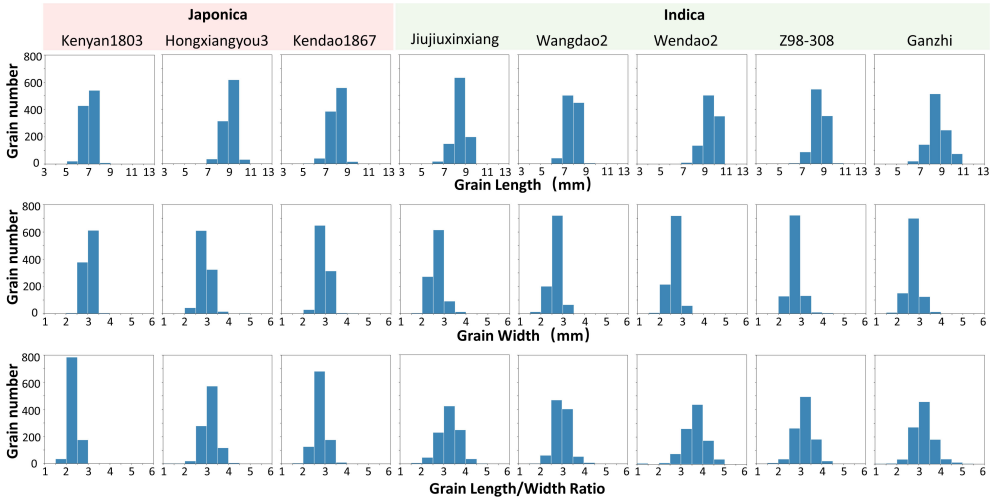


FIGURE 11
Analysis of on-panicle grain traits of eight rice varieties.

The method proposed in this study achieves high-throughput and high-accuracy extraction of on-panicle rice grain traits without separating the branch. However, there are still some directions for improvement. Firstly, portable visible-light scanning devices could be developed for researchers to use in field environments. Secondly, simultaneous imaging of multiple rice panicles is possible, which can double the efficiency of trait analysis. For the deep learning model, more occlusion scenes and occlusion degrees can be designed to improve the model's accuracy when applied to multiple varieties.

5 Conclusions

This study proposed a high-throughput and separating-free method for extracting on-panicle rice grains phenotyping traits based on visible light scanning imaging and deep learning. Samples from eight varieties were used to verify the accuracy of the method. The results showed that the method proposed in this paper could obtain images of rice panicles within 60 seconds and automatically extract 15 traits of on-panicle grains in about 10 seconds. Compared with manual measurement, the R^2 values of the method on grain counting, grain length, grain width, grain length/width ratio and grain perimeter reach 0.99, 0.96, 0.83, 0.90 and 0.84, respectively. The difference in the distribution of grain traits among different varieties indicates that this method can effectively distinguish varieties and help screen high-quality traits. In general, the method proposed in this paper can be used to realize the rapid measurement of rice grain traits and has the potential to be extended to the field environment and other crops.

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/BME-PhenoTeam/Method-for-on-panicle-rice-grain-detection.

Author contributions

YL and LY designed the research. YL and JW performed the experiments, analyzed the data, and wrote the manuscript. LF, LY, and QL helped to perform the experiments. QL and LY supervised the project. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by Hainan Yazhou Bay Seed Lab (B21HJ0904), Sanya Yazhou Bay Science and Technology City (SCKJ-JYRC-2023-25), Hainan Provincial Natural Science Foundation of China (322MS029).

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/fpls.2023.1219584/full#supplementary-material

References

Al-Tam, F., Adam, H., Anjos, A., D., Lorieux, M., Larmande, P., Ghesquière, A., et al. (2013). P-TRAP: a panicle trait phenotyping tool. *BMC Plant Biol.* 13 (1), 1–14. doi: 10.1186/1471-2229-13-122

Bradski, G. (2000). The openCV library. *Dr. Dobb's Journal: Software Tools Prof. Programmer* 25 (11), 120–123.

Crossa, J., Pérez-Rodríguez, P., Cuevas, J., Montesinos-López, O., Jarquín, D., De Los Campos, G., et al. (2017). Genomic selection in plant breeding: methods, models, and perspectives. *Trends Plant Sci.* 22 (11), 961–975. doi: 10.1016/j.tplants.2017.08.011

Deng, R., Qi, L., Pan, W., Wang, Z., Fu, D., and Yang, X. (2022). Automatic estimation of rice grain number based on a convolutional neural network. *JOSA A* 39 (6), 1034–1044. doi: 10.1364/JOSAA.459580

Duan, L., Yang, W., Bi, K., Chen, S., Luo, Q., and Liu, Q. (2011a). Fast discrimination and counting of filled/unfilled rice spikelets based on bi-modal imaging. *Comput. Electron. Agric.* 75 (1), 196–203. doi: 10.1016/j.compag.2010.11.004

Duan, L., Yang, W., Huang, C., and Liu, Q. (2011b). A novel machine-vision-based facility for the automatic evaluation of yield-related traits in rice. *Plant Methods* 7 (1), 1–13. doi: 10.1186/1746-4811-7-44

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

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition. 580–587. doi: 10.1109/CVPR.2014.81

Gong, L., Lin, K., Wang, T., Liu, C., Yuan, Z., Zhang, D., et al. (2018). Image-based on-panicle rice [Oryza sativa L.] grain counting with a prior edge wavelet correction model. *Agronomy* 8 (6), 91. doi: 10.3390/agronomy8060091

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Las Vegas, NV, USA). 770–778. doi: 10.1109/CVPR.2016.90

Hu, W., Zhang, C., Jiang, Y., Huang, C., Liu, Q., Xiong, L., et al. (2020). Nondestructive 3D image analysis pipeline to extract rice grain traits using X-ray computed tomography. *Plant Phenomics* 2020. doi: 10.34133/2020/3414926

Huang, C., Duan, L., Liu, Q., and Yang, W. (2013). Development of a whole-feeding and automatic rice thresher for single plant. *Math. Comput. Model.* 58 (3-4), 684–690. doi: 10.1016/j.mcm.2011.10.033

Huang, C., Li, W., Zhang, Z., Hua, X., Yang, J., Ye, J., et al. (2022). An intelligent rice yield trait evaluation system based on threshed panicle compensation. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.900408

Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. (2017). "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Honolulu, HI, USA). 1125–1134. doi: 10.1109/CVPR.2017.632

Li, R., Li, M., Ashraf, U., Liu, S., and Zhang, J. (2019). Exploring the relationships between yield and yield-related traits for rice varieties released in China from 1978 to 2017. *Front. Plant Sci.* 10, 543. doi: 10.3389/fpls.2019.00543

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Honolulu, HI, USA). 2117–2125. doi: 10.1109/CVPR.2017.106

Mirza, M., and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv*. preprint arXiv:1411.1784. doi: 10.48550/arXiv.1411.1784

Ren, S., He, K., Girshick, R., 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*. 39 (6), 1137–1149. doi: 10.1109/TPAMI.2016.2577031

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. Proceedings, Part III 18, 234-241* (Munich, Germany: Springer International Publishing). doi: 10.48550/arXiv.1505.04597

Su, L., and Chen, P. (2019). A method for characterizing the panicle traits in rice based on 3D micro-focus X-ray computed tomography. *Comput. Electron. Agric.* 166, 104984. doi: 10.1016/j.compag.2019.104984

Sun, D., Robbins, K., Morales, N., Shu, Q., and Cen, H. (2022). Advances in optical phenotyping of cereal crops. *Trends Plant Sci.* 27 (2), 191–208. doi: 10.1016/j.tplants.2021.07.015

Tester, M., and Langridge, P. (2010). Breeding technologies to increase crop production in a changing world. *Science* 327 (5967), 818–822. doi: 10.1126/science.1183700

Wang, D., Jiawei, C., Shen, L., Wang, A. H., and Zhou, J. (2022). The development of a vision-based phenotypic analysis algorithm for measuring spikelet-related traits in rice. *Plant Physiol. J.* 58 (5), 957–971. doi: 10.13592/j.cnki.ppj.200009

Watt, M., Fiorani, F., Usadel, B., Rascher, U., Muller, O., and Schurr, U. (2020). Phenotyping: new windows into the plant for breeders. *Annu. Rev. Plant Biol.* 71, 689–712. doi: 10.1146/annurev-arplant-042916-041124

Whan, A. P., Smith, A. B., Cavanagh, C. R., Ral, J. P. F., Shaw, L. M., Howitt, C. A., et al. (2014). GrainScan: a low cost, fast method for grain size and colour measurements. *Plant Methods* 10, 1–10. doi: 10.1186/1746-4811-10-23

Wu, W., Liu, T., Zhou, P., Yang, T., Li, C., Zhong, X., et al. (2019). Image analysis-based recognition and quantification of grain number per panicle in rice. *Plant Methods* 15, 1–14. doi: 10.1186/s13007-019-0510-0

Yang, W., Feng, H., Zhang, X., Zhang, J., Doonan, J. H., Batchelor, W. D., et al. (2020). Crop phenomics and high-throughput phenotyping: past decades, current challenges, and future perspectives. *Mol. Plant* 13 (2), 187–214. doi: 10.1016/j.molp.2020.01.008

Yang, W., Guo, Z., Huang, C., Duan, L., Chen, G., Jiang, N., et al. (2014). Combining high-throughput phenotyping and genome-wide association studies to reveal natural genetic variation in rice. *Nat. Commun.* 5 (1), 5087. doi: 10.1038/ncomms6087

Yu, L., Shi, J., Huang, C., Duan, L., Wu, D., Fu, D., et al. (2021). An integrated rice panicle phenotyping method based on X-ray and RGB scanning and deep learning. *Crop J.* 9 (1), 42–56. doi: 10.1016/j.cj.2020.06.009

Zhang, Q. (2007). Strategies for developing green super rice. *Proc. Natl. Acad. Sci.* 104 (42), 16402–16409. doi: 10.1073/pnas.0708013104

Zhao, S., Zheng, H., Chi, M., Chai, X., and Liu, Y. (2019). Rapid yield prediction in paddy fields based on 2D image modelling of rice panicles. *Comput. Electron. Agric.* 162, 759–766. doi: 10.1016/j.compag.2019.05.020

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 27 November 2023
DOI 10.3389/fpls.2023.1268098

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Jana Shafi,
Prince Sattam Bin Abdulaziz University, Saudi Arabia
Danfeng Hong,
Chinese Academy of Sciences (CAS), China
Teng Miao,
Shenyang Agricultural University, China

*CORRESPONDENCE
Peng Wu
wp@zafu.edu.cn

†These authors share first authorship

RECEIVED 27 July 2023
ACCEPTED 09 November 2023
PUBLISHED 27 November 2023

CITATION
Yi X, Wang J, Wu P, Wang G, Mo L, Lou X, Liang H, Huang H, Lin E, Maponde BT and Lv C (2023) AC-UNet: an improved UNet-based method for stem and leaf segmentation in Betula luminifera. *Front. Plant Sci.* 14:1268098. doi: 10.3389/fpls.2023.1268098

COPYRIGHT
© 2023 Yi, Wang, Wu, Wang, Mo, Lou, Liang, Huang, Lin, Maponde and Lv. 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.

AC-UNet: an improved UNet-based method for stem and leaf segmentation in Betula luminifera

Xiaomei Yi[1†], Jiaoping Wang[1†], Peng Wu[1*], Guoying Wang[1], Lufeng Mo[1], Xiongwei Lou[1], Hao Liang[1], Huahong Huang[1], Erpei Lin[1], Brian Tapiwanashe Maponde[1] and Chaihui Lv[2]

[1]College of Mathematics and Computer Science, Zhejiang A&F University, Hangzhou, China, [2]Hangzhou Ganzhi Technology Co Ltd., Hangzhou, China

Plant phenotypic traits play an important role in understanding plant growth dynamics and complex genetic traits. In phenotyping, the segmentation of plant organs, such as leaves and stems, helps in automatically monitoring growth and improving screening efficiency for large-scale genetic breeding. In this paper, we propose an AC-UNet stem and leaf segmentation algorithm based on an improved UNet. This algorithm aims to address the issues of feature edge information loss and sample breakage in the segmentation of plant organs, specifically in Betula luminifera. The method replaces the backbone feature extraction network of UNet with VGG16 to reduce the redundancy of network information. It adds a multi-scale mechanism in the splicing part, an optimized hollow space pyramid pooling module, and a cross-attention mechanism in the expanding network part at the output end to obtain deeper feature information. Additionally, Dice_Boundary is introduced as a loss function in the back-end of the algorithm to circumvent the sample distribution imbalance problem. The PSPNet model achieves mIoU of 58.76%, mPA of 73.24%, and Precision of 66.90%, the DeepLabV3 model achieves mIoU of 82.13%, mPA of 91.47%, and Precision of 87.73%, on the data set. The traditional UNet model achieves mIoU of 84.45%, mPA of 91.11%, and Precision of 90.63%, and the Swin-UNet model achieves . The mIoU is 79.02%, mPA is 85.99%, and Precision is 88.73%. The AC-UNet proposed in this article achieved excellent performance on the Swin-UNet dataset, with mIoU, mPA, and Precision of 87.50%, 92.71%, and 93.69% respectively, which are better than the selected PSPNet, DeepLabV3, traditional UNet, and Swin-UNet. Commonly used semantic segmentation algorithms. Experiments show that the algorithm in this paper can not only achieve efficient segmentation of the stem and leaves of Betula luminifera but also outperforms the existing state-of-the-art algorithms in terms of both speed. This can provide more accurate auxiliary support for the subsequent acquisition of plant phenotypic traits.

KEYWORDS

Betula luminifera, stem and leaf division, UNET, hollow space pyramidal pooling, crossed attention

1 Introduction

Plant phenotyping is an emerging science that links genetics with plant physiology, ecology, and agriculture (Li et al., 2020). Plant phenotypes, at the latent level, allow for the extraction of important traits such as plant size, shape, and growth dynamics (Zhou et al., 2018). At a deeper level, they can reflect physical, physiological, and biochemical traits that characterize the structure and function of plant cells, tissues, organs, plants, and populations. Plant stem and leaf segmentation are important for obtaining plant phenotypic traits at different growth cycles. The key to plant phenotypic analysis is the effective and correct segmentation of plant organs. Since 1990, research related to plant organ segmentation, especially for diseased leaf identification, has been emerging. The phenotyping of 2D images is usually based on traditional image processing, machine learning, and pattern recognition algorithms, such as threshold-based segmentation (Fu et al., 2019), edge detection (Wang et al., 2018b), region growing (Scharr et al., 2016), clustering (Abinaya and Roomi, 2016), and their combined extensions to each other (Kalyoncu and Toygar, 2015; Pape and Klukas, 2015a; Pape and Klukas, 2015b). Although the above methods can also achieve image classification and segmentation, the results are slightly less satisfactory. In recent years, with the application of deep learning in phenotypic data parsing, a new perspective has been taken to solve some of the data parsing bottlenecks encountered in phenomics research. In particular, deep learning has made a major breakthrough in the field of semantic segmentation. Deep learning, based on convolutional neural networks (CNN), has reached an advanced level in image classification and segmentation. Sadeghi-Tehran P et al. (Sadeghi-Tehran et al., 2019) segmented images into hyperparameters by using simple linear iterative clustering to obtain canopy-related features. These features were then fed into a CNN classification model to achieve semantic segmentation of wheat ears. Tamvakis P N et al. (Tamvakis et al., 2022) used deep learning methods (supervised and unsupervised learning-based approaches) to semantically segment images of grape leaves. They developed an automatic leaf phenotype analysis object detection system by segmentation that generates information about the structure and function of the leaves. Frank Gyan Okyere et al. (Okyere et al., 2023) developed a neural network-based segmentation tool to achieve high-throughput phenotypic analysis of cylinder beans and wheat. Conventional segmentation methods are more commonly implemented for the segmentation of crop stems, leaves, and fruits, and less frequently for forestry applications.

Betula luminifera is an economically valuable forest tree widely distributed in China. It is commonly found in the southern region of the Qinling Mountains and Huaihe River, at elevations ranging from 600 to 1700 meters above sea level. The wood of Betula luminifera is excellent, with a yellowish or reddish-brown color, fine texture, and hardness, making it highly valuable with diverse applications.To address the efficient acquisition of plant phenotypic traits, this study utilized self-collected images of Betula luminifera. It employed a modified version of the traditional UNet (Ronneberger et al., 2015), replacing the coding part with the VGG16 backbone feature extraction network (Simonyan and Zisserman, 2014; Deng et al., 2009). Additionally, the study introduced the ASPP module (Chen et al., 2018); (Chen et al., 2018); (Chen et al., 2017) with further improvements (He et al., 2016) (Wang et al., 2018a) and incorporated the cross-attention mechanism (CCA) (Huang et al., 2018). For the loss function, the compound loss Dice_Boundary (Kervadec et al., 2019); (Ma et al., 2021) was employed. The resulting model, called AC-UNet, aimed to re-fine the segmentation of Betula luminifera's stems and leaves. The study conducted experiments using Betula luminifera seedlings from the Pingshan Experimental Base of Zhejiang Agriculture and Forestry University in northwestern Zhejiang Province as test subjects. The experimental results were compared with those of PSPNet (Zhao et al., 2017), DeepLabV3 (Chen et al., 2018), UNet, and Swin-UNet (Cao et al., 2021),The findings demonstrated that the proposed model outperformed other segmentation algorithms in terms of performance. Specifically, it exhibited more detailed feature extraction along the edges of the plant stems and leaves, leading to an overall better plant restoration.

The contribution of this paper consists of the following two main parts:

1. Constructing a Betula luminifera dataset with Betula luminifera seedlings as the experimental object, based on three lineages, namely Taihuyuan in Hangzhou, Napo in Guangxi, and Anhua in Hunan.

2. A semantic segmentation method is proposed based on an improved UNet piggy-backing on ASPP and CCA. The model uses VGG16 as the backbone network to extract deep semantic features, piggybacks on ASPP modules with appropriate hole expansion rates set, fuses the CCA mechanism with Dice_Boundary loss, and captures long-range global feature information by reducing the number of network parameters and network depth to obtain multiscale semantic information and improve segmentation accuracy.

The construction of the data set meets the data needs of the scientific research community and ecologists for Betula glabra research, and provides a powerful tool for ecosystem monitoring and plant genetics research. The improved UNet semantic segmentation technology comes from the challenges encountered in light bark research. With a deep understanding of the complexity of seedlings and the limitations of traditional segmentation methods, detailed experiments have demonstrated the significant performance of this algorithm in processing Betula glabra data sets. We believe that this method is not only applicable to Betula glabra but also has the potential to Its wide range of applications include medical image segmentation, cartography, and botanical research.

2 Materials and methods

2.1 Experimental dataset

A forestry dataset was constructed based on image segmentation of Betula luminifera seedlings to obtain a large amount of accurate data about Betula luminifera. These images are used to segment plant organs such as leaves and stems, providing an effective aid for plant monitoring and plant phenotype analysis.

Seedling cultivation and data collection were carried out at the Pingshan Experimental Base of Zhejiang Agriculture and Forestry University. The study area is located in Lin'an District, Hangzhou City, Zhejiang Province, at Zhejiang Agriculture and Forestry University, Jincheng Street, with geographical coordinates ranging from 118°51' to 119°52' E and 29°56' to 30°23' N. The area has a subtropical monsoon climate with four distinct seasons, abundant light, and rainfall, making it suitable for the cultivation of Betula luminifera seedlings. The three seedlings selected for this study were sourced from provinces south of their natural distribution in the Qinling and Huaihe River basins (Zhejiang, Guangxi, and Hunan) and covered a wide range of seedling morphology to verify the applicability of the method to different types of plants. The seedlings for data collection were selected from the uniformly cultivated Betula luminifera at the Pingshan Experimental Base of Zhejiang Agriculture and Forestry University. There were a total of 300 plants. The cultivation took about 30 days. The height of the plants ranged from 10 to 35 cm, and the seedlings grew upright. To facilitate data collection, each seedling was individually transplanted into a uniform calibre plastic pot for numbering. The cultivation greenhouse was maintained at a temperature of 23°C during the day with natural light, 20°C at night, and a uniform humidity setting of 70%.

In this study, black velvet cloth was used as the shooting background during collection to reduce the impact of background objects and light source scattering and provide a stable environment for plant shooting. The iPhone 13 Pro Max mobile phone is used as a plant shooting device to obtain high-quality plant images. The same shooting device is always used during collection, which stabilizes the image quality and helps capture the microscopic details of the seedlings. Image collection will be carried out at different times in November 2022 and December 2022 to ensure that the collected data sets are in a stable growth state at the same stage. During filming, the lens was held flush with the target of the photographed plant, and the distance from the plant sample to the lens was kept at d = 90 cm (Figure 1). The plants were placed on a tray at the bottom of the platform, and they were rotated in turn at 90-degree clockwise angles. Multiple angles were taken for the front, back, left, and right sides of each plant, with the light source placed at the viewpoint directly opposite the sample, pointing towards the photographed sample. A total of 1200 valid images were taken during this data collection process. After screening and culling (removing images with high longitudinal overlap and sparse foliage), 490 images were obtained, forming the original image dataset.

The plant stems and leaves in the original image dataset were accurately labeled using the Labelme image annotation software, as shown in Figure 2. The labeled images contain three semantic categories: the background part, stem, and leaf. The pixel values assigned to these categories are as follows: the background part is 0, the stem is 1, and the leaf is 2. The dataset consists of a total of 490 RGB labeled images, which were randomly divided into a training set and a validation set in a ratio of 9:1. This resulted in 442 training samples being inputted into the UNet model for training, while the remaining 48 im-ages were used for validation.

2.2 OTSU

The maximum variance between classes method, commonly known as the Otsu method, is a self-fitting method for automatically finding thresholds for the bimodal case. It was proposed by the Japanese scholar Nobuyuki Otsu in 1979 and is currently recognized as a relatively reasonable choice for threshold segmentation, yielding good segmentation results. As the name suggests, the method uses the idea of maximizing the variance between the target and background regions for segmenting images. In other words, the optimal threshold T is chosen to maximize the variance between the target and the background, defining the region smaller than the threshold T as D1 and the region larger than the threshold as D2. This allows the required region to be distinguished based on the threshold definition. The advantage of this method is its simplicity and speed of calculation. It is not easily affected by image brightness and contrast and is widely used in image binarization segmentation.

2.3 HSV colour threshold splitting

HSV is a color space based on the intuitive properties of color, created by A.R. Smith in 1978 and also known as the hexagonal

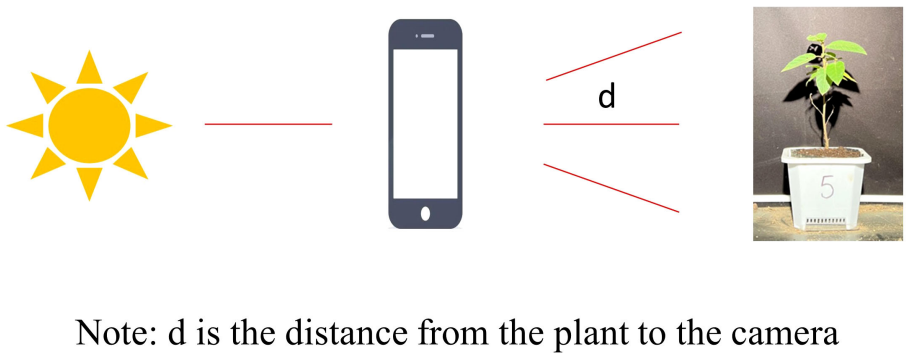


FIGURE 1
Diagram of the plant shot.

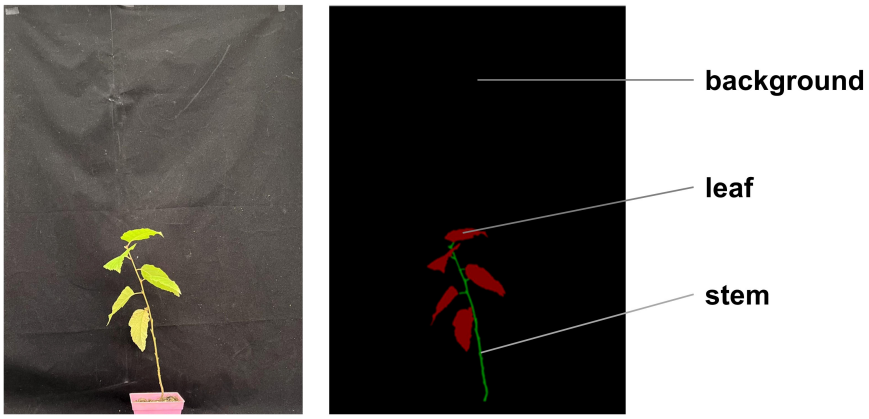


FIGURE 2
Schematic diagram of image annotation.

cone model. It is a color system that is more commonly used in people's lives compared to RGB. HSV is commonly found in TV remote controls, painting palettes, and brightness adjustments in video software. Subjectively, the HSV color system aligns more with how people describe color. The parameters of color in this system are: Hue, Saturation, and Value.

Hue (H) represents popular perceptions of colors like red, green, blue, etc. However, more refined expressions can be used such as plum red, magenta, grass green, dark green, and so on.

Saturation (S) refers to the intensity or shade of a color. It is a concept that takes values in the range of 0-100%. For example, in the case of red, bright red has high saturation as it represents a pure color. If mixed with other shades of color, the saturation decreases, such as in the case of pink.

Value (V) represents brightness or purity of color, ranging from 0 to 100%. This value is commonly used when adjusting the brightness of a screen.

Referring to the color range table in Table 1, the region of interest (ROI) in the image is selected. Generally, before carrying out this step, denoising is required. However, in this paper, since it is based on the mask map of the prediction results and there is no noise effect, denoising is performed directly on the extracted leaf part.

2.4 Improved UNet-based plant image segmentation algorithm

2.4.1 Network architecture

The UNet network (Ronneberger et al., 2015) was proposed in 2015, and at the time of its mention, its main application was semantic segmentation of medical images. The emergence of UNet has greatly reduced the amount of data required for training deep learning neural networks, which originally required thousands of annotated data to be trained, and it pioneered the application of neural networks to image segmentation. This network is still widely used despite the birth of many segmentation networks. The structure of the traditional UNet model is shown in Figure 3. The highlight of this structure is that the whole network presents a U-shaped structure, hence the name UNet. The UNet network is very simple, with the first half acting as feature extraction and the second half as upsampling. This structure is also called an encoder-decoder structure in some literature. The downsampling part refers to the basic structure of a convolutional neural network, with two convolutional units composed of 3x3 convolutions, each followed by a ReLU and a 2x2 maximum pooling, while doubling the number of feature channels to capture context for feature extraction and learning. In the upsampling section, two convolutional units

TABLE 1 HSV colour space colour range.

	black	gray	white	red		orange	yellow	green	cyan	blue	purple
Hmin	0	0	0	0	156	11	26	35	78	100	125
Hmax	180	180	180	10	180	25	34	77	99	124	155
Smin	0	0	0	43		43	43	43	43	43	43
Smax	255	43	30	255		255	255	255	255	255	255
Vmin	0	46	221	46		46	46	46	46	46	46
Vmax	46	220	255	255		255	255	255	255	255	255

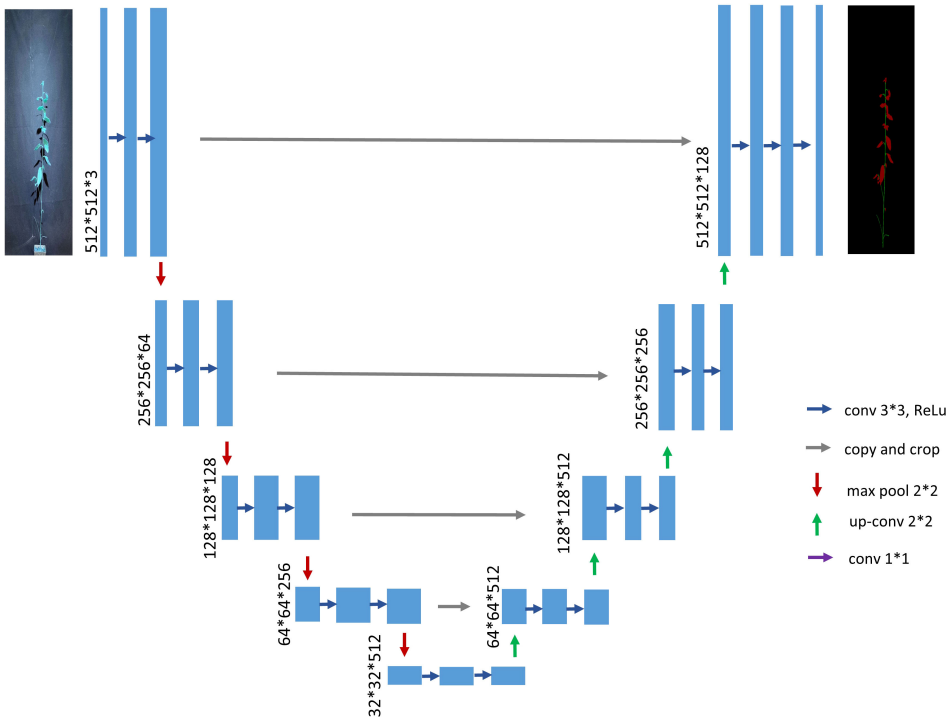


FIGURE 3
Traditional UNet network architecture diagram.

composed of 3x3 convolutions, immediately followed by 2x2 convolutions, halve the number of feature channels that were originally doubled and concatenate them with the corresponding cropped feature maps in the encoding section. The missing values at the boundaries after each previous convolution step are filled in. Finally, each 64-component feature vector is mapped to the desired number of classes using a 1x1 convolutional unit.

The UNet model achieves superior segmentation results on various datasets. The feature information of plants itself is relatively stable, and there are no special or novel feature information. Therefore, both high-level semantic features and low-level semantic features are extremely important. As one of the current excellent semantic segmentation networks, the UNet network also has some shortcomings. Firstly, as each pixel point needs to take a patch, it makes the patches of two neighboring pixel points too similar, resulting in a significant amount of redundancy. This redundancy not only leads to a poor segmentation situation but also reduces the training speed of the network. Secondly, it is challenging to achieve both localization accuracy and access to contextual information simultaneously. The larger the patch size, the more maximum pooling layers are required, which in turn reduces the localization accuracy. Additionally, as the number of pooling layers increases, more information is lost.

Then, directly inputting the shallow network information into the decoder part will cause a low rate of obtaining semantic information of stem-and-leaf edges, resulting in poor segmentation accuracy. To improve the training performance and address the aforementioned deficiencies, this paper introduces the following improvements to the traditional UNet model architecture:

(1) Using VGG16 as the main stem feature extraction part of UNet (Ronneberger et al., 2015), which significantly reduces the amount of parameter computation of the model, decreases memory occupation, and improves the computational speed.

(2) Introducing an improved ASPP module (Chen et al., 2018) in the middle of the encoder and de-coder, which expands the sensory field without losing semantic information and enhances the network's feature extraction capability.

(3) Introducing CCA (Huang et al., 2018) in the decoding part to reduce GPU memory usage and im-prove model segmentation accuracy.

(4) Replacing the loss function with Dice_Boundary composite loss (Kervadec et al., 2019); (Ma et al., 2021) to ad-dress the imbalance of pixel distribution between categories. The improved model structure is shown in Figure 4.

2.4.2 Optimised feature extraction module

The external environment presents various interferences in the Betula luminifera im-age acquisition process. Additionally, the UNet network itself employs a specific number of convolution kernels in the encoding process to extract image features. This multi-step convolutional operation leads to excessive redundancy in the feature map of the segmentation model, resulting in poor semantic interpretation of Betula luminifera images and reduced network training speed. To reduce parameter redundancy within the UNet network, enhance network depth for improved classification accuracy, and extract more abstract higher-order features from the image, VGG16 is utilized as the backbone feature network of UNet. The use of pre-trained mature models

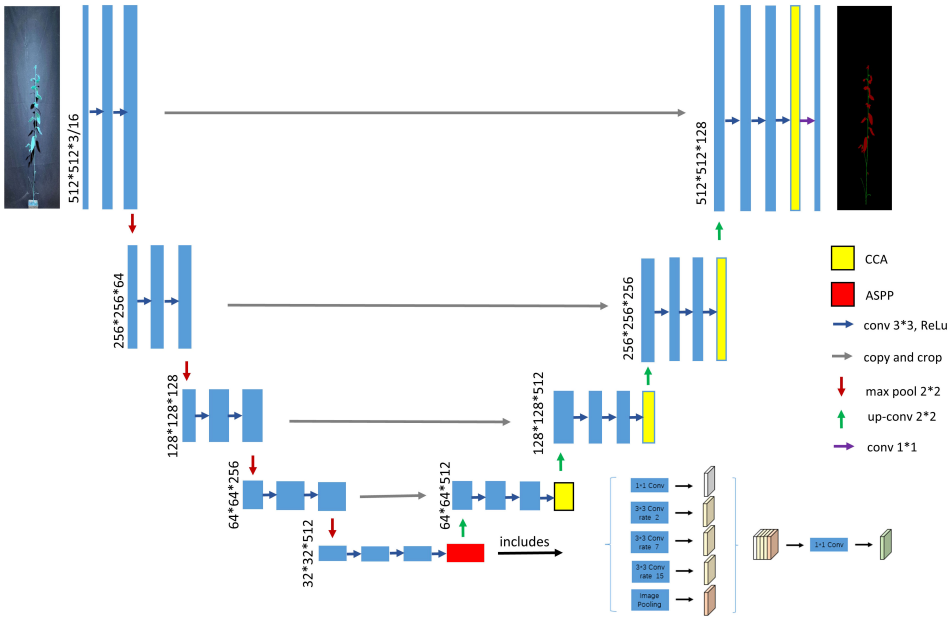


FIGURE 4
Diagram of the improved UNet network architecture.

significantly enhances the training speed of the UNet network while ensuring accuracy. Compared with other networks, VGG16 can employ multiple 3x3 convolutional kernels instead of large-scale convolutional kernels, thus reducing the number of parameters during network operation. In this algorithm, the three fully connected layers of VGG16 are omitted due to their excessive consumption of computational resources. Moreover, VGG16 and VGG19 have demonstrated better segmentation effects in practical applications. Compared with VGG19, VGG16 has three fewer convolutional layers, making it a shallower network. Given that the network's quality is ensured, this paper selects the network with fewer parameters. The structure of VGG16 as the main feature extraction network is depicted in Figure 5. Furthermore, this paper adopts pre-training weights from Imagenet (Deng et al., 2009) for transfer learning to improve the model's generalization.

2.4.3 Fused multidimensional feature acquisition

Betula luminifera images contain multi-scale objects, such as small leaves versus larger leaves, smooth branches versus branches with small forks, and so on. The image is subjected to continuous convolution and pooling or other downsampling operations in the network to integrate the multi-scale contextual information, which tends to result in low-resolution feature maps in the network, making it impossible to reconstruct the image details.In order to overcome the disadvantages of local information loss and lack of correlation of distant information due to the grid effect when using a single null convolution, the expansion rate is changed based on the original null convolution in the ASPP module (Chen et al., 2018). A type of null convolution that can increase the sensory field while still maintaining sensitivity to details is proposed. The optimized cavity convolution can effectively expand the receptive field of the convolution kernel to incorporate larger contextual information without increasing the number of parameters and computational effort.

In the semantic segmentation algorithm, the two-dimensional hole convolution is achieved by inserting 0 between each pixel of the convolution kernel. For a convolution kernel with a size of k×k, the size after the hole convolution is kd×kd, where kd = k + (k−1) × (r−1). Figure 6 shows the size of the convolution kernel receptive fields at different expansion rates, and the dilated convolution receptive fields with expansion rates of 1, 2, and 4 are 3×3, 5×5, and 9×9, respectively.

The ASPP module uses multiple parallel cavity convolution layers with different sampling rates. The features extracted for each sampling rate are further processed in separate branches and fused to generate the final result. This approach allows for the extraction of multiscale features of the object and image context while ensuring high image resolution. Wang P (Wang et al., 2018a) et al. found that improper settings of the original parallel cavity convolution expansion rate could easily cause a "grid effect," as shown in Figure 7A.

A reasonable expansion rate setting should be as shown in Figure 7B, which not only avoids the loss of relevant information but also captures the target context at different scales. Let's define the maximum distance between the nonzero values of the convolution kernels of the ith layer as follows:

$$M_i = max[M_{i+1} - 2r_i, M_{i+1} - 2(M_{i+1} - r_i), r_i](M_n = r_n) \quad (1)$$

According to the literature (Wang et al., 2018a), the void convolution's growth rate should adhere to the following theory: If the expansion rates for N convolutions and void convolutions of size K*K are figured to be [r1, r2,…, ri,…, rn], then the formula satisfies M_2 k, where ri de-notes the ith void convolution's expansion rate and M_i denotes the ith layer's maximum

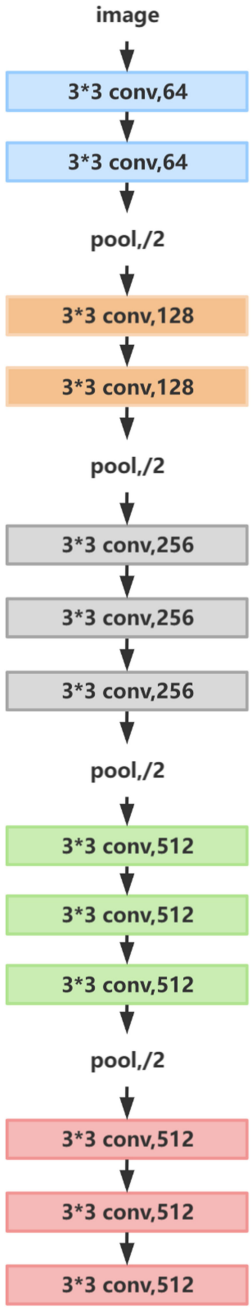


FIGURE 5
Optimised backbone feature extraction module.

expansion rate. There shouldn't be a common factor relationship among the growth rates of the same group (e.g., 2, 4, 8, etc.), as this will still result in lattice effects. For example, when K = 3 and r = [1, 2, 5], no lattice effect will occur after validation; however, when K = 3 and r = [1, 2, 9], which do not meet the requirements after validation, a lattice effect will occur.

Therefore, this paper follows the above design principles and uses multiple parallel convolution layers with different sampling rates. Experimentally, it resets a set of null convolutions with expansion rates of 1, 2, 7, and 15 (M_i = 3 = k). The features extracted from each sampling rate are processed separately and fused into the final result. The adjusted ASPP module is shown in Figure 8.

As shown above, after the fourth block, the parallel architecture is added. The left side of the parallel part consists of a 1x1 convolution, three 3x3 convolution kernels (with void rates of 2, 7, and 15), and pooling operations. The right side represents the image-level features, wherein the features are globally pooled on average, convolved, and then fused. After parallelization and subsequent convolution with 256 1x1 convolution kernels (the convolution layer is followed by a BN layer), the resulting feature map is upsampled to the desired feature dimension.

2.4.4 Integrating efficient attention

The influence of environmental factors such as temperature, humidity, and light intensity leads to variations in the growth states of each Betula luminifera plant. Consequently, there are difficulties in acquiring features. The attention mechanism, inspired by the human attention mechanism, focuses on important information features. The objective of this study is to achieve high-precision segmentation of Betula luminifera stems and leaves. By focusing on different feature pheromones of stems and leaves to achieve a higher precision segmentation effect, cross attention (CCA) is introduced to enable the network to learn more interested regions, thus avoiding the loss of too much semantic information. This enhancement leads to improved segmentation performance of the model.

The Non-local approach is proposed to address the problem of long dependencies (Wang et al, 2021). In this case, the CCA module (Huang et al., 2018) replaces the global attention mechanism in Non-local with a cross-shaped attention mechanism. This modification allows individual pixels to obtain global contextual dependencies through the cross module, thanks to a double-loop operation. As a result, it effectively enhances feature extraction and achieves leading performance in segmentation-based benchmarks. Moreover, the CCA module is GPU memory friendly, providing a significant solution to the issue of excessive parameters in the UNet network and the resource consumption during the model training process.

Compared to Non-local, CCA reduces the FLOPS by 85%. The input image undergoes feature extraction by the backbone network, and the fused CCA not only mitigates the loss of local information but also captures long-distance global information, thereby improving the network's feature extraction capability.

In summary, this paper introduces the CCA module in the decoding part, which operates through the attention mechanism illustrated in Figure 9.

2.4.5 Optimisation of loss functions

Unlike conventional semantic segmentation of objects, the stem and leaf segmentation of Betula luminifera has fewer categories, which include three parts: stem, leaf, and background. The other two categories are more challenging to distinguish overall due to the uncertainty in the proportion of different plant length distributions. They are also several orders of magnitude smaller than the spatial occupation of the background. There is a well-known drawback to

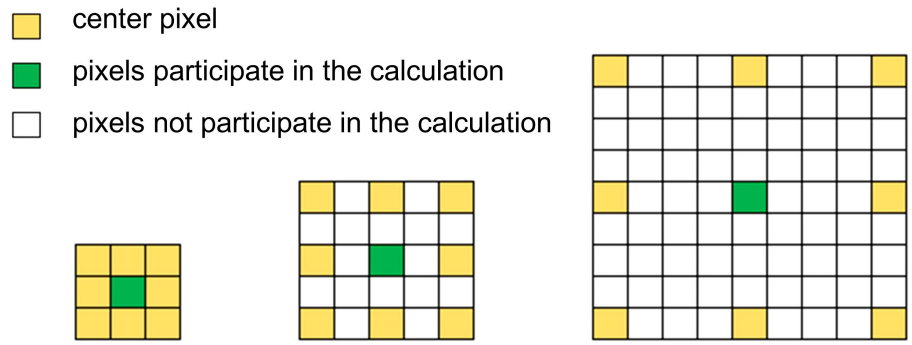


FIGURE 6
Field of perception for convolution at different expansion rates R.

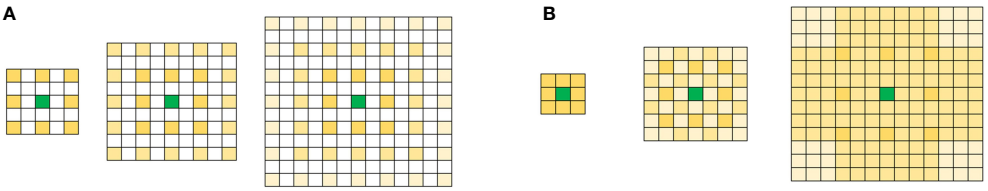


FIGURE 7
Null convolution **(A)** Null convolution "lattice effect".**(B)** Combined hole convolution with reasonable expansion rate.

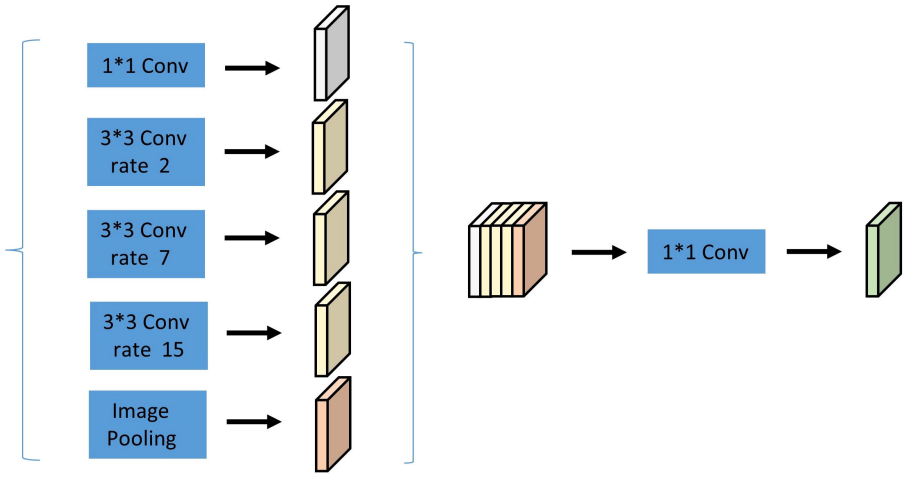


FIGURE 8
Hollow Pyramid Pooling Module.

the highly unbalanced problem in that it assumes all samples and categories are of equal importance. This assumption typically leads to instability in training and results in decision boundaries that are biased towards the more numerous categories.

To address the above imbalance problem and consider the image as a whole, this paper combines Dice loss (Ma et al., 2021) with Boundary loss (Kervadec et al., 2019) and introduces a compound loss function, Dice_Boundary, for optimizing the loss of the training model. The function is defined as follows:

$$l_{DiceBoundary} = l_{Dice} + l_{Boundary} \quad (2)$$

An ensemble similarity measure function called the former Dice loss function is utilised to lessen the inaccuracy between the segmented and labelled images. It is typically used to determine

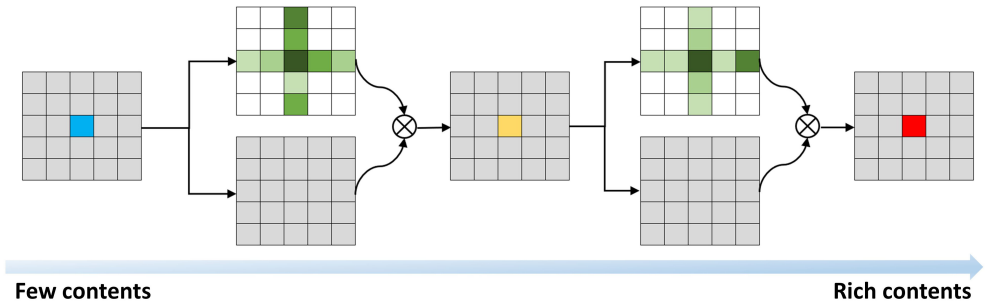


FIGURE 9
CCA module.

how similar two samples are, producing results in the [0,1] range. Greater overlap between the expected and actual results, which denotes improved performance, is shown by a larger score. In contrast, a lower loss value is preferred, as shown in equation (3).

$$l_{Dice}=1-\frac{2\sum_{c=1}^{C}\sum_{i=1}^{N}g_i^c s_i^c}{\sum_{c=1}^{C}\sum_{i=1}^{N}g_i^c+\sum_{c=1}^{C}\sum_{i=1}^{N}s_i^c} \quad (3)$$

where denotes the predicted label, denotes the true label, N is the number of pixels in the image, and C is the number of classifications. The latter Boundary loss function takes the form of a contour space rather than a region metric to alleviate the difficulties of the highly unbalanced problem. In addition, the Boundary loss supplements the region in-formation and is usually applied to segmentation tasks with a high imbalance, as shown in equation (4).

$$\mathbf{Dist}(\partial G,\partial S)=\int_{\partial G}||y_{\partial S}(p)-p||^2 dp \quad (4)$$

where 'p' denotes a point on the edge of 'A', and 'q' is the corresponding point on 'B', in other words, 'q' represents the intersection with 'A' found at point 'p' on 'B'. '||.||' de-notes the paradigm.

$$\mathbf{Dist}(\partial G,\partial S)\approx 2\int_{\Delta S} D_G(q)dq \quad (5)$$

where ΔS denotes the distance between even contours, and D is the distance map relative to the boundary, in other words, D denotes the distance between any point q and the nearest point on the contour. Equation (5) is used to derive equation (6).

$$\mathbf{Dist}(\partial G,\partial S)=2(c(q)s(q)dq-\int_{\Omega}\varnothing_G(q)g(q)dq) \quad (6)$$

$$l_{Boundary}=\int_{\Omega}\varnothing_G(q)S_\theta(q)dq \quad (7)$$

where S is the binary indicator function of the region S and is the horizontal set representation of the boundary. For S = [value], the softmax output of the network replaces [value] in equation (7) to obtain the boundary loss of equation (6).

3 Experimental analysis and discussion

3.1 Data preprocessing

3.1.1 Shearing

Image clipping is an operation that cuts out a specific area in an image. Clipping can remove noise and avoid large-area image exposure problems; highlight areas of interest, reduce interference, and make the image more focused on the target area; resize, adjust For a specific input size, it facilitates the processing of the model. The images collected in this research have problems such as overexposure of light, unprotrusive regions of interest, images taken by mobile phones, and inconsistent image sizes. Therefore, the cutting method is used to preprocess the images to meet the needs of the algorithm and improve the efficiency of image processing. quality and effect.

3.1.2 Median filtering

Images are subject to temperature and humidity, magnetic fields, losses during signal transmission, vibration noise, etc., during formation or transmission, resulting in degradation of image quality and distortion of the final imaging results. These factors inevitably have an impact on later image analysis and research. In the image acquisition of this study, the irradiation of the physical light source and natural light at different times of the day caused greater disturbance to the subsequent image processing work. To reduce the effects of noise, noise reduction was applied using image filtering processing.

Based on the characteristics of the collected Betula luminifera images, a spatial domain filtering process is used. Spatial domain filtering consists of linear and non-linear filtering. In this study, the final approach utilizes the median filtering algorithm in non-linear filtering to achieve noise reduction processing of the images. Median filtering has the effect of removing impulse noise and preserving edge details of the image.

As a typical type of non-linear smoothing filter, the basic principle of median filtering is to replace the value of a point in a digital image or digital sequence with the median of the values of the points in a neighborhood of that point. This allows pixels with a relatively large difference in gray value compared to the

surrounding pixels to be replaced, effectively eliminating isolated noise points. The median filtering formula can be expressed as:

$$F_i = Med\{F_{i-m}, \ldots, F_i, \ldots, F_{i+m}\} \quad (8)$$

(For a one-dimensional sequence, taking p numbers for median processing, m = (p-1)/2.)

For two-dimensional images undergoing median filtering, the filter window is also two-dimensional. This window can take on various shapes, including lines, squares, circles, crosses, and so on. The formula for two-dimensional median filtering can be expressed as:

$$F_{i,j} = Med\{f_{i,j}\} \quad (9)$$

(Med for the number of filter windows)

Mean filtering is a common linear filtering algorithm that determines the average of the noise components, as the name suggests. In the procedure, the average value of the adjacent pixels in a template is used to replace the original pixel value. The target pixel itself and the eight pixels around it that are centred on it make up the template. Unlike mean filtering, this method preserves the image's edge information for further image processing while also addressing the problems of blurring image details and loss of features (Figure 10A).

Bilateral filtering, like median filtering, is a non-linear filtering method that combines the spatial proximity of an image and the similarity of pixel values in a compromise process. It takes into account both spatial domain information and grayscale similarity to achieve edge-preserving denoising. In this study, the output of the median filter is almost identical to the original image. The bilateral filter has little effect on the pixel values of the edges. However, due to the preservation of too much high-frequency information, the noise in the color image is not cleanly filtered, resulting in a lack of detailed texture, as seen in the figure with the missing leaf veins (Figure 10B).

Gaussian filtering is a linear smoothing filter that is suitable for removing Gaussian noise and is widely used in the noise reduction process of image processing. In layman's terms, Gaussian filtering is the process of weighted averaging of the entire image. Each pixel's value is obtained through a weighted average of its own value and the values of other pixels in its neighborhood. While the difference between the Gaussian filtered image and the median filtered image may not appear significant, the image processing speed is much slower with Gaussian filtering compared to median filtering (Figure 10C).

3.2 Experimental environment parameter settings and evaluation indicators

The experimental device has an NVIDIA GeForce RTX 3060 GPU, a 12th Gen Intel(R) Core(TM) i5-12400 2.50 GHz processor, and 12 GB of video memory. Windows 64-bit with CUDA 11.2 is the operating system used in the software environment. The deep learning framework is PyTorch, and Python is the programming language. As the deep learning framework, PyTorch was employed. A batch training approach was used to train the network. For



FIGURE 10
Comparison of filtering methods. **(A)** Median filtering VS Mean filtering. **(B)** Median filtering VS Bilateral filtering. **(C)** Median filtering VS Gaussian filtering.

training, different batches of the training and validation sets were created. the network model's traversal of every image in the training set is calculated as one iteration. The network model is initialized using pre-trained weights to initialize the backbone network. The initial learning rate is 0.001 and is optimized by the Adam algorithm to calculate the adaptive learning rate for each weight parameter.

In order to quantify the segmentation effect of the segmentation method in this article on light bark images and compare the segmentation performance of different methods, the evaluation criteria introduced in this article mainly include category average pixel accuracy (mPA), precision (Precision), average intersection over union (mIoU) and F1 score. Among them, FN means that the model incorrectly classified it as a negative example, but it is actually a positive example; FP means that the model incorrectly classified it as a positive example, but it is actually a negative example; TP means that the model correctly classified it as a positive example, but it is actually a positive example; TN means The model correctly classified them as negative examples, but they were actually negative examples. The detailed division is shown in Table 2.

3.2.1 Mean Pixel Accuracy

Mean pixel accuracy(mPA), Calculate the proportion of pixels per class that are correctly classified. mPA is expressed as:

$$Pi = \frac{TP}{TP + TF} \tag{10}$$

(Note: Pi indicates pixel accuracy for each category)

$$\mathbf{mPA} = \frac{\mathbf{sum(Pi)}}{\mathbf{class\ number}} \tag{11}$$

3.2.2 Precision

Precision, also known as the accuracy rate, measures the proportion of correct predictions (true cases) out of all predictions that are positive:

$$Precision = \frac{TP}{TP+FP} \tag{12}$$

3.2.3 Mean intersection over union

Mean Intersection over Union (mIoU) is the ratio of the intersection between the true label value and the predicted value to the union between the true value and the predicted value. mIoU is expressed as follows:

$$mIoU = \frac{TP}{FP+FN+TP} \tag{13}$$

TABLE 2 Confusion matrix for classification results.

The Real Deal	Predicted results	
	Positive examples	Counter examples
Positive examples	True example	False counterexample
Counter examples	False example	true counterexample)

3.2.4 F_1 Score

The F1-Score, also known as the Balanced F1-Score, is defined as the average sum of the precision and recall rates. The expression for the F-Score is:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{14}$$

where Recall is expressed as the recall rate, which measures the probability of a category being correctly predicted among the true values, and the expression for Recall is:

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

FLOPs/G denotes the model complexity of the network and Params/M denotes the total number of parameters of the entire network model.

3.3 Backbone network performance comparison

In response to the problem of poor segmentation performance of stem and leaf details and edge parts in the experiment of Betula luminifera stem and leaf segmentation, two schemes are proposed to modify the UNet backbone network using the deep feature extraction network VGG16 and the feature extraction network ResNet50 with a residual network structure. Two types of UNet networks are trained under the same conditions to segment the stems and leaves of Betula luminifera, respectively. By comparing the effects of the two backbone networks on the UNet algorithm in stem and leaf segmentation, it is shown that UNet equipped with the VGG16 backbone network has a more precise detection effect, which is closer to the accuracy of the segmentation algorithm proposed in this article. Please refer to UNet-3 in Table 3.

The training loss curves and validation loss curves for the two backbone networks are displayed in Supplementary Figures 11A, B, respectively. The 100 training rounds' worth of loss curves are shown in Supplementary Figure 11A. Supplementary Figure 11A shows that the loss values of both backbone networks on the training set first decrease fast and then roughly level out, indicating better convergence. On the other hand, the validation loss curve offers a better representation of the network's performance on brand-new, untested data.

The validation loss values for ResNet50 swing noticeably in the later phases of training, as seen by the loss curves for the validation set in Supplementary Figure 11B. This suggests that the network's ResNet50 structure has been severely overfitted. The validation loss

TABLE 3 Accuracy comparison experiments for different backbone networks.

Method	VGG16	ResNet50	mIoU	mPA
UNet-1		√	85.34%	91.27%
UNet-2	√		86.69%	92.52%
UNet-3	√		87.50%	93.36%

values of VGG16, however, stay smooth and essentially converge in the late training period, demonstrating that the model has attained a respectable training effect.

3.4 Ablation experiments

In order to verify the improvement of segmentation performance of improved UNet, this paper conducts segmentation performance ablation experiments on the self-made optical Betula luminifera data set. Replace the backbone feature extraction network of UNet itself with VGG16, add the ASPP module between the backbone feature extraction module and the enhanced feature extraction module, add the CCA attention module to each feature layer of the enhanced feature extraction module, and introduce Dice_Boundary Loss, setting Nine model experimental comparisons are detailed as follows:

UNet-D: In the traditional UNet model, the Dice_Boundary Loss loss function is introduced;

UNet-VD: Based on UNet-D, the backbone feature extraction network is replaced by VGG16;

UNet-AD: Based on UNet-D, the ASPP module is added between the backbone feature extraction module and the enhanced feature extraction module;

UNet-CD: Based on UNet-D, the CCA attention module is added after strengthening each feature layer of the feature extraction module;

UNet-VAD1: Based on UNet-VD, the ASPP (1,6,12,18) module is added between the backbone feature extraction module and the enhanced feature extraction module;

UNet-VAD2: Based on UNet-VD, the ASPP(1,2,4,8) module is added between the backbone feature extraction module and the enhanced feature extraction module;

UNet-VAD3: Based on UNet-VD, the ASPP(1,2,7,15) module is added between the backbone feature extraction module and the enhanced feature extraction module;

UNet-VCD: Based on UNet-VD, the CCA attention module is added after strengthening each feature layer of the feature extraction module;

UNet-VACD: Based on UNet-VAD, the CCA attention module is added after strengthening each feature layer of the feature extraction module, which is the method in this article.

Table 4 details the performance gains obtained through the combination of different modules. The input size of the training images is 512*512. From the results of the ablation experiment, it can be seen that comparing UNet and UNet-D, the mIoU, mPresicion, and mPA indicators have improved. It shows that introducing the Dice_Boundary Loss loss function into the model has certain benefits in improving the accuracy of the model.

Comparing UNet and UNet-VD, the index values of UNet-VD in mIoU, mPresicion, and mPA have all increased, confirming that replacing the feature extraction network with VGG16 is the main factor in improving the model segmentation performance.

Comparing the three models UNet-VAD1, UNet-VAD2, and UNet-VAD3, the indicators in mIoU, mPresicion, and mPA are all higher than UNet-D. The index of UNet-VAD3 is the best among the three, that is, the segmentation performance of this model is better, indicating that the dilation rate of (1, 2, 7, 15) used in dilated convolution is the main factor in improving segmentation performance.

Compared with UNet-D, UNet-VD, UNet-AD, and UNet-CD, the mIoU, mPresicion, and mPA indicators have all improved. It shows that both dilated convolution ASPP and cross-attention mechanism CCA can improve the segmentation performance of the model to a certain extent. Among them, UNet-AD has a higher degree of improvement in performance indicators than the other three models. Therefore, the addition of ASPP is an overall improvement. The workhorse of model segmentation performance.

Comparing UNet-VCD and this article's method UNet-VACD, the replacement of the backbone feature network and the addition of CCA did not significantly improve the efficiency. The introduction of ASPP improved the segmentation accuracy of the model to a certain extent. This comprehensive evaluation process allows us to evaluate the performance and stability of the proposed AC-UNet model more efficiently.

3.5 Performance comparison of different segmentation models

The method presented in this paper is compared with several advanced segmentation methods, namely PSPNet, DeepLabV3, UNet and Swin-UNet, on the Betula luminifera dataset. To ensure the rigor and fairness of the comparison experiments, all segmentation methods employ the same experimental equipment, uniform image size, identical parameter set-tings, and consistent training processes. The differences in data, including mIoU, mPA, FLOPs, and Params, between the methods are evaluated and compared. As depicted in Table 5, the AC-UNet plant stem and leaf segmentation model, designed in this paper, exhibited the best segmentation performance.

Table 5 show the results of comparing AC-UNet with the remaining four networks. The mIoU, mPA, Precision, and F1, all four metrics of the method model in this paper are higher than the remaining three networks, with F1 = 93.52%, indicating the

TABLE 4 Results of ablation experiments.

Method	mIoU	Presicion	mPA
UNet	84.45%	92.02%	91.11%
UNet-D	86.03%	92.10%	91.73%
UNet-VD	86.69%	92.49%	92.52%
UNet-AD	86.71%	92.32%	92.48%
UNet-CD	86.68%	92.30%	92.45%
UNet-VAD1	86.95%	92.79%	92.54%
UNet-VAD2	87.08%	92.79%	92.71%
UNet-VAD3	87.16%	92.75%	92.84%
UNet-VCD	86.80%	92.60%	92.53%
UNet-VACD	87.50%	93.69%	93.36%

10.3389/fpls.2023.1268098

TABLE 5 Comparison of evaluation indicators for different segmentation models.

Method	mIoU	mIoU_stem mIoU_leaf	mPA	Precision	F1	FLOPs/G	Params/M
PSPNet	58.76%	26% 52%	73.24%	66.90%	69.93%	6.031	2.376
DeepLabV3	82.13%	64% 83%	91.47%	87.73%	89.56%	52.875	5.814
UNet	84.45%	70% 85%	91.11%	90.63%	92.38%	451.706	24.891
Swin-UNet	79.02%	57% 80%	85.99%	88.73%	87.36%	91.497	41.342
AC-UNet	87.50%	75% 88%	93.36%	93.69%	93.52%	482.001	34.244

excellent learning performance of the model. More importantly, AC-UNet performed best in the metric of mIoU, achieving considerable improvement over the other network models. It is evident from Supplementary Figure 12 that Ours outperforms PSPNet, DeepLabV3 and UNet by around 28.32%, 4.95%,3.05% and8.48%, respectively, with segmentation values for stem improving by 49%, 11%, 5% and 18%, respectively, and segmentation values for leaf improving by 36%, 5%, 3% and 8%, respectively.

The DeepLabV3 and UNet classical networks' respective parameters for the network are 2.376M, 5.184M, and 24.891M, respectively, with the exception of PSPNet, as shown by the four different models. The difference in mPA was not perceived to be very significant, despite the fact that the number of parameters changed significantly, particularly for DeepLabV3 and classical UNet, which have respective mPAs of 91.47% and 91.11%. In addition, Swin-UNet has the smallest model complexity and can segment stems and leaves in a short time. However, in addition to its speed advantage, it fails to provide better segmentation results. It is evident from Table 5 that DeepLabV3 and UNet have a lot more parameters that the network used in this study—5.814 and 24.891, respectively.

This indicates that the segmentation effect is not necessarily enhanced due to network lightweighting in the case of a small number of parameters. The mIoU, mPA and Precision, of the equally lightweight PSPNet are only 58.76%, 73.24% and 69.93%, respectively, which are less effective in segmentation. It can be seen that the lightweighting changes are not necessarily applicable to the needs of the segmentation algorithm in this paper.

The relationship between segmentation losses and the total number of iterations was discovered using the earlier suggested technique for segmentation training on Betula luminifera data, as shown in Supplementary Figure 13A, B. It can be clearly seen from Supplementary Figure 13A that AC-UNet has a lower loss value in the initial stage. The algorithm has excellent phenotypic fitting ability and better generalization ability. As can be seen from Supplementary Figure 13B. The segmentation loss becomes lower and smaller as the number of iterations rises. For PSPNet, the loss suddenly decreases after 50 iterations and then tends to level off. At around 60 repetitions, the segmentation loss for the remaining three networks starts to stabilise. It is essential to note that the network utilised in this paper's technique has a tiny initial loss value, showing that its convergence impact is far better than that of the other networks. This shows that the network presented in this paper's convergence impact has a good convergence effect.

Supplementary Figure 14 is an example of model segmentation output, where A is the original image, B is the label, and the rest C, D, and E are the fusion prediction map of the label and the original image, the stem prediction map and the leaf prediction map. This example segmented output image shows images of three different morphologies of Betula luminifera, including leaves and stems. The image is segmented into regions where stems are marked in green and leaves in red. However, there are faults or fuzzy boundaries in some places, such as the junction of stems and leaves, and the edges of leaves. The model may not be able to clearly distinguish them due to the occlusion of the stems by the leaves of the plants. Overall, the algorithm in this paper performs well on the example image, and most of the stems and leaves are accurately segmented. However, it should also be noted that for some complex areas, further optimization is still required.

The segmentation has been portrayed using a custom dataset in order to demonstrate how the performance of the method in this research differs from that of other methods. The dataset's visualisation results are shown in Supplementary Figure 15, where (A) depicts the original image, (B) the label, and the images that remain (C), (D), (E), (F) and (G), respectively, correspond to PSPNet, DeepLabV3, UNet, Swin-UNet, and the approach suggested in this paper, AC-UNet.

Supplementary Figure 15C-PSPNet shows a substantial inaccuracy in segmentation accuracy. The image is barely segmented out in the C column, with only vaguely distinguishable stems and leaves. It has no reference point compared to the rest of the methods. In Supplementary Figure 15D-DeepLabV3, the segmentation effect is rough, and the branch and stem parts appear disconnected. A comparison reveals that the segmentation treatment of the detail part in this paper's method is closer to the

label, providing higher segmentation accuracy. Supplementary Figure 15E-UNet and Supplementary Figure 15G-AC-UNet both achieve better segmentation results, but ours demonstrates more pragmatic segmentation results for mutilated leaves and better detailing of the edges of leaves and stems. Supplementary Figure 15F -Swin-UNet can be clearly seen with the naked eye, and some stem and leaf segmentations appear to be mis-segmented, segmented and stacked. In contrast, AC-UNet shows better results in stem and leaf details, incomplete leaf segmentation, and leaf edge segmentation. Therefore, the performance of the algorithm in this paper is more stable and less prone to the above problems, which makes it a more reliable algorithm choice. It can be seen from the segmentation effect that AC-UNet has a higher-precision segmentation benefit in the stem and leaf segmentation task. This benefit has obvious advantages in the later measurement of phenotype correlation coefficients, ensuring that it can be used in specific environments. Stability and reliability.

Methods of this paper demonstrates better and superior segmentation results for plant stems and leaves.

3.6 Phenotypic analysis of Betula luminifera

Select any 15 groups of plant objects from the Betula glabra dataset and evaluate and predict the crown area of the plants. The calculation of the crown area in this paper is evaluated by referencing half of the product of the plant height and crown width. The segmentation network proposed in this paper is used for image prediction, and the predicted image is obtained as shown in Supplementary Figure 16A. Using HSV color threshold segmentation, select the red threshold part in the mask image for segmentation and extraction. Refer to Table 1 in Section 2.3 for the values. The segmentation effect is shown in Supplementary Figure 16C. Finally, through OTSU, the leaf part we need, which is the foreground part in Supplementary Figure 16C, is segmented and extracted. Lastly, the corresponding feature grayscale binary image is obtained, as shown in Supplementary Figure 16D. Based on the binary image, the crown area of the plant is calculated.

Take three groups of plants with different shapes from the visual analysis for binarization, and analyze the proportion of stems and leaves in the image (because the background occupies a large area, we take 10 as the whole image). At the same time, due to the segmentation effect of Swin-UNet There is a large error and overlap, so Swin-UNet is not included in the comparison. It can be seen from the stacked Supplementary Figure 17 (where A, B, C, and D respectively represent PSPNet, DeepLabV3, UNet, and the method AC-UNet in this article, and 1, 2 and 3 are label maps) that in each subgroup of A, B, and C, except for B3 They are all significantly different from the proportion of stems and leaves in the label map. The distribution of the proportion of stems and leaves in each group of category D is approximately the same as that in the label map.

We determined the proportionate relationship between the image and the actual plant based on the height and length of the crown cross-section of the actual, measured plant. The actual values and the expected results were then compared and analysed. The data's regression analysis reveals that the model fits the data well because the R2 value for leaf is 0.99882, which is greater than 0.8 and near to 1. The 15 data points are all close to the regression line in Supplementary Figure 18, illustrating a high positive correlation between the predicted and actual values.

This calculation method eliminates the need for destructive sampling of plants and facilitates the development of continuous dynamic observation of the same research object while reducing the time required for determination in terms of manpower on a large number of samples compared to conventional methods. These methods include the squared paper method, the paper-cut weighing method, the leaf-area-meter determination method, and the image-processing method through scanning and photographing. The design incorporation of CCA and ASPP in the network expands the contextual horizon in the deep learning process, allowing the network to learn more detailed information during the training process, thereby facilitating better plant segmentation performance. Further-more, it is more evident from the segmentation results that the fusion of attention mechanism and ASPP in UNet leads to further enhancement in refining plant image details. This enhancement provides strong support for the final realization of plant phenotypic traits acquisition

4 Discussion

Earlier vision-based segmentation research was more applied in remote sensing images (urban, agriculture and forestry) and crops. Hong and his collaborators (Hong et al., 2021), in view of the limitations of convolutional neural networks (CNNs) when sampling, incorporated an improved batch GCN (miniGCN) and proposed an end-to-end network FuNet with a fusion strategy to ensure network stability On the basis of reducing computational costs, high-efficiency remote sensing image segmentation is achieved, which opens up new ideas for solving restrictive problems in the segmentation field. In addition, Hong et al. (Hong et al., 2023)also introduced HighDAN, a high-resolution domain-adaptive network architecture that can solve cross-city or region problems. The network achieved the best segmentation performance on the constructed multi-modal remote sensing benchmark data set (C2Seg data set). Solve bottlenecks that hinder urban planning and development. Some of them have made corresponding improvements to the UNet model and achieved good results (Genze et al., 2023; Wang et al., 2023). However, some advanced segmentation models based on deep learning are not completely suitable for plant stem and leaf segmentation. This is because plant stems and leaves contain more feature information than the plant as a whole, but occupy a smaller proportion of pixels in the image, making segmentation difficult.

In this study, we demonstrated the advantages of AC-UNet in stem-leaf segmentation in Betula luminifera populations and the convenience it brings in the acquisition of plant phenotypic traits in the later stage, which is expected to be used in the field of tree species segmentation and plant phenotypic traits acquisition make a certain contribution. This is of great significance for early plant breeding and species health assessment. The mIoU value of AC-

UNet in the segmentation of plant stems and leaves reached 87.5%, confirming that the method overcomes the effect of extracting stem and leaf detail information to a certain extent due to problems such as missing edge information, faults at the junction of stems and leaves, and uneven image samples. Almost difficult. The proposed method still faces some limitations: increasing the data of Betula luminifera in other different geographical areas, enhancing the rationality and universality of the research; AC-UNet performed well in extracting details of, conifers, shrubs, etc.) needs to be further explored; study the lightweight of the network, and improve its efficiency on the basis of ensuring the segmentation accuracy of the network model; solve the problem of leaf occlusion during the growth of plants, in order to further improve the performance of the network model. Acquisition accuracy of type traits.

5 Conclusion and outlook

The existing semantic segmentation network cannot achieve better segmentation of plant stems and leaves. The extraction of plant edges, joints, and details is poor, and accurate segmentation of organs such as stems and leaves is not possible. In order to solve these problems, this paper proposes a new segmentation network called AC-UNet, which addresses the segmentation challenges specific to Betula luminifera's stem and leaf organs. Considering the unique characteristics of leaf edge details and stem-leaf connections in the segmentation prediction process of Betula luminifera's stem and leaf organs, the AC-UNet algorithm, an improved version of UNet, is introduced. This algorithm aims to enhance the overall segmentation accuracy by addressing the issues of insufficient edge information and disconnections in conventional segmentation algorithms. Additionally, a composite loss function called Dice_Boundary, which combines the Dice and Boundary metrics, is introduced at the back-end of the network to tackle the problem of imbalanced image samples.

This paper focuses on experimental observations using Betula luminifera seedlings planted in the northwest of Zhejiang Province. A performance comparison is conducted among different models including PSPNet, DeepLabV3, and UNet, and based on the results, an improved AC-UNet model is designed on the foundation of UNet. The experimental results demonstrate that AC-UNet significantly enhances the accuracy of stem and leaf segmentation for Betula luminifera, achieving an mIoU value of 87.50% and accurately extracting detailed parts of the plant. Follow-up research will focus on planting Betula luminifera seedlings and other tree species (broad leaves, conifers, shrubs, etc.) in different geographical locations to support the universal applicability of this algorithm in obtaining plant phenotypic information. Future applications will expand from stem and leaf segmentation to tree segmentation, botany, etc., and provide new core technology research paths.

Data availability statement

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

Author contributions

XY: Supervision, Writing – review & editing. JW: Software, Writing – original draft. PW: Conceptualization, Methodology, Writing – review & editing. GW: Formal Analysis, Writing – review & editing. LM: Resources, Writing – review & editing. XL: Data curation, Writing – review & editing. HL: Visualization, Writing – review & editing. HH: Funding acquisition, Writing – review & editing. EL: Data curation, Writing – review & editing. BM: Investigation, Writing – review & editing. CL: Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This research was funded by National Key R&D Plan Project Sub Project, grant number 2021YFD2200304-2, and Zhejiang Province Key R&D Plan Project, grant number 2021C02037.

Acknowledgments

We are appreciative of the reviewers' valuable suggestions on this manuscript and the editor's efforts in processing the manuscript.

Conflict of interest

Author CL is employed by Hangzhou Ganzhi 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.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2023.1268098/full#supplementary-material

References

Abinaya, A., and Roomi, S. M. M. (2016). "asmine flower segmentation: A superpixel based approach," in *2016 International Conference on Communication and Electronics Systems (ICCES)*. 1–14. doi: 10.1109/CESYS.2016.788992

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., and Tian, Q.(2021). Swin-unet: unet-like pure transformer for medical image segmentation. *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)* 13803, 205–218. doi: 10.1007/978-3-031-25066-8_9

Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2018). DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 834–848. doi: 10.1109/TPAMI.2017.2699184

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017). *Rethinking atrous convolution for semantic image segmentation*. Available at: http://arxiv.org/abs/1706.05587.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., et al. (2009). "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. doi: 10.1109/CVPR.2009.5206848

Fu, L., Tola, E., Al-Mallahi, A., Li, R., and Cui, Y. (2019). A novel image processing algorithm to separate linearly clustered kiwifruits. *Biosyst. Eng.* 183, 184–195. doi: 10.1016/j.biosystemseng.2019.04.024

Genze, N., Wirth, M., Schreiner, C., Ajekwe, R., Grieb, M., and Grimm, D. G. (2023). Improved weed segmentation in UAV imagery of sorghum fields with a combined deblurring segmentation model. *Plant Methods* 19, 1–12. doi: 10.1186/s13007-023-01060-8

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. *Indian J. Chem. - Sect. B Org. Med. Chem.* 45, 1951–1954. doi: 10.1002/chin.200650130

Hong, D., Gao, L., Yao, J., Zhang, B., Plaza, A., and Chanussot, J. (2021). Graph convolutional networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 59, 5966–5978. doi: 10.1109/TGRS.2020.3015157

Hong, D., Zhang, B., Li, H., Li, Y., Yao, J., Li, C., et al. (2023). *Cross-City Matters: A Multimodal Remote Sensing Benchmark Dataset for Cross-City Semantic Segmentation using High-Resolution Domain Adaptation Networks*. Available at: http://arxiv.org/abs/2309.16499.

Huang, Z., Wang, X., Wei, Y., Huang, L., Shi, H., Liu, W., et al. (2018). *CCNet: criss-cross attention for semantic segmentation*. Available at: http://arxiv.org/abs/1811.11721.

Kalyoncu, C., and Toygar, Ö. (2015). Geometric leaf classification. *Comput. Vis. Image Underst.* 133, 102–109. doi: 10.1016/J.CVIU.2014.11.001

Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., and Ben, A. I. (2019). Boundary loss for highly unbalanced segmentation. In: *International conference on medical imaging with deep learning*.. 285–296.

Li, Z., Guo, R., Li, M., Chen, Y., and Li, G. (2020). A review of computer vision technologies for plant phenotyping. *Comput. Electron. Agric.* 176, 105672. doi: 10.1016/j.compag.2020.105672

Ma, J., Chen, J., Ng, M., Huang, R., Li, Y., Li, C., et al. (2021). Loss odyssey in medical image segmentation. *Med. Image Anal.* 71, 102035. doi: 10.1016/j.media.2021.102035

Okyere, F. G., Cudjoe, D., Sadeghi-Tehran, P., Virlet, N., Riche, A. B., Castle, M., et al. (2023). Machine learning methods for automatic segmentation of images of field- and glasshouse-based plants for high-throughput phenotyping. *Plants* 12, 2035. doi: 10.3390/plants12102035

Pape, J.-M., and Klukas, C. (2015a). "3-D Histogram-Based Segmentation and Leaf Detection for Rosette Plants." in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. (Springer Verlag), 61–74. doi: 10.1007/978-3-319-16220-1

Pape, J.-M., and Klukas, C. (2015b). Utilizing machine learning approaches to improve the prediction of leaf counts and individual leaf segmentation of rosette plant images. *Proc. Comput. Vis. Probl. Plant Phenotyping* 3, 1–12.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: Convolutional networks for biomedical image segmentation," in *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Springer Verlag), 234–241. doi: 10.1007/978-3-319-24574-4_28

Sadeghi-Tehran, P., Virlet, N., Ampe, E. M., Reyns, P., and Hawkesford, M. J. (2019). DeepCount: in-field automatic quantification of wheat spikes using simple linear iterative clustering and deep convolutional neural networks. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.01176

Scharr, H., Minervini, M., French, A. P., Klukas, C., Kramer, D. M., Liu, X., et al. (2016). Leaf segmentation in plant phenotyping: a collation study. *Mach. Vis. Appl.* 27, 585–606. doi: 10.1007/s00138-015-0737-3

Simonyan, K., and Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. Available at: http://arxiv.org/abs/1409.1556.

Tamvakis, P. N., Kiourt, C., Solomou, A. D., Ioannakis, G., and Tsirliganis, N. C. (2022). *Semantic image segmentation with deep learning for vine leaf phenotyping*. Available at: http://arxiv.org/abs/2210.13296.

Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., et al. (2018a). Understanding convolution for semantic segmentation. *Proc.-2018 IEEE Winter Conf. Appl. Comput. Vision*, WACV 2018 2018-January, 2018.00163. doi: 10.1109/WACV.2018.00163

Wang, H., Ding, J., He, S., Feng, C., Zhang, C., Fan, G., et al. (2023). MFBP-UNet: A network for pear leaf disease segmentation in natural agricultural environments. *Plants* 12, 3209. doi: 10.3390/plants12183209

Wang, X., Girshick, R., Gupta, A., and He, K. (2021). Non-local Neural Networks. *Compos. Math.* 1079–1119. doi: 10.1112/S0010437X21007144

Wang, Z., Wang, K., Yang, F., Pan, S., and Han, Y. (2018b). Image segmentation of overlapping leaves based on Chan–Vese model and Sobel operator. *Inf. Process. Agric.* 5, 1–10. doi: 10.1016/j.inpa.2017.09.005

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). "Pyramid scene parsing network," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*. (Institute of Electrical and Electronics Engineers Inc), 6230–6239. doi: 10.1109/CVPR.2017.660

Zhou, J., Tardieu, F., Pridmore, T., Doonan, J., Reynolds, D., Hall, N., et al. (2018). Plant phenomics:: history, present status and challenges. *J. Nanjing Agric. Univ.* 41, 580–588. doi: 110.7685/jnau.201805100

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 11 December 2023
DOI 10.3389/fpls.2023.1323453

Check for updates

OPEN ACCESS

EDITED BY
Huabing Zhou,
Wuhan Institute of Technology, China

REVIEWED BY
Baofeng Su,
Northwest A&F University, China
Xian Li,
Agricultural Information Institute, Chinese Academy of Agricultural Sciences, China

*CORRESPONDENCE
Yun Shi
shiyun@caas.cn

RECEIVED 17 October 2023
ACCEPTED 22 November 2023
PUBLISHED 11 December 2023

CITATION
Li H, Yang P, Liu H, Liu X, Qian J, Yu Q, Geng C and Shi Y (2023) An improved YOLOv5s model for assessing apple graspability in automated harvesting scene. *Front. Plant Sci.* 14:1323453. doi: 10.3389/fpls.2023.1323453

COPYRIGHT
© 2023 Li, Yang, Liu, Liu, Qian, Yu, Geng and Shi. 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 improved YOLOv5s model for assessing apple graspability in automated harvesting scene

Huibin Li[1], Peng Yang[2], Huaiyang Liu[3], Xiang Liu[3], Jianping Qian[1], Qiangyi Yu[1], Changxing Geng[3] and Yun Shi[1]*

[1]State Key Laboratory of Efficient Utilization of Arid and Semi-arid Arable Land in Northern China, Institute of Agricultural Resources and Regional Planning, Chinese Academy of Agricultural Sciences, Beijing, China, [2]Agricultural Algorithm Research Department, Suzhou Zhongnong Digital Intelligence Technology Co., Ltd, Suzhou, China, [3]School of Mechanical and Electrical Engineering, Soochow University, Suzhou, China

Introduction: With continuously increasing labor costs, an urgent need for automated apple- Qpicking equipment has emerged in the agricultural sector. Prior to apple harvesting, it is imperative that the equipment not only accurately locates the apples, but also discerns the graspability of the fruit. While numerous studies on apple detection have been conducted, the challenges related to determining apple graspability remain unresolved.

Methods: This study introduces a method for detecting multi-occluded apples based on an enhanced YOLOv5s model, with the aim of identifying the type of apple occlusion in complex orchard environments and determining apple graspability. Using bootstrap your own atent(BYOL) and knowledge transfer(KT) strategies, we effectively enhance the classification accuracy for multi-occluded apples while reducing data production costs. A selective kernel (SK) module is also incorporated, enabling the network model to more precisely identify various apple occlusion types. To evaluate the performance of our network model, we define three key metrics: AP_{GA}, AP_{TUGA}, and AP_{UGA}, representing the average detection accuracy for graspable, temporarily ungraspable, and ungraspable apples, respectively.

Results: Experimental results indicate that the improved YOLOv5s model performs exceptionally well, achieving detection accuracies of 94.78%, 93.86%, and 94.98% for AP_{GA}, AP_{TUGA}, and AP_{UGA}, respectively.

Discussion: Compared to current lightweight network models such as YOLOX-s and YOLOv7s, our proposed method demonstrates significant advantages across multiple evaluation metrics. In future research, we intend to integrate fruit posture and occlusion detection to f]urther enhance the visual perception capabilities of apple-picking equipment.

KEYWORDS

apple harvesting, BYOL, attention mechanism, occlusion detection, YOLOv5S

1 Introduction

During the apple maturation season, orchard managers typically employ a significant temporary workforce to ensure the timely harvesting and sale of the apples. However, in recent years, escalating labor costs and the scarcity of manpower have posed significant challenges for these managers (Liu et al., 2019). Consequently, there is an increasing demand for automated apple-picking equipment in the agricultural sector, which represents a pivotal opportunity for the development of such technology. Over the past few decades, apple-picking equipment has garnered substantial attention from both domestic and international researchers (Li et al., 2022). While vision-based apple-picking control technologies have experienced rapid advancement, the hand–eye coordination efficiency of the equipment remains sub-optimal and has been identified as a key factor affecting its performance (Jiao et al., 2020). Occlusion is considered one of the primary challenges in improving visual control technology for apple-picking. This factor has a negative impact, as occlusion by leaves, branches, or other apples can prolong the apple identification time and reduce accuracy.

At present, apple-picking equipment can harvest apples that are unobstructed or merely occluded by leaves; however, apples concealed by branches or other apples pose a significant challenge. During automated harvesting,if the equipment cannot discern the graspability of an apple based on its occlusion status, the equipment may fail to grasp the apple or even become damaged, severely compromising its safety. In this study, the graspability of apples refers to whether the apple's growing environment is suitable for robotic hands to safely harvest them (Yan et al., 2021).To enhance the selective grasping capabilities of apple-picking equipment, it is imperative for network models to discern occlusions produced by branches, leaves, and apples. Recent deep learning-based apple identification network model research has predominantly focused on the DasNet (Kang and Chen, 2019; Kang and Chen, 2020), YOLO (Dean et al., 2019; Wu et al., 2021; Wang et al., 2022), R-CNN (Dandan and Dongjian, 2019; Zhang et al., 2020), and Mask R-CNN (Jia et al., 2020; Chu et al., 2021) series of models. However, most studies have conducted apple identification through single-class recognition, overlooking the impacts of occlusions on harvesting. To mitigate risks during harvesting and boost operational efficiency, apple-picking equipment should be capable of precisely detecting various apple occlusion scenarios prior to harvesting, subsequently determining the graspability of apples; however, such detection methods are inherently more challenging, as they rely on subtle features based on the apple's local position (Minervini et al., 2016), making these fine-grained features elusive.

In conducting multi-occlusion apple detection, it is imperative to first compile a comprehensive data set representing various apple occlusion types, ensuring that these data accurately depict a myriad of occlusion scenarios. However, the compilation of such a data set is not only time-consuming and costly, but also susceptible to mislabeling of occlusion categories, which can compromise the accuracy of the final model. Furthermore, many network models, burdened by their substantial weights, exhibit sub-par real-time performance. In contrast, lightweight models, while boasting rapid computational speeds, often suffer from diminished recognition accuracy. To address these challenges, there is a pressing need to explore techniques centered on model-based label generation, parameter optimization, and architectural design. Wang et al. proposed an R-FCN network model based on ResNet-34 that adeptly identifies apples in the presence of overlap, leaf occlusion, and surface shadows, achieving recognition recall and accuracy rates of 85.7% and 95.1%, respectively (Dandan and Dongjian, 2019). Jia (Jia et al., 2020) introduced a lightweight modification into Mask R-CNN by integrating ResNet and DenseNet, and the model's detection precision and recall rates reached 97.31% and 95.70%, respectively. However, the model's detection speed still fell short of real-time detection requirements. Addressing this, Kuznetsova (Kuznetsova et al., 2020) proposed a pre-processing and post-processing approach based on YOLOv3, achieving a rapid detection speed of 19 ms per frame. Yan (Yan et al., 2022) designed the Bottleneck CSP-B module and an SE attention module based on YOLOv5m, making preliminary strides in detecting apple occlusion scenarios; nevertheless, instances of misidentification or outright non-recognition of apples were observed. Kang (Kang and Chen, 2020) introduced LedNet, which offers extensive data labeling capabilities, with the aim of enhancing fruit detection precision. While existing studies have made progress in terms of apple detection, there remains a pivotal need to address misidentification issues in multi-occlusion apple scenarios. This factor is crucial to ensure the precise determination of apple graspability and fulfill the demands of apple harvesting operations.

For this study, mature apples that remained unharvested in an orchard were selected as the subjects of investigation, and an occlusion-aware apple detection method based on an enhanced YOLOv5s model was proposed. Utilizing the results from this multi-occlusion apple detection method, the graspability of the apples was further assessed. To effectively reduce the need for annotations, minimize data preparation costs, and improve the performance of the YOLOv5s backbone, a training scheme based on self-supervised learning and knowledge transfer was employed. Additionally, the selective kernel module was integrated, enabling the refined YOLOv5s to more accurately identify apples with multiple occlusions, thereby enhancing the apple harvesting equipment's ability to determine apple graspability. This research offers a viable solution for precisely discerning apple graspability and has significant implications for improving the efficiency and safety of apple harvesting equipment.

2 Materials and methods

2.1 Apple orchard environment

Yantai City, located in the northeastern part of the Shandong Province, has geographical coordinates of 119°34′E to 121°57′E longitude and 36°16′N to 38°23′N latitude. Recognized as the birthplace of modern apple cultivation in China, Yantai is also among the country's primary apple-growing cities. The apple image data utilized in this study were collected in October 2021 from the Zoumaling Orchard in Biguo Town, Zhaoyuan County, Yantai City.

This orchard utilizes a modern spindle-shaped planting structure. The apple trees are spaced approximately 3.5 m apart, with a distance of about 1.5 m between individual trees and an average height of around 3.5 m, as depicted in Figure 1. During the apple maturation phase, the apples display a vibrant red hue, are densely clustered, and become relatively large, with an average weight of 319 g per apple. Prior to harvesting, the apple trees were sprayed with defoliants by orchard management personnel, which expedite the shedding of leaves. Consequently, by the time of apple maturation, fewer leaves remained on the apple trees, revealing a more pronounced presence of branches and resulting in a sparse canopy pattern. This distribution of branches and leaves not only provides the apples with increased sunlight exposure, but also presents a realistic scenario for research into the automated harvesting of apples.

2.2 Data collection and annotation

2.2.1 Data collection

An Intel D455 camera was employed to capture images at a range of 0.3 to 1.0 m from the apple trees. To ensure diversity in the captured images, the potential effects of varying weather and lighting conditions on the images were thoroughly considered. Images were taken during three distinct periods—morning, noon, and afternoon—and under both clear and cloudy weather conditions. These images were captured under various lighting modes, including front-lit, back-lit, and side-lit, as illustrated in Figure 2. In total, 5000 images with a resolution of 1280 × 720 pixels were collected, all of which were saved in the PNG format. After eliminating images with high redundancy, a final set of 2800 high-quality apple images were retained.



FIGURE 1
Planting scene of Zoumaling Orchard in Biguo Town, Zhaoyuan County.



FIGURE 2
Images under different lighting conditions. **(A)** Back-lit image. **(B)** front-lit image. **(C)** side-lit image. **(D)** low-light image.

2.2.2 Data annotation

Meticulous annotation of the images was conducted based on occlusion of the apple surfaces by branches, leaves, and other apples. All occlusion scenarios within the images were categorized into eight classes: No occlusion (N), Leaf occlusion (L), Apple occlusion (F), Branch occlusion (B), Leaf and Apple occlusion (LF), Leaf and Branch occlusion (BL), Branch and Apple occlusion (BF), and combined Leaf, Branch, and Apple occlusion (BLF). The LabelImg annotation software was employed (Zhuk et al., 2015), with labels generated in txt format. The results of the various occlusion annotations are depicted in Figure 3. From the perspective of actual apple harvesting operations, apples were classified into three categories based on their occlusion status: apples categorized as N or L were deemed to be Graspable Apples (GA), as the harvesting process remains unaffected when apples are either unobstructed or solely obstructed by leaves; apples categorized as F or LF were categorized as Temporarily Ungraspable Apples (TUGA) as, once the apples obstructing the surface are harvested, these apples can become subsequent grasping targets; and apples categorized as B, BL, BF, or BLF were classified as Ungraspable Apples (UGA),primarily due to branch obstructions, which could potentially damage the apples or the harvesting equipment if direct harvesting were attempted.

As detailed in Table 1, the data set contained a total of 36,803 annotated bounding boxes, among which ungraspable apples constituted the majority, accounting for 51.5% of total annotations. Graspable apples represented 42.0% of the total, while temporarily ungraspable apples made up 7.5%. The annotated results were divided into training, validation, and test sets at a ratio of 7:1:2, serving the purposes of network model training, optimization, and performance evaluation, respectively. During the training process, data augmentation techniques were employed, primarily involving the addition of noise to and forming mosaics of the images, as well as adjustments to contrast and brightness.

2.3 Construction of detection model

2.3.1 Methodology overview

To determine the graspability of the apples, we introduce a detection method for multi-occluded apples based on an enhanced YOLOv5s model. In particular, this method determines the graspability of the fruit based on the occlusion detection results. The technical framework of this method is depicted in Figure 4. Initially, data collection, annotation, and augmentation are conducted, establishing an eight-category occluded apple data set. The YOLOv5s model was employed for fully supervised data training, and the backbone of the post-training model was extracted to serve as the teacher backbone model for guided training. Given the data set size constraints, a joint training strategy combining knowledge transfer and self-supervised learning algorithms was devised, primarily aiming to construct a more robust student backbone model. To further optimize YOLOv5s, we integrated the SK module (Li et al., 2019). Ultimately, the student backbone model was utilized to initialize the enhanced YOLOv5s backbone. With the aid of the augmented training set, fully supervised fine-tuning was conducted in order to achieve optimal performance of the improved YOLOv5s model.

2.3.2 Improvement of YOLOv5s

In the context of the application requirements for embedded computing in apple harvesting equipment, the network model must possess the capability to rapidly and accurately identify apples (De-An et al., 2011). We chose YOLOv5s, which was designed specifically for embedded systems, for the baseline network model as it strikes a balance between detection speed and accuracy. YOLOv5s primarily consists of three components: the Backbone, Neck, and Head. To enhance the model's performance, modifications were made to both the backbone and Neck sections; see the overall architecture depicted in Figure 5.



FIGURE 3
Annotation results for various occluded apples in an image using the LabelImg. N stands for No occlusion, B stands for Branch occlusion, BL stands for Leaf and Branch occlusion, BLF stands for Leaf, Branch, and Apple occlusion.

TABLE 1 Statistics for three types of apple targets.

No. of Apples	No. of Graspable Apples		No. of Temporarily Ungraspable Apples		No. of Ungraspable Apples			
	N	L	F	LF	B	BL	BF	BLF
36803	9856	5597	1936	474	10,579	6405	1296	660

The backbone is responsible for transforming the input image into multi-layer feature maps suitable for object detection tasks. This component primarily consists of Conv modules, C3 modules, and Spatial Pyramid Pooling Fast (SPPF) modules. The Conv module encompasses convolution (Conv2d), Batch Normalization, and the SiLU activation function. The C3 module draws inspiration from DarkNet53 in YOLOv3 (Redmon and Farhadi, 2018), combined with the design philosophy of CSPNet (Wang et al., 2020), and includes three Conv and multiple Bottleneck modules. The Bottleneck module employs the residual structure from ResNet (He et al., 2015), primarily in two variations: the first path uses a 1 × 1 convolution to halve the channel number of the feature map before a 3 × 3 convolution extracts features, ensuring consistent input and output channel numbers; while the second path uses a direct shortcut for residual connection, thus achieving feature fusion. The other variation omits the feature fusion step when no shortcut is applied. The C3 module aims to enhance the network's depth and receptive field, thereby improving its feature extraction capabilities. Inspired by SPPNet (He et al., 2014), the SPPF module replaces a large pooling kernel with

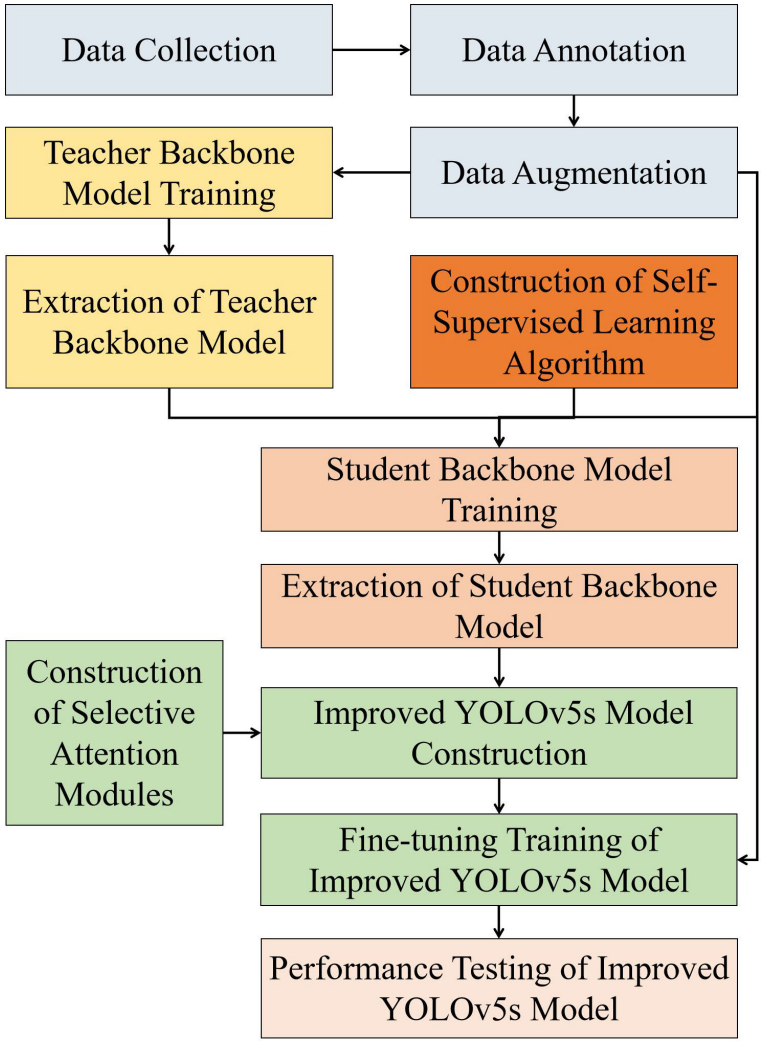


FIGURE 4
Technological framework of the proposed approach. In the technical framework, the same color represents the same experimental stage, grey boxes represent raw images pre-processing, yellow boxes represent the construction of teacher backbone model, dark yellow boxes represent the construction of self-supervised learning algorithm, light red boxes represent the construction of student backbone model, green boxes represent the improvement process of YOLOv5s model, and light pink represents the performance outputs of the improved YOLOv5s model.

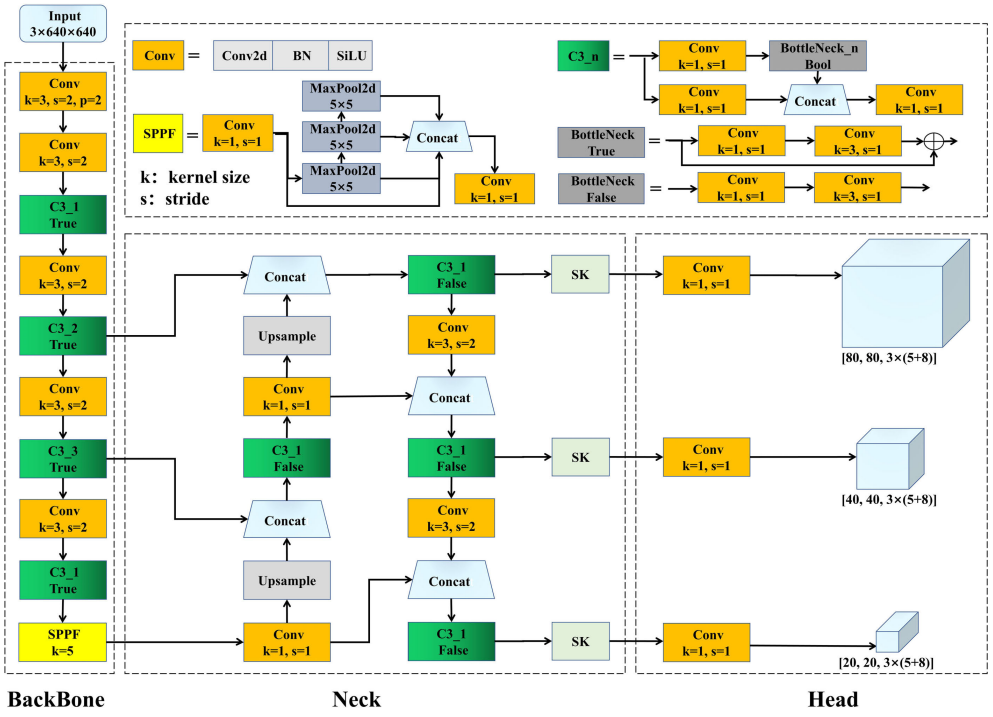


FIGURE 5
Improved YOLOv5s architecture.

multiple smaller ones, thereby enhancing the execution speed and feature expressiveness. In Sections 2.3.3 and 2.3.4, we describe a guided pre-training strategy based on self-supervised learning and knowledge transfer, developed with the aim of training a backbone capable of fine-grained feature extraction for multi-obstructed apple detection.

The Neck module is tasked with integrating feature maps at different levels, producing feature maps with multi-scale information, and forwarding those maps to the Head section. This component is composed of Conv modules, Upsample, Concat, and a C3 module without a shortcut. Moreover, the design of the Neck incorporates structures from FPN (Lin et al., 2016) and PAN (Liu et al., 2018), employing both top-down and bottom-up feature extraction methods, thus facilitating the fusion of shallow graphic features and deep semantic features of the network. In Section 2.3.5, we detail how the SK module is introduced to enhance the Neck's focal representation capabilities for target region features. The Head primarily conducts multi-scale object detection on the feature maps integrated by the Neck. This module's design aims to expand the channel numbers of the three differently sized feature maps in the Neck. The expanded channel number calculation method is presented in Equation 1.

$$CM = (OAC+5)\times NA \quad \text{(Eq. 1)}$$

where *OAC* represents the number of occluded apple categories, and the 5 represents five parameters: the bounding box center's x- and y-coordinates, width, height, and confidence score). *NA* represents the number of anchors per detection layer. In this study, *OAC* is 8,and *NA* is 3.

2.3.3 Pre-training of the student backbone model based on BYOL

In recent years, Self-Supervised Learning (SSL) has gained significant attention in the realm of image processing, offering a novel approach to model training that does not rely on manually annotated data (Jing and Tian, 2019). By autonomously extracting labels from a vast amount of unlabeled data, this training method substantially reduces the dependency on annotated data, leading to significant savings in both time and cost. Early SSL methods typically relied on designing specific predictive tasks, such as estimating image rotation angles or color arrangements, thereby encouraging the model to discern meaningful image features (Doersch et al., 2015). More recently, researchers have identified SSL techniques that draw similar image features closer while pushing dissimilar ones apart, such as Momentum Contrast (He et al., 2019), BYOL (Grill et al., 2020), and SimCLR (Chen et al., 2020b). Notably, BYOL stands apart from other contrastive learning methods that rely on negative samples; instead, BYOL learns image representations from two distinct image views derived from a target network and an online network, respectively. This strategy not only streamlines the learning process, but also achieves efficient feature representation without the use of any negative samples. Given the potential of SSL in deep learning, this study leverages BYOL to enhance the performance of the YOLOv5s backbone.

The initial step involved setting up the target network model and the online network model. The backbone of YOLOv5s was first selected as the online encoder. Subsequently, the weights of the online network model were cloned to produce the target encoder, the calculation method is presented in Equation 2.

$$F(x) = W_{v5s_{B_i}}(x) \quad \text{(Eq.2)}$$

where $F(x)$ is the feature tensor extracted from the input image and $W_{v5s_{B_1}}$, $i \in \{\text{online}, \text{target}\}$ represents the type of encoder.

To enhance the encoder's generalization capability, we devised a data augmentation strategy considering the characteristics of agricultural images. Initially, random cropping and horizontal flipping of the images were employed, supplemented with color adjustments and brightness/contrast modifications, succinctly termed Color Adjustment (CRAJ). The calculation method for generating augmented images from the original images is presented in Equation 3.

$$x_k = \textbf{RandomCrop}(\textbf{RandomHorizontalFlip}(\textbf{CRAJ}(x))) \quad \text{(Eq. 3)}$$

where x_k, $k \in \{1, 2\}$ represents the augmented image.

Subsequently, construction of the projection head and predictor was carried out. Within the online network model, both the projection head and predictor are composed of a multi-layer perceptron (MLP). The prediction calculation method for the online network model is presented in Equation 4.

$$z_{online_i} = W_2\sigma(W_1F(x_i)) \quad \text{(Eq. 4)}$$

In the target network model, the projection head consists of a single MLP and does not include a predictor. The projection calculation method for the online network model is presented in Equation 5.

$$z_{target_i} = \sigma(W_1^{'}F(x_i)) \quad \text{(Eq. 5)}$$

where W_1 and W_2 represent the weights of the projection head and predictor in the online network model, respectively; $W_1^{'}$ denotes the weights of the projection head in the target network model; and σ is the ReLU activation function.

Subsequently, construction of the BYOL loss function was undertaken. The loss calculation method is presented in Equation 6.

$$L = \sum_{i=0}^{1} 2 - 2 \times \frac{\langle z_{online_i}, z_{target_i} \rangle}{||z_{online_i}||_2 \times ||z_{target_i}||_2} \quad \text{(Eq. 6)}$$

where the inner product of vectors is denoted by $\langle \cdot, \cdot \rangle$, z_{online_1} represents the output processed by the online encoder when processing x_1, z_{target_1} signifies the output processed by the target encoder when processing x_2, and L is the result of the loss computation.

Subsequently, an overarching training optimization strategy for the network model was devised. Utilizing standard backpropagation and the Adam optimizer, the gradient of the loss function L with respect to the weights of the online encoder was computed, allowing for updating of the online weights. Concurrently, to stabilize the self-supervised training process, we employed an exponential moving average strategy to update the weights of the target encoder, which was calculated in Equation 7.

$$W_{target} = \beta \times W_{target} + (1 - \beta) \times [W_{v5s_{B_{online}}}, W_1] \quad \text{(Eq. 7)}$$

where W_{online} represents the combination of $[W_{v5s_{B_online}}, W_1, W_2]$ and W_{target} represents $[W_{v5s_{B_target}}, W_1^{'}]$. For β, a value of 0.90 was set to update the weights of the target encoder.

The self-supervised training process of the YOLOv5s backbone based on BYOL is illustrated in Figure 6. We utilized 5000 images to deeply pre-train the backbone of YOLOv5s in a self-supervised manner. The BYOL method efficiently learns features while relying solely on the loss of the online network. Upon completion of the pre-training step, the acquired weights—encapsulating vital visual feature information about apple trees—were stored within the YOLOv5s backbone. These weights could then be applied to downstream object detection tasks. In the subsequent phase, we fine-tuned the YOLOv5s backbone using the test data set, resulting in the final BYOL-improved YOLOv5s.

2.3.4 Pre-training of the student backbone model based on knowledge transfer

In the realm of deep learning, the process of knowledge transfer primarily refers to utilizing a model trained on one task as a starting point for training on another task (Passalis and Tefas, 2018). The foundational concept is to transfer the knowledge from the teacher

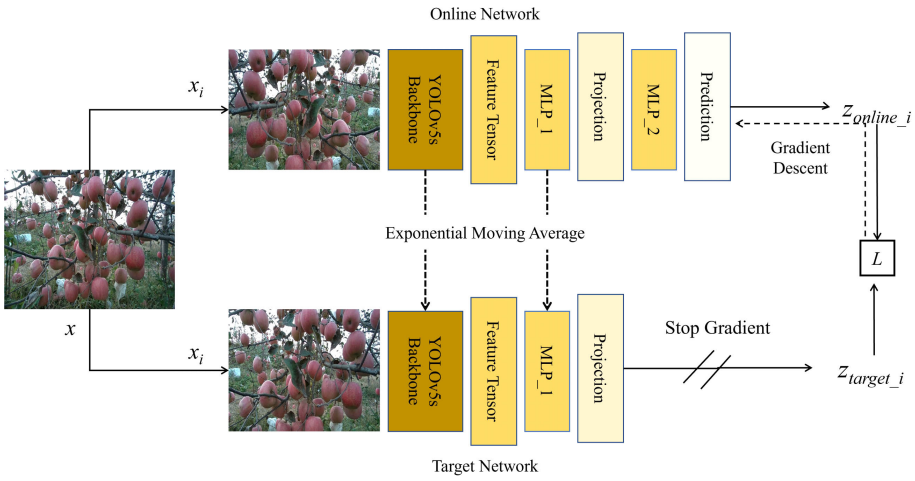


FIGURE 6
Self-supervised training framework of the YOLOv5s backbone based on BYOL.

backbone model to the student backbone model, with the hope that the student backbone model may approach or even surpass the performance of the teacher model (Chen et al., 2020a). We focused on enhancing the feature extraction capability of the YOLOv5s backbone, exploring how to further amplify the backbone's feature extraction ability through knowledge transfer methods by leveraging the pre-trained YOLOv5s.

In orchard environments, apples are frequently occluded by leaves, branches, and other apples. It is crucial to consider how to enable the model's backbone to learn about the shapes, sizes, and textures of such obstructions. We employed a backbone distillation approach, utilizing intermediate feature activation layers to enable the student backbone model to learn from the teacher model. These intermediate feature activation layers can accurately represent the shapes and textures of leaves, branches, and apples, thereby offering improved detection in scenarios with multiple obstructions. Given the inherently commendable performance of YOLOv5s and based on preliminary experimental results, we decided to use the backbone of YOLOv5s trained with supervised data as the teacher backbone model. We chose the untrained YOLOv5s backbone as the student backbone model. This design strategy aims to achieve self-guidance and transfer learning for YOLOv5s, thus promoting enhanced backbone performance. Throughout this process, multiple intermediate feature activation layers in the teacher backbone model are utilized. For each intermediate layer, denoted by l, we compute the corresponding feature activation results F_T^l. To enable the student backbone model, denoted by *S*, to learn the information from these intermediate feature activation layers, we designed a feature matching loss, which was calculated in Equation 8.

$$L_{feature}^{l} = \frac{1}{N_l} \Sigma_{i=1}^{N_l} ||F_T^{l,i} - F_S^{l,i}||_2^2 \quad \text{(Eq. 8)}$$

where N_1 represents the number of feature channels in layer l, while $F_T^{l,i}$ and $F_S^{l,i}$ denote the feature tensors of the student and teacher backbone models at layer l in channel i, respectively. In our practical experiments, we selected the fifth feature activation layer as preliminary experiments indicated that the model's backbone performance reached its peak when l is 5.

For the knowledge transfer process, we employed several techniques to ensure training stability and expedite convergence, including learning rate decay, early stopping strategies, and data augmentation. We configured the optimizer as Adam with an initial learning rate of 0.001 and weight decay of 0.0005. The learning rate was scheduled to decrease by 2% every 10 epochs.

To fully leverage the limited training data set and quantity of unlabeled data, we further explored combinations of self-supervised learning methods in addition to knowledge transfer, with the aim of enhancing the performance of the YOLOv5s backbone for improved results in object detection tasks. The specific architecture is illustrated in Figure 7, and the overall loss calculation method derived from the combination of these two approaches is presented in Equation 9.

$$L_{all} = \alpha L + \beta L_{feature}^{l} \quad \text{(Eq. 9)}$$

where L represents the contrastive loss generated through self-supervised learning and $L_{feature}^{l}$ denotes the loss arising from knowledge transfer. Additionally, α and β are hyperparameters, with βcontrolling the strength of knowledge transfer and α regulating the impact of self-supervision on model training. In

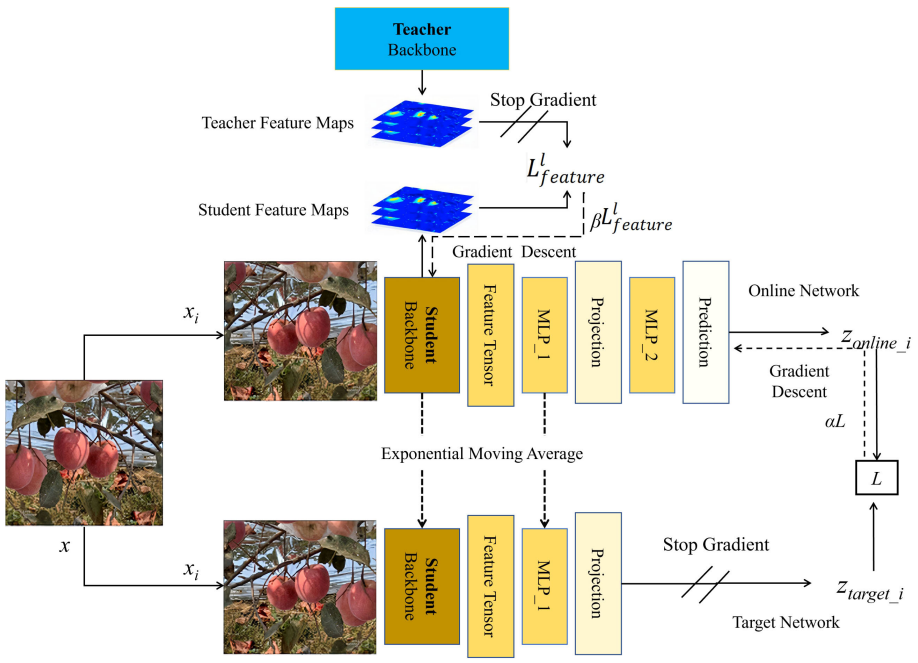


FIGURE 7
Guided training of the student backbone model through fusion of the teacher backbone model and BYOL.

our experiments, these hyperparameters were set as 0.1 and 0.9, respectively.

2.3.5 Selective Kernel module

Attention mechanisms have recently become indispensable in the design of deep learning models, especially when addressing intricate image problems (Zhang et al., 2018). The Squeeze-and-Excitation (SE) attention mechanism optimizes feature weights at the channel level (Hu et al., 2017), yet its responsiveness to specific spatial contexts remains limited. In contrast, CBAM aims to integrate both spatial and channel attention (Woo et al., 2018), but its performance still requires improvement when handling multi-scale and intricate occlusion scenarios. Given the demand for detecting apples with various types of occlusion—especially considering the sensitivity to diverse occlusion patterns and target size variations—a strategy that can dynamically adjust the receptive field has become crucial. Considering this need, the SK module has a unique advantage (Li et al., 2019): it endows each spatial location with the ability to dynamically select convolutional kernels, offering profound contextual understanding of different occlusion types, thereby achieving more refined and adaptive feature extraction.

The SK module is illustrated in Figure 8. In this model, the input feature tensor X first undergoes full convolution operations with two distinct kernel sizes. For this study, 3 × 3 and 5 × 5 convolutional kernels were employed, with dilation parameters set to 1 and 2, respectively, yielding two feature maps (denoted A1 and A2) matching the dimensions of the original feature map. Subsequently, the corresponding elements of A1 and A2 are summed to produce an overall feature map, B, which retains the dimensions of the original input feature map. B is then subjected to a global average pooling operation, resulting in the feature map S. A fully connected layer (FC) is then utilized to extract channel attention information, producing a further feature map Z, with dimensions of d × 1 × 1. Then, the feature map Z is separately processed by two softmax functions, a and b, to obtain the channel attention information. The channel attention information is then multiplied element-wise with the feature maps A1 and A2, outputting two channel attention feature maps, denoted C1 and C2. To further emphasize key features and suppress extraneous information, C1 and C2 are fused by adding their corresponding positions, yielding a final feature map Y, with dimensions H × W × C.

2.4 Model training and performance metrics

2.4.1 Training equipment

We conducted all experiments on a deep learning server equipped with a 64-core Intel Xeon(R) Gold 6226R mailto:v4@2.90v4@2.90 Hz CPU, 251.6 GB of RAM, and a 1.9 TB solid-state drive, along with two 16 GB NVIDIA Tesla V100 GPUs. On the software side, the server ran the Ubuntu 20.04 operating system with NVIDIA driver version 495.46, PyTorch 1.10, CUDA 11.5, and cuDNN 8.2.4.

2.4.2 Training details

We adopted the following training strategy. First, we performed teacher backbone model training based on YOLOv5s and a labeled data set of multi-occluded apples. This training ensured that the model could better understand and handle apple occlusion scenarios. Once the model converged, we saved the optimal weights and extracted the backbone weights for further use in training the student backbone model. Next, we extracted the backbone portion from the improved YOLOv5s model and integrated it into both the online and target backbones of BYOL. During this stage of training, while the teacher backbone was frozen, we iteratively updated the student backbone model using the self-supervised learning loss and knowledge transfer loss. After training on 5000 orchard images, we obtained an optimal student backbone model. Next, we swapped the optimal student backbone model with the improved YOLOv5s backbone and proceeded to fine-tune the model. Notably, the entire training process was divided into two stages: The backbone freezing stage and the backbone unfreezing stage. In the initial 100 iterations of the backbone freezing stage, the backbone parameters remained unchanged and we only fine-tuned the neck and head networks. The initial learning rate for this stage was set to 0.002, and we used the Adam optimizer with a momentum parameter of 0.85. If the loss did not decrease between two iterations, the learning rate was halved. After 100 iterations, we entered the backbone unfreezing stage, where all network parameters were updated. The initial learning rate was set to 0.001, and the learning rate update strategy was the same as in the previous stage. Ultimately, when the network model converged, we obtained the YOLOv5s backbone optimized for multi-occluded apples.

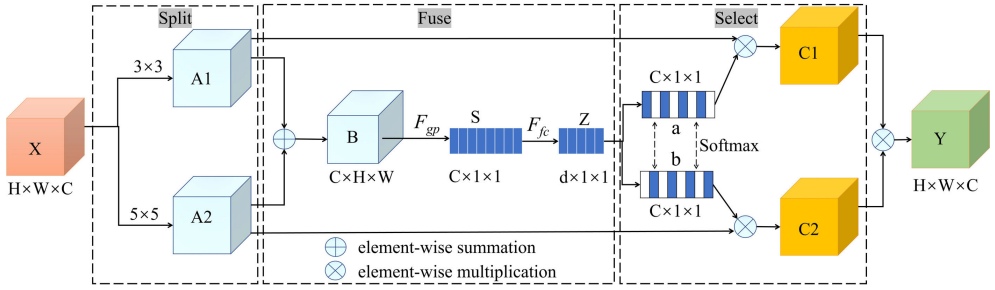


FIGURE 8
Selective kernel module.

2.4.3 Performance metrics

We evaluated the performance of the trained network model using four metrics: Precision (P), Recall (R), Average Precision (AP), and Mean Average Precision (mAP). The specific calculation methods for these metrics are presented in Equations 10–13.

$$\mathbf{P} = \mathbf{TP}/(\mathbf{TP} + \mathbf{FP}) \quad \text{(Eq. 10)}$$

$$\mathbf{R} = \mathbf{TP}/(\mathbf{TP} + \mathbf{FN}) \quad \text{(Eq. 11)}$$

$$\mathbf{AP} = \int_0^1 \mathbf{P_n}(\mathbf{R_n})\mathbf{d} \quad \mathbf{R_n} \quad \text{(Eq. 12)}$$

$$\mathbf{mAP(n)} = 0.125 \times \sum_{\mathbf{n}=1}^{8} \mathbf{AP(n)} \quad \text{(Eq. 13)}$$

where P represents the proportion of correctly predicted boxes among all predicted boxes and R represents the proportion of correctly predicted boxes among all labeled boxes. To assess the model's performance in different categories, we used AP(n), which denotes the average precision for the nth class of multi-occluded apples, and mAP, which represents the average precision across the eight types of occluded apples. Here, TP stands for the number of predicted boxes correctly matched with annotated boxes, FP represents the number of incorrectly predicted boxes, and FN represents the number of labeled boxes that are not predicted.

3 Results

3.1 Detection results and analysis

To precisely assess the performance of the improved YOLOv5s model in terms of apple graspability detection, validation was conducted on a test set comprising 560 images. For the evaluation process, three critical metrics were defined: AP_{GA}, AP_{TUGA}, and AP_{UGA}, representing the average precision of detection for graspable, temporarily ungraspable, and ungraspable apples, respectively. Table 2 presents a performance comparison between the improved YOLOv5s and the original YOLOv5s. Notably, when compared to the original network, the improved YOLOv5s exhibited increases of 2.08%, 3.03%, and 3.65% in the mAP, AP_{GA}, and AP_{UGA} metrics, respectively, while showing a slight decline of 0.45% in the AP_{TUGA} metric. This result suggests that the improved YOLOv5s achieved enhanced detection accuracy for the GA and UGA categories, with only a minor decrease in performance for the TUGA category. Figure 9 provides a comparative visualization of detection outcomes for both models, in which instances of misidentification by YOLOv5s are indicated by yellow circles. Ultimately, the improved YOLOv5s model achieved accurate discernment.

In image detection tasks conducted under various lighting conditions, the original YOLOv5s exhibited several misclassification errors. Specifically, under side lighting (Figure 9A), the L was misclassified as N and BL as L. These errors were primarily attributed to the subdued brightness of the apple leaves in shadowed areas towards the upper right, leading to indistinct leaf features and, consequently, misidentification of N. Additionally, substantial leaf occlusion diminished the salient characteristics of prominent branches, causing misidentification of BL. In the back-lit case (Figure 9B), low light occluded subtle features of L and BL, resulting in their misclassification as N and B, respectively. In low-lighting situations (Figure 9C), confusion between BL and L was observed. Under front-lit condition 1 (Figure 9D), similarities between background and target features resulted in detection failures. This inadequacy was a consequence of YOLOv5s losing certain features during the convolution and pooling processes. In addition, under front-lit condition 2 (Figure 9E), the shadow formed by the leaves on the apples led to B being mistaken for BL. The improved YOLOv5s model ameliorated the detection outcomes for all aforementioned tasks, yielding superior performance in terms of capturing fine-grained features. This result highlights the model's enhanced ability to discern between similar categories. Overall, the improved YOLOv5s consistently excelled across diverse lighting conditions, fulfilling the perceptual needs of apple harvesting equipment more effectively and significantly mitigating the risk of misjudging apple graspability.

3.2 Ablation study

To validate the positive impact of each proposed improvement on the performance of the YOLOv5s model, we conducted ablation experiments, the results of which are presented in Table 3. During the training of YOLOv5s, we employed an online network self-supervised learning strategy based on BYOL. The purpose of this strategy was to enhance the feature extraction capabilities of the online network model with respect to the images. By introducing a teacher backbone model to train the student backbone model, we aimed to more accurately map the teacher feature space to the student feature space. Additionally, we integrated the SK module with the goal of optimizing the detection capabilities for occluded targets at different scales (including distance and size), thereby reducing instances of missed detections and errors.

Following training on the YOLOv5s backbone under the BYOL self-supervised learning strategy, the backbone was integrated into

TABLE 2 Comparative detection performance results between YOLOv5s and improved YOLOv5s.

Network Model	mAP (%)	AP_{GA} (%)	AP_{TUGA} (%)	AP_{UGA} (%)	FLOPs(G)	FPS
YOLOv5s	91.29	91.55	92.20	90.13	16.4	120
Improved YOLOv5s	94.54	94.78	93.86	94.98	19.2	101

10.3389/fpls.2023.1323453

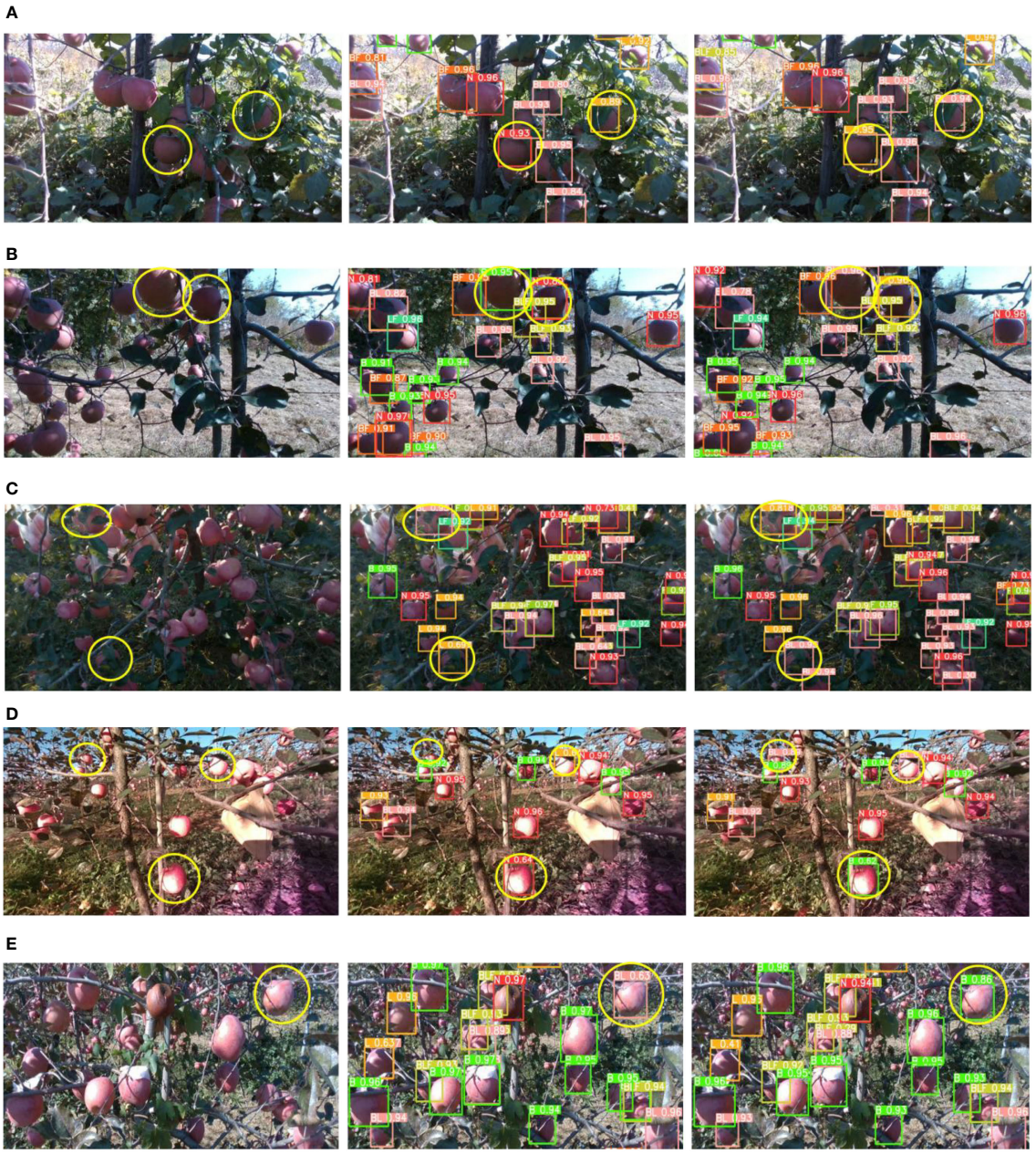


FIGURE 9
Recognition results before and after improvement of YOLOv5s. **(A)** Comparison of detection results under side-lit conditions. **(B)** Comparison of detection results under back-lit conditions. **(C)** Comparison of detection results under low-light conditions; **(D)** Comparison of detection results under front-lit condition 1. **(E)** Comparison of detection results under front-lit condition 2.

TABLE 3 Results of the ablation experiments.

YOLOv5s	BYOL	KT	SK	mAP (%)	AP_{GA} (%)	AP_{TUGA} (%)	AP_{UGA} (%)	FLOPs(G)	FPS
√	×	×	×	91.29	91.55	92.20	90.13	16.4	120
√	√	×	×	93.04	93.30	93.05	92.77	16.4	120
√	√	√	×	93.77	93.82	92.60	93.90	16.4	120
√	×	×	√	93.71	94.08	93.32	93.73	19.2	101
√	√	√	√	94.54	94.78	93.86	94.98	19.2	101

*"×" indicates the module is not used, while "√" indicates the module has been used.

the YOLOv5s model. Subsequently, YOLOv5s was fine-tuned using the test set. The end result was a YOLOv5s model reinforced through the BYOL self-supervised approach. The performance improvements in mAP, AP_{GA}, AP_{TUGA}, and AP_{UGA} were 1.75%, 1.75%, 0.85%, and 2.64%, respectively, indicating the enhanced ability of the backbone to extract the features of apples. The Reference image (Figure 10A) was selected to provide a visual comparative analysis of the backbone feature maps before and after BYOL training with YOLOv5s, as shown in Figures 10B, C. Additionally, specific attention was paid to the feature maps of the fifth layer. In post-training with the BYOL strategy, the convolutional layers indicated improved detection of the subtle contours and textures of branches and apples. The feature maps from this layer—in terms of both quality and extent—noticeably surpassed those from the original YOLOv5s model, providing solid evidence for the efficacy of the BYOL strategy in enhancing the fine-grained feature extraction capabilities of the YOLOv5s backbone.

We further evaluated the improvement of the backbone's performance through a guided training strategy integrating BYOL approaches with knowledge transfer. Across the various evaluation metrics, performance increases of 0.73%, 0.52%, and 1.13% were observed in mAP, AP_{GA}, and AP_{UGA}, respectively. However, a decline of 0.45% was observed in AP_{TUGA}. Both the computational complexity and inference speed of the network remained unaffected. We carefully examined the disparities between the multi-level feature maps of the hidden layers in the teacher backbone model and the student backbone model in order to compute the regularization loss. This loss was successfully integrated with the self-supervised learning loss to iteratively update the student backbone model. Notably, while the teacher backbone model was trained based on a self-supervised learning approach using YOLOv5s, the teacher backbone model still offers beneficial guidance on the hidden features of the student backbone model. This guidance is possibly due to the supervisory signals generated by the teacher backbone model, which provide a clear learning direction for the student backbone model at the same scale. This positively influenced the convergence process of the student backbone model.

Upon integrating only the SK module into YOLOv5s, further performance enhancement was realized. Specifically, the improved model experienced increases of 2.42%, 2.53%, 1.12%, and 3.6% in mAP, AP_{GA}, AP_{TUGA}, and AP_{UGA}, respectively. Although inclusion of the SK module led to a computational complexity increase of 2.8 GFLOPs, the computational speed still adequately met real-time processing requirements. To elucidate the reasons for this performance enhancement, the output feature maps of the detection network model across three sizes were mapped to pseudocolor images in the original size and overlaid onto the original images, allowing for visualization of the output features. These visualized feature images were generated in three resolutions: 80 × 80, 40 × 40, and 20 × 20, as depicted in Figure 11. In Figure 11A, the 20 × 20 resolution feature map primarily highlights the higher-order features of the apples while simultaneously smoothing out background details. This representation aids in more accurately distinguishing between background and target apples during detection. With the introduction of the SK module, one can directly observe a pronounced enhancement in the model's apple perception capabilities, thus reducing omissions in apple detection. The particular feature map shown in Figure 11B primarily accentuates the background. Before the introduction of the SK module, the extracted landmarks were somewhat coarse; however, with the SK module, there was a significant expansion in the model's feature perceptive range. In Figure 11C, the 80 × 80 feature map reveals more profound background perception and heightened differentiation between all apple features, thereby validating that integration of the SK module justifiably and effectively elevated the performance of the improved model.

4 Discussion

In the complex environments characteristic of orchards, harvesting equipment needs to not only precisely detect the locations of fruits but also intelligently determine the types of obstruction preventing access to the fruit. It is crucial to discern the fruit's graspability to help such equipment avoid rigid obstructions and achieve damage-free harvesting of apples. As shown in Table 4, the methodology proposed in this study achieved scores of 94.54%, 94.78%, 93.86%, and 94.98% in the mAP, AP_{GA}, AP_{TUGA}, and AP_{UGA} metrics, respectively. These

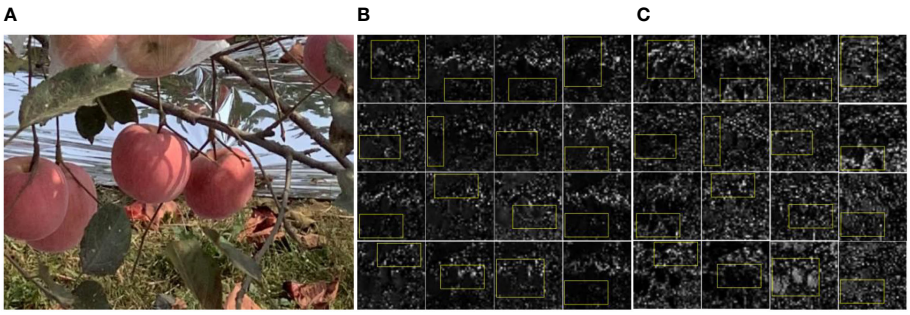


FIGURE 10
Backbone Feature maps of YOLOv5s Before and After Improvement. **(A)** Reference Image. **(B)** Backbone feature maps without BYOL. **(C)** Backbone feature maps with BYOL.

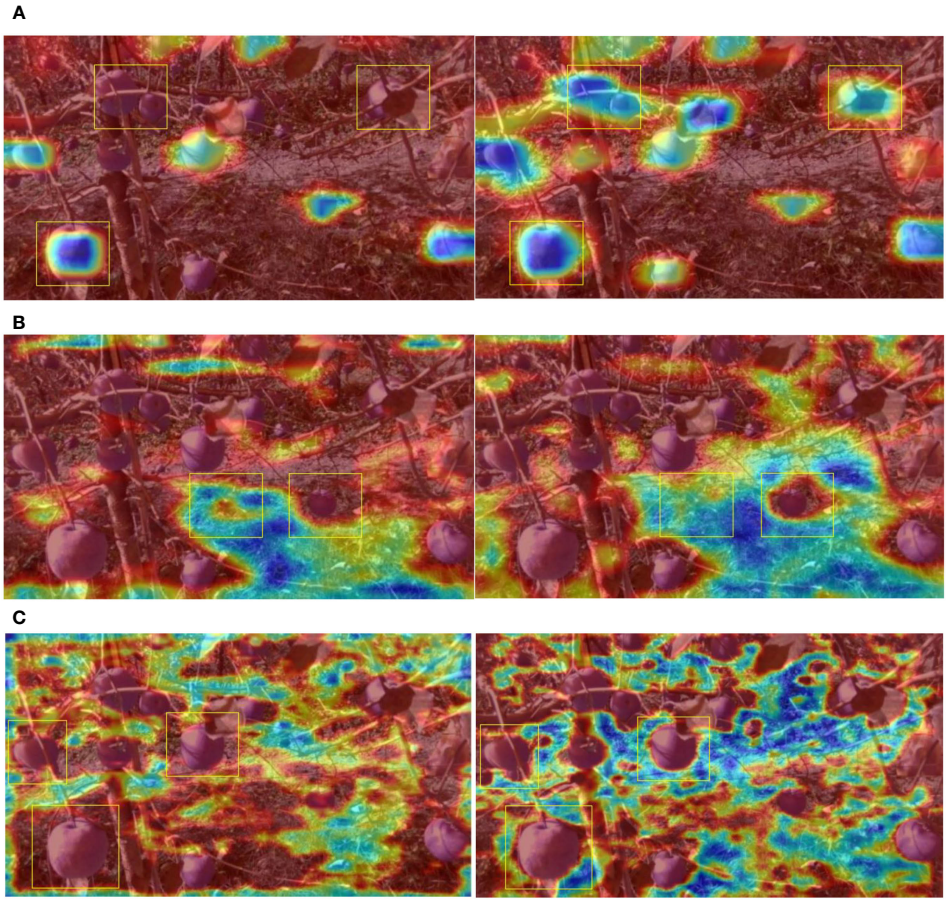


FIGURE 11
Visualization of features maps before and after addition of the SK module. **(A)** Comparison of the visualization results for 20 × 20 feature maps with and without the SK module. **(B)** Comparison of visualization results for 40 × 40 feature maps with and without the SK module. **(C)** Comparison of the visualization results for 80 × 80 feature maps with and without the SK module.

results demonstrate that the proposed approach provides robust support for both actual harvesting needs and future research in related domains. To specifically discuss the advantages and underlying reasons for the improved performance of YOLOv5s over contemporary similar models, we conducted comparative experiments with the improved YOLOv5s and other popular object detection network models. Additionally, YOLOv5x was incorporated to observe the peak performance of the YOLOv5 series, serving as a benchmark for optimal performance. Lightweight network models primarily include YOLOX (Ge et al., 2021), YOLOv4-s (Bochkovskiy et al., 2020), and YOLOv7s (Wang et al., 2022). To ensure fairness and consistency in testing, a uniform data set was employed to deeply train and assess the performance of multiple network models. Table 4 provides the detection results, detailing not only the Floating Point Operations (FLOPs) of each model, but also the processing speed advantages and disadvantages of each model, represented in terms of Frames Per Second (FPS).

The results indicated that, due to its larger weights, the YOLOv5x model distinguished itself from the many evaluated models, especially in the four evaluation metrics mAP, AP_{GA}, AP_{TUGA}, and AP_{UGA}.

TABLE 4 Test results for different network models.

Network Model	mAP (%)	AP_{GA} (%)	AP_{TUGA} (%)	AP_{UGA} (%)	FLOPs(G)	FPS
YOLOv5x	95.00	95.36	94.60	95.04	205.5	59
YOLOv5s	91.29	91.55	92.20	90.13	16.4	120
YOLOv4-s	88.40	87.64	89.65	87.91	15.4	164
YOLOv7s	72.20	70.56	71.94	74.10	13.2	113
YOLOX-s	90.40	91.56	89.85	89.79	26.8	73
improved YOLOv5s	94.54	94.78	93.86	94.98	19.2	101

However, the differences between the improved YOLOv5s and YOLOv5x on these key indicators were relatively minimal (0.46%, 0.58%, 0.74%, and 0.06%, respectively). These findings provide a critical insight: in the domain of graspable apple detection, the improvements introduced in this paper enabled YOLOv5s to achieve performance nearly on par with that of the YOLOv5x model. In terms of detection speed, the improved YOLOv5s significantly surpassed YOLOv5x, offering a distinct advantage for deployment in actual embedded devices. When juxtaposed with other prevalent lightweight network models, the improved YOLOv5s exhibited conspicuous performance enhancements in the mAP metric over YOLOv4-s, YOLOX-s, YOLOv7s, and YOLOv5s (by 3.25%, 4.5%, 22.32%, and 6.14%, respectively). Although YOLOv4-s presented outstanding inference speeds—reaching up to 164 FPS—its performance in various AP metrics was less than ideal, with results akin to those of YOLOv7s. This result offers a salient lesson: it is imprudent to solely prioritize speed at the expense of accuracy. Conversely, while YOLOv7s possesses a straightforward model structure, its overall performance was relatively underwhelming, suggesting that this model may not be appropriate for high-precision granular detection tasks. The integration of the SK module, despite enhancing the model's computational demands, impacted its inference speed. However, the authors in (Suo et al., 2021) determined that the picking time for a singular apple is approximately 2780 milliseconds. This result suggests that, even with a slight decrease in detection speed, our model remains adept at meeting the real-time requirements of agricultural apple harvesting equipment.

In summary, the advantages of the method proposed in this study were apparent in three primary areas. Initially, the proposed training approach and enhancement strategies for the network model enabled precise identification of various apple occlusion types within images. This method not only allows for determination of the graspability of apple targets, thus saving data annotation costs, but also achieved the stipulated design objectives. Furthermore, the detection performance of the improved YOLOv5s was markedly superior when compared to similar algorithms, making it well-suited to the damage-free harvesting needs of apple-picking equipment. The improved YOLOv5s retained its lightweight attributes, suggesting its significant potential for deployment in embedded hardware systems and laying a foundation for broader applications. On the other hand, certain limitations to our approach were identified. For example, the training process for our network model is intricate. Compared to the training protocol of the original YOLOv5s, this backbone requires multiple training iterations, prolonging the training duration. Additionally, the methodological data sets employed in this research largely prioritized red apples, leading to potential compromises in detection efficacy for non-red varieties, such as yellow and green apples. Finally, our detection strategy does not account for the potential impacts of fruit pose variations on apple graspability.

5 Conclusions

In response to the demand for more efficient and safe apple harvesting equipment, we proposed an improved YOLOv5s-based multi-occluded apple detection network model, which can efficiently identify graspable, temporarily ungraspable, and ungraspable apples. By incorporating knowledge transfer and BYOL strategies, along with integration of the SK module, the improved YOLOv5s model achieved optimized detection performance. Experimental data confirmed that this model offers strong performance in detecting multi-occluded apples, obtaining AP_{GA}, AP_{TUGA}, and AP_{UGA} scores of 94.78%, 93.86%, and 94.98%, respectively; furthermore, compared to the original YOLOv5s, our model presented improvements of 3.23%, 1.66%, and 4.85%, respectively, for these metrics. Although our proposed SK module slightly increased the computational complexity, it significantly enhanced detection accuracy and discrimination while still meeting the speed requirements for practical harvesting. When compared to state-of-the-art popular lightweight network models, the improved YOLOv5s model presented clear advantages in detection accuracy and approached the performance level of larger network models such as YOLOv5x. For future research, we intend to focus on integrating fruit occlusion types with fruit poses in the detection model, in order to further enhance the model's practical value.

Data availability statement

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

Author contributions

HBL: Writing – original draft, Writing – review & editing, Conceptualization, Formal Analysis, Methodology, Supervision, Validation. PY: Formal Analysis, Methodology, Writing – review & editing. HYL: Formal Analysis, Writing – review & editing. XL: Investigation, Writing – review & editing. JQ: Project administration, Resources, Writing – original draft. QY: Formal Analysis, Investigation, Supervision, Conceptualization, Writing – review & editing. CG: Resources, Visualization, Writing – original draft. YS: Conceptualization, Funding acquisition, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the following projects: research and development of key technologies for agricultural situation parameter acquisition and integrated application of sensing equipment-2022LQ02004 and National Natural Science Foundation of China-Regional Innovation and Development Joint Fund (Jilin) U19A2061.

Acknowledgments

We would like to express our gratitude to the Shandong Academy of Agricultural Sciences for providing us with the data collection site, which greatly facilitated the efficient gathering of experimental data. We also extend our thanks to all the open-source projects and data sets used in this work, as they supported our specific experimental comparisons.

Conflict of interest

Author PY was employed by Suzhou Zhongnong Digital Intelligence 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

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). YOLOv4: optimal speed and accuracy of object detection. *ArXiv*. doi: 10.48550/arXiv.2004.10934

Chen, D., Mei, J.-P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., et al. (2020a). "Cross-layer distillation with semantic calibration," in *AAAI Conference on Artificial Intelligence*.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. (2020b). A simple framework for contrastive learning of visual representations. *ArXiv*, 1597–1607. doi: 10.48550/arXiv.2002.05709

Chu, P., Li, Z., Lammers, K., Lu, R., and Liu, X. (2021). Deep learning-based apple detection using a suppression mask R-CNN. *Pattern Recognition Lett.* 147, 206–211. doi: 10.1016/j.patrec.2021.04.022

Dandan, -.W., and Dongjian, -.H. (2019). Recognition of apple targets before fruits thinning by robot based on R-FCN deep convolution neural network. *Trans. Chin. Soc. Agric. Eng. (Transactions CSAE)* 35, 156. doi: 10.11975/j.issn.1002-6819.2019.03.020

De-An, Z., Jidong, L., Wei, J., Ying, Z., and Yu, C. (2011). Design and control of an apple harvesting robot. *Biosyst. Eng.* 110, 112–122. doi: 10.1016/j.biosystemseng.2011.07.005

Dean, -.Z., Rendi, -.W., Xiaoyang, -.L., and Yuyan, -.Z. (2019). - Apple positioning based on YOLO deep convolutional neural network for picking robot in complex background. *Trans. Chin. Soc. Agric. Eng. (Transactions CSAE)* 35, 164. doi: 10.11975/j.issn.1002-6819.2019.03.021

Doersch, C., Gupta, A. K., and Efros, A. A. (2015). "Unsupervised visual representation learning by context prediction," in *2015 IEEE International Conference on Computer Vision (ICCV)*. 1422–1430.

Ge, Z., Liu, S., Wang, F., Li, Z., and Sun, J. (2021). YOLOX: exceeding YOLO series in 2021. *ArXiv*. doi: 10.48550/arXiv.2107.08430

Grill, J.-B., Strub, F., Altch'e, F., Tallec, C., Richemond, P. H., Buchatskaya, E., et al. (2020). Bootstrap your own latent: A new approach to self-supervised learning. *ArXiv*. 33, 21271–21284.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. (2019). "Momentum contrast for unsupervised visual representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9726–9735.

He, K., Zhang, X., Ren, S., and Sun, J. (2014). 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

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.

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. (2017). "Squeeze-and-excitation networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7132–7141.

Jia, W., Tian, Y., Luo, R., Zhang, Z., Lian, J., and Zheng, Y. (2020). Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. *Comput. Electron. Agric.* 172, 105380. doi: 10.1016/j.compag.2020.105380

Jiao, Y., Luo, R., Li, Q., Deng, X., Yin, X., Ruan, C., et al. (2020). Detection and localization of overlapped fruits application in an apple harvesting robot. *Electronics* 9, 1023. doi: 10.3390/electronics9061023

Jing, L., and Tian, Y. (2019). Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 4037–4058. doi: 10.1109/TPAMI.2020.2992393

Kang, H., and Chen, C. (2019). Fruit detection and segmentation for apple harvesting using visual sensor in orchards. *Sensors* 19, 4599. doi: 10.3390/s19204599

Kang, H., and Chen, C. (2020). Fruit detection, segmentation and 3D visualisation of environments in apple orchards. *Comput. Electron. Agric.* 171, 105302. doi: 10.1016/j.compag.2020.105302

Kuznetsova, A., Maleva, T., and Soloviev, V. (2020). Using YOLOv3 algorithm with pre- and post-processing for apple detection in fruit-harvesting robot. *Agronomy* 10 (7), 1016. doi: 10.3390/agronomy10071016

Li, H., Guo, W., Lu, G., and Shi, Y. (2022). Augmentation method for high intra-class variation data in apple detection. *Sensors (Basel)* 22, 6325. doi: 10.3390/s22176325

Li, X., Wang, W., Hu, X., and Yang, J. (2019). "Selective kernel networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 510–519.

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.

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*. 8759–8768.

Liu, X., Zhao, D., Jia, W., Ji, W., and Sun, Y. (2019). A detection method for apple fruits based on color and shape features. *IEEE Access* 7, 67923–67933. doi: 10.1109/ACCESS.2019.2918313

Minervini, M., Fischbach, A., Scharr, H., and Tsaftaris, S. A. (2016). Finely-grained annotated datasets for image-based plant phenotyping. *Pattern Recognition Lett.* 81, 80–89. doi: 10.1016/j.patrec.2015.10.013

Passalis, N., and Tefas, A. (2018). "Learning deep representations with probabilistic knowledge transfer," in *European Conference on Computer Vision*. (Cham: Springer International Publishing).

Redmon, J., and Farhadi, A. (2018). YOLOv3: an incremental improvement. *ArXiv*. doi: 10.48550/arXiv.1804.02767

Suo, R., Gao, F., Zhou, Z., Fu, L., Song, Z., Dhupia, J., et al. (2021). Improved multi-classes kiwifruit detection in orchard to avoid collisions during robotic picking. *Comput. Electron. Agric.* 182, 106052. doi: 10.1016/j.compag.2021.106052

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," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7464–7475.

Wang, C. Y., Liao, H. Y. M., 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 (CVPRW))*. 1571–1580.

Woo, S., Park, J., Lee, J.-Y., and Kweon, I.-S. (2018). CBAM: convolutional block attention module. *ArXiv*, 3–19. doi: 10.1007/978-3-030-01234-2_1

Wu, L., Ma, J., Zhao, Y., and Liu, H. (2021). Apple detection in complex scene using the improved YOLOv4 model. *Agronomy* 11, 476. doi: 10.3390/agronomy11030476

Yan, B., Fan, P., Lei, X., Liu, Z., and Yang, F. (2021). A real-time apple targets detection method for picking robot based on improved YOLOv5. *Remote Sens.* 13, 1619. doi: 10.3390/rs13091619

Yan, B., Fan, P., Wang, M., Shi, S., Lei, X., and Yang, F. (2022). Real-time apple picking pattern recognition for picking robot based on improved YOLOv5m. *Nongye Jixie Xuebao/Transactions Chin. Soc. Agric. Machinery* 53, 28–38 and 59. doi: 10.6041/j.issn.1000-1298.2022.09.003

Zhang, H., Goodfellow, I. J., Metaxas, D. N., and Odena, A. (2018). Self-attention generative adversarial networks. *ArXiv*, 7354–7363.

Zhang, J., Karkee, M., Zhang, Q., Zhang, X., Yaqoob, M., Fu, L., et al. (2020). Multi-class object detection using faster R-CNN and estimation of shaking locations for automated shake-and-catch apple harvesting. *Comput. Electron. Agric.* 173, 105384. doi: 10.1016/j.compag.2020.105384

Zhuk, S., Tkachenko, M., and Skriabin, N. (2015). *labelImg*. GitHub repository.

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 07 March 2024
DOI 10.3389/fpls.2024.1297390

Check for updates

OPEN ACCESS

EDITED BY
Zhenghong Yu,
Guangdong Polytechnic of Science and Technology, China

REVIEWED BY
Yunchao Tang,
Guangxi University, China
Qing-Wei Wang,
Chinese Academy of Sciences (CAS), China

*CORRESPONDENCE
Qiguan Shu
qiguan.shu@tum.de

RECEIVED 20 September 2023
ACCEPTED 30 January 2024
PUBLISHED 07 March 2024

CITATION
Shu Q, Yazdi H, Rötzer T and Ludwig F (2024) Predicting resprouting of Platanus × hispanica following branch pruning by means of machine learning.
Front. Plant Sci. 15:1297390.
doi: 10.3389/fpls.2024.1297390

COPYRIGHT
© 2024 Shu, Yazdi, Rötzer and Ludwig. 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.

Predicting resprouting of Platanus × hispanica following branch pruning by means of machine learning

Qiguan Shu [1]*, Hadi Yazdi [1], Thomas Rötzer [2] and Ferdinand Ludwig [1]

[1]Professorship for Green Technologies in Landscape Architecture, TUM School of Engineering and Design,Technical University of Munich, Munich, Germany, [2]Chair for Forest Growth and Yield Science, Technical University of Munich, Freising, Germany

Introduction: Resprouting is a crucial survival strategy following the loss of branches, being it by natural events or artificially by pruning. The resprouting prediction on a physiological basis is a highly complex approach. However, trained gardeners try to predict a tree's resprouting after pruning purely based on their empirical knowledge. In this study, we explore how far such predictions can also be made by machine learning.

Methods: Table-topped annually pruned Platanus × hispanica trees at a nursery were LiDAR-scanned for two consecutive years. Topological structures for these trees were abstracted by cylinder fitting. Then, new shoots and trimmed branches were labelled on corresponding cylinders. Binary and multiclass classification models were tested for predicting the location and number of new sprouts.

Results: The accuracy for predicting whether having or not new shoots on each cylinder reaches 90.8% with the LGBMClassifier, the balanced accuracy is 80.3%. The accuracy for predicting the exact numbers of new shoots with the GaussianNB model is 82.1%, but its balanced accuracy is reduced to 42.9%.

Discussion: The results were validated with a separate dataset, proving the feasibility of resprouting prediction after pruning using this approach. Different tree species, tree forms, and other variables should be addressed in further research.

KEYWORDS

tree manipulation, branch pruning, resprout pattern, TLS, tree QSM, machine learning

1 Introduction

Disturbances to tree growth, like ice storms, fires, wind, and diseases, are common in nature (Hauer et al., 2006; Simler et al., 2018). They cause great loss in trees' biomass, especially above the ground. In view of this, resprouting is a vital survival strategy for most tree species: new shoots can grow out of dormant buds rapidly at certain positions after the disturbance. This process is recognized as a major force in forest regeneration (Matula et al., 2019) and significantly impacts forest dynamics (Martini et al., 2008). Humans recognized and harnessed these phenomena from early times (Petit and Watkins, 2003; Candel-Pérez et al., 2022). A famous example is pollarding, where all the shoots of a tree crown are regularly cut off to encourage the growth of new sprouts, which were used as firewood and material for weaving baskets.

Regardless of the practical use, it is a highly interesting but, at the same time, a very complex challenge to understand and predict the resprouting patterns of trees caused by disturbances on a physiological basis. These patterns are firstly determined by axillary buds, which either form new shoots or enter dormancy (Suzuki, 2002). This "decision" is essentially controlled by hormone signals. Auxin was considered one of the primary mediators in the 20th century, while new findings indicate that cytokinins (Salam et al., 2021; Schneider et al., 2022) and strigolactones (Gomez-Roldan et al., 2008) play a major role in apical dominance and branching inhabitation respectively. Without a clear conclusion yet regarding their exact mechanisms, studies tried to understand resprouting patterns from other micro and macro perspectives: its relation to genetic regulation (Hill and Hollender, 2019), in responding to seasonal adaptation (Singh et al., 2022), or by an explanation known as Low Energy Syndrome (Martín-Fontecha et al., 2018).

However, these endogenous physiological processes do not tell the whole story of resprouting. Leaf area and light are redistributed after the disturbances, which then affects photosynthetic processes (Balandier et al., 2000). This does not simply mean a decrease in photosynthetic capacities but involves the reallocation of carbon- and other resources among plant organs such as fruits (Kohek et al., 2015; Tosto et al., 2023) and flowers (Grechi et al., 2022). What makes the impact of this disturbance even more complex is timing. For example, summer pruning on an apple tree typically causes a temporary loss of apical dominance and an increase in its cytokinin supply. But depending on its exact timing, the dominance may be delayed or even prevented (Saure, 1987). As a result, a precise analysis of how a disturbance reshapes a tree using a physiological approach must address the primary status of the hormone, resource reallocation, and the timing issue. To our knowledge, no research has brought all these aspects together so far.

Even without any precise analytical tools regarding resprouting analysis, skilled practitioners learn how to prune a tree in their charge. They neither measure its sap-flows with multiple sensors nor meter the cytokinin concentration in chemistry labs. By going around the tree and observing the main branches, they decide where to prune. Their decisions are based on empirical knowledge of natural phenomena, derived initially from accurate observations of causes and effects – the tree's resprouting reaction to the loss of branches by pruning. Countless repetitions of similar processes have been experimented in horticulture over centuries (Saunders, 1898). For a gardener, their primary pruning skills may start with a set of general rules written in a manual book (Brickell and Joyce, 1996). Then, their skills will independently evolve further through repeated work practices specific to different climate zones, species, etc. Suppose their pruning decisions lead to resprouting reactions largely similar to their expectations, gardeners finally prove to be able to predict the tree's response purely on visual observation and geometrical patterns without digging deep into simulating physiological processes.

In horticulture and arboriculture, we currently see a strong trend toward the automation of pruning by machines or robots (Sam et al., 2022). So far, these are comparatively standardized actions (Li et al., 2021; Sam et al., 2022), but the more complex the tasks become in this regard, the more important is a plausible, robust, and prompt prediction of the growth reaction of a tree to pruning. At the same time, it can be assumed that in the future, trees worldwide will increasingly experience growth disturbances due to the consequences of climate change (drought, stronger and more frequent storms), which will be coupled with a loss of biomass and subsequent resprouting. In order to assess the development of such trees, for example, in an urban context, also here a plausible, robust, and prompt prediction of resprouting in response to the previous loss of branches and twigs is necessary.

In this regard, physiological forecasts seem to be too complex, rely on too many often-unknown parameters (e.g., weather), and thus are likely to be too sensitive to errors and too slow [in reference to, i.e., the applications in forecasting building energy performance (Chakraborty and Elzarka, 2019; Fathi et al., 2020)]. The study at hand aims to develop the basics for a prediction model on the basis of geometric patterns corresponding to the approach of experienced gardeners using a concrete example.

Rapid development in remote sensing is providing a solid base for this aim. First of all, terrestrial LiDAR scanners can capture detailed geometry of objects with a precision of up to 3 mm from multiple standing positions (RIEGL, 2023). This method proves capable of capturing a tree's trunk and branches with more than 10 mm diameter (Gobeawan et al., 2018; Yang et al., 2022) during its leaf-off state (Kükenbrink et al., 2022). Raw data is stored in the form of a discrete point cloud. Furthermore, different approaches have been developed to extract tree structure: skeleton abstraction following occupancy grids (Bucksch et al., 2010; Sun et al., 2022); branch direction by eigenvectors of point patches or sections (Bremer et al., 2013; Raumonen et al., 2013); skeleton as the Dijkstra's shortest path from the tree base to ends (Du et al., 2019; Li et al., 2022, 2022; Wang et al., 2014); skeleton redrawn with searching steps (Hackenberg et al., 2014); learning the reconstruction pattern through a neural network (Liu et al., 2021). Overall, this abstracted information about tree architecture is called the quantitative structure model (QSM) (Åkerblom et al., 2017; Shu et al., 2022). In this way, every segment of the tree stem or branch can be retrieved, containing its diameter, length, axial direction, and hierarchy in the whole branching structure, as well as the pointer to its parent and child segments.

These data for a computational model can be compared with human experiences. The process for an algorithm to "learn from experience" without being explicitly programmed was defined as machine learning (Samuel, 1959). Over 70 years of development, machine learning models have proven capable and efficient to inherently solve the 5 typical problems of data science, namely classification, anomaly detection, regression, clustering, and reinforcement learning (Alzubi et al., 2018). Among them, classification models assign class labels to testing instances where the high dimensional predictor features are known (Kotsiantis, 2007). Specific to our research, QSMs provide the high dimensional features for describing tree segments while the resprouting response of the trees are the class labels. In handling them, the classification models have the advantage of 1) capturing intricate and non-linear patterns within data autonomously (i.e., Hassona et al., 2021). The resprouting patterns are likely to be non-linear to features in QSMs (see section 2.4). 2) They work for both binary and multi-class classification problems (i.e., Teimoorinia et al., 2020). The position of new shoots is a binary problem, while the number of new shoots is a multi-class problem. 3) They have good scalability to large datasets (Gupta et al., 2016). The total number of tree segments can be large (see section 2.2). 4) They can self-update through new training datasets. This allows the prediction to improve its accuracy or be adapted to more species and forms if having corresponding data (see section 4). Additionally, from a practical aspect, open-source packages such as scikit-learn (Pedregosa et al., 2011) have integrated common classification models of machine learning, offering easy access to adapt parameters for different applications. These characteristics collectively make machine learning an attractive and powerful approach for addressing resprouting prediction of trees following pruning.

Equipped with the digital tools above, accurate information regarding tree structures can be collected and processed in analogy to what a real gardener does. Building on this, we are addressing the following questions: How can we predict the position and number of resprouting shoots based on a purely "visual approach" (pattern recognition)? Which machine learning model achieves the best accuracy for this task?

2 Materials and methods

2.1 Study case

To address our questions, we looked for tree cases that are frequently pruned in a distinct manner under similar environmental conditions. At Bruns Nursery, Bad Zwischenahn in north Germany, so-called table-topped plane trees (*Platanus* × *hispanica*) are grown in a clearly defined area under standardized conditions. The crowns of these trees are shaped into a flat layer through labour-intensive maintenance. This form probably originates from Baroque gardens, where plants were kept in an orthogonal manner to enhance the orientation or perspective (Dobrilovič, 2010). Due to the expansion of the crown like an umbrella, it is still used in European cities nowadays for shading squares and pedestrian areas (e.g., the central square at Labouheyre, France). To produce such trees, there are two phases in general. In the first phase, a young plane tree with a naturally grown canopy is intensively trimmed. At around 3 meters in height, six branches are selected and bent horizontally into different directions with equal angles in between. Where necessary, bamboo sticks are added as temporary supports to force the branch into the aimed direction (see "1st year" in Figure 1). In the second phase, new shoots or even some of the older shoots from these six main branches are carefully selected and pruned by experienced gardeners. Pruning decisions are important at this phase to enable shoot growth only at desired positions. Some shoots reserved from previous years could still be trimmed off if there appears another new shoot that becomes a better option. This procedure is repeated in the following years (see "2nd-6th year" in Figure 1). Multiple reiterations of the tree by resprouting result in a complex branching pattern. Due to the annual pruning and relatively complex branching pattern, the second phase of these cases is considered effective in analysing the abilities of machine learning models in predicting resprouting patterns based on quantitative structural tree models under complex yet repetitive conditions. It should be noticed that the aim of this study is not recreating this specific form of tree geometry like the table-topped *Platanus* × *hispanica* but to gain fundamental knowledge regarding resprouting reactions of trees.

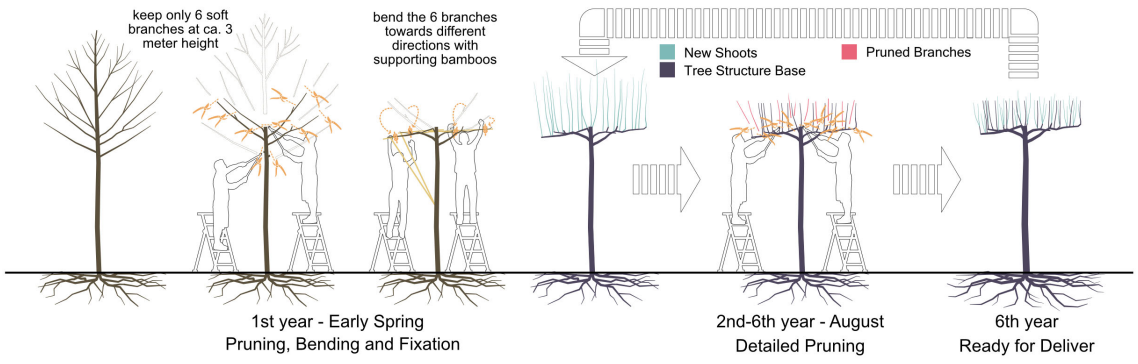


FIGURE 1
The procedure for producing a table-topped *platanus* through iterative branch and shoot selection and pruning with an intensive labour force.

10.3389/fpls.2024.1297390

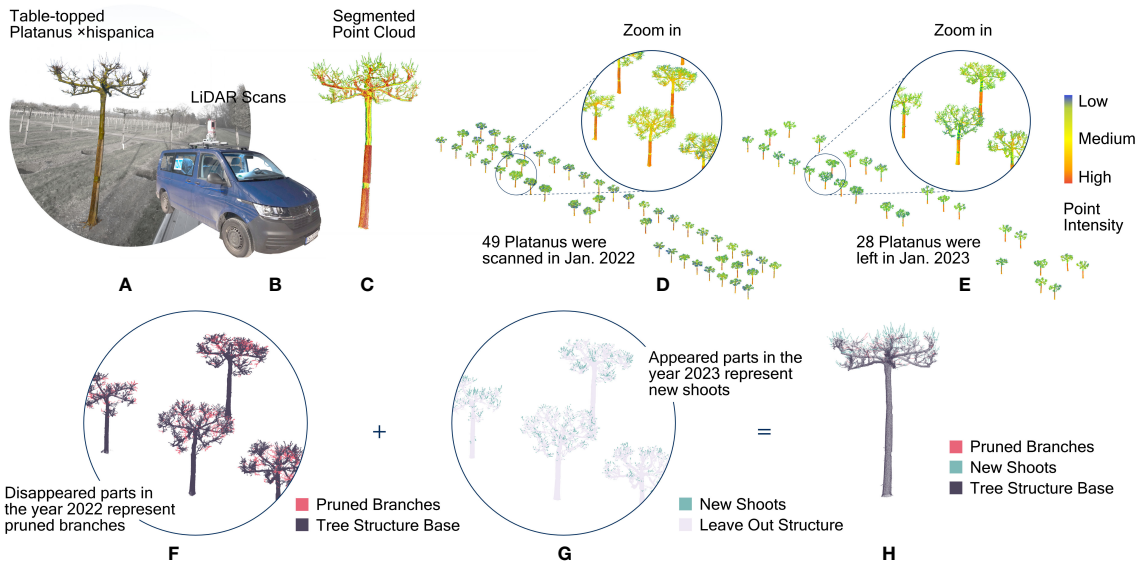


FIGURE 2
The overall procedure for detecting pruned branches and new shoots from point clouds of LiDAR scans in two consecutive years. **(A)** A photo of the table-topped plane trees grown at the nursery; **(B)** LiDAR scanner mounted on the vehicle; **(C)** segmented point cloud of the tree shown in the photo; **(D)** the segmented point clouds of individual plane trees in 2022; **(E)** the segmented point clouds of individual plane trees in 2023; **(F)** labeled points representing the pruned branches on the point clouds acquired in 2022; **(G)** labeled points representing the new shoots on the point clouds acquired in 2023; **(H)** an integrated point cloud with points labeled as unchanged structure base, pruned branches and new shoots.

2.2 Data acquisition and pre-processing

In the subsequent two winters, namely in January 2022 and January 2023, an area consisting of 3- and 4-year-old table-topped Platanus (see Figure 2A) planted in 3 rows at Bruns Nursery were scanned with LiDAR scanner RIEGL VZ-400i. The scanner was mounted on a tripod in 2022, while mounted on a vehicle (see Figure 2B) in 2023. All the scans were set to the "Panorama30" standard (with angular resolution 0.030°) and conducted in a "stop-and-go" method. Scanning positions were located along each row at every third tree (ca. 12 m). Point clouds from different scan positions were automatically registered in RiSCAN Pro in reference to GNSS coordinates recorded with Leica Zeno FLX100 plus (Leica, 2023; Yazdi et al., 2024). The original GNSS coordinates indicate accuracies ranging between 0.68 to 0.80 m at different scan positions. Therefore, the reliability of GNSS was set to low during the automatic registration and the multistation adjustment. With all the steps above, we got two point clouds containing all the tree cases for the years 2022 and 2023, respectively. Afterward, individual trees were segmented manually (see Figure 2C). This manual step is efficient for our cases because those trees planted in the nursery were almost perfectly aligned at an equal distance, and their crowns did not touch each other. The ground surface was flat and clear. There were no irrelevant objects, such as shrubs around tree trunks. A total of 49 plane trees were scanned in 2022 while the number of trees scanned in 2023 was 28 (due to tree sales during 2022, see Figures 2D, E). As a result, we got point clouds of 28 plane trees for both years.

The next step was to identify changes in the geometrical structure of the trees in these two years (see Figure 2H). For this purpose, the two corresponding scans regarding the same trees must be aligned. The GNSS coordinates have an offset of up to 0.8 meters, which is insufficient for our demand. The most common algorithm for matching 3d models precisely, namely Iterative Closest Point (Rusinkiewicz and Levoy, 2001), does not work for these tree cases because the new shoots and the extensive pruning on tree branches have altered their geometries significantly. A supervised alignment by manually picking point pairs on corresponding branch surfaces also caused visible deviations owing to the girth growth. Finally, we manually aligned all tree pairs individually using multiple views. This guaranteed the best possible alignment despite significant geometrical changes between the two scans. Only then were we able to precisely detect the changes caused by growth and pruning between the point clouds. In principle, point sets that only appeared in the scan of 2022 and disappeared in the scan of 2023 should represent branches pruned away. Conversely, point sets that were only found in 2023 should represent new shoots. In practice, an object has no identical points on its surface in two independent scans. To identify geometrical changes on the two point clouds, cloud-to-cloud distance (Jafari et al., 2017) was applied. For each point in one point cloud, this function calculates its distance to its nearest neighbour in the other point cloud using the Hausdorff distance (Taha and Hanbury, 2015). This calculation was conducted in CloudCompare, where the octree level is set to "auto" (Girardeau-Montaut, 2023). Based on the cloud-to-cloud distance values, a minimum distance threshold ranging between 0.020 to 0.045 m was customized to each point cloud for segmenting unchanged and changed tree segments (see Figures 2F, G). When the alignment of the tree was precise, and little noise was around the branches, the threshold was set smaller to tell apart more accurate changes. Points whose distances were larger than the thresholds represent tree segments

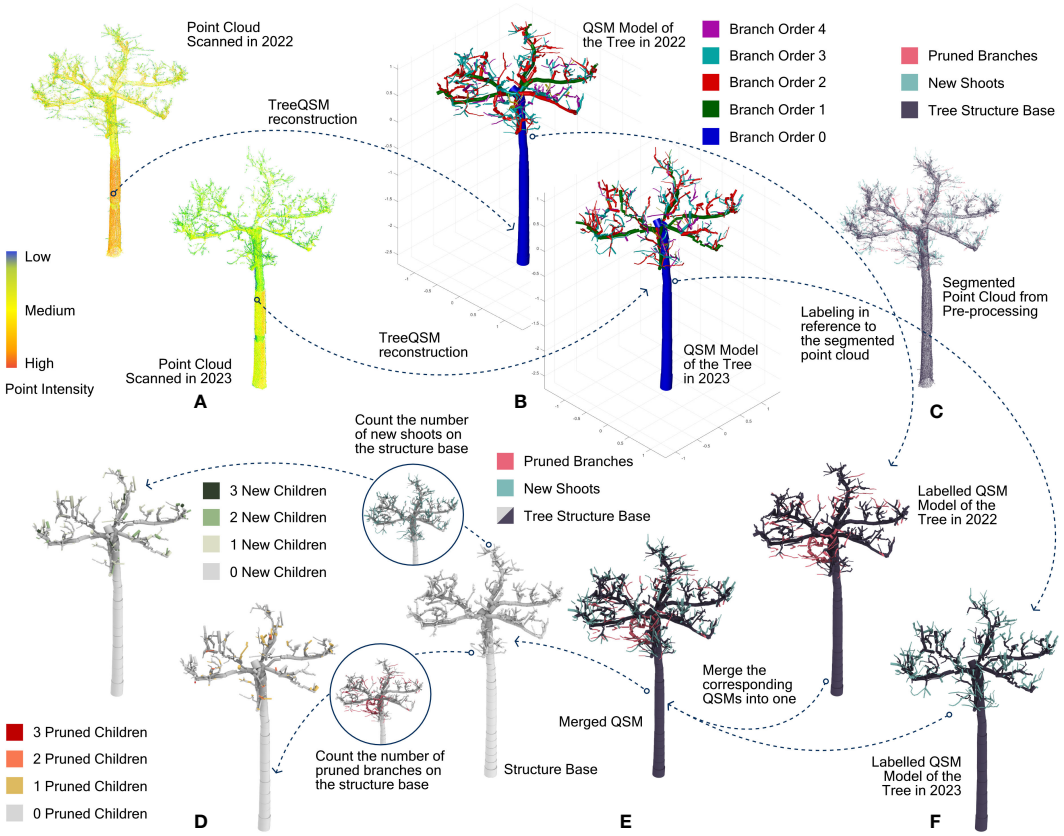


FIGURE 3
The overall procedure for labelling and reorganizing the dataset. **(A)** point clouds of the same tree scanned in 2022 and 2023 respectively; **(B)** QSM models out of the point clouds; **(C)** labeled point cloud as the reference; **(D)** final labeled QSM dataset consisting of the unchanged structure base and the numbers of their pruned and new children cylinders; **(E)** integrated QSM with cylinders labeled as unchanged structure base, pruned branches and new shoots. **(F)** labeled QSMs representing the trees scanned in 2022 and 2023, respectively.

that do not exist in the other scan. For those points in the scan of 2022, those changed points represent pruned branches, while those points in the scan of 2023 represent new shoots (see Figure 2H).

Parallel to change detection, the point clouds were also used to create quantitative structural models (QSMs) of the trees (see Figure 3A) by TreeQSM (Raumonen et al., 2013) in MATLAB (The MathWorks Inc., 2023) (see Figure 3B). Raumonen et al. (2013) integrated multiple automatic steps in this pipeline to recreate precise cylindrical models out of the dense point cloud of an individual tree. The main steps are defining small sets of patches on tree surfaces; segmenting patches into a trunk and branches using iterative searching steps; fitting cylinders on point clouds of the same branch; optimizations to reduce the error caused by noises and occultations; generating statistics on cylinders and the tree. Besides TreeQSM, some other open-source QSM reconstructing tools like AdTree (Du et al., 2019) and AdQSM (Fan et al., 2020) build tree structures by the Dijkstra's shortest path and the minimum spanning tree, respectively. In primary tests by the authors, they appeared to be more sensitive to outliers in our dataset. Primarily when they built detailed twigs at the branch's high end, shoots were invented on fake skeletons initiated by the outliers in the point clouds. Therefore, they did not reflect the actual sprouting pattern. Compared to them, TreeQSM fits only cylinders to point patches in defined sizes. This approach performs better in noise and outlier resistance than those methods using Dijkstra's shortest path and the minimum spanning tree, being the most faithful in describing the accurate tree geometries among the mentioned tools. One limitation of the TreeQSM tool lies in the robustness of the branch segmentation due to some random seeds in patch generation. Following the manual book (Raumonen, 2022), we tested 18 configurations of different settings regarding the patch sizes for reconstructing the QSMs in TreeQSM on each point cloud. For each configuration further, the reconstruction was repeated 15 times to reduce the impacts of pseudo-random numbers. Finally, the QSM with minimum mean distances from points to trunk and branch cylinders was chosen as the model for the corresponding point cloud using the embedded function named "select_optimum". It should be addressed again that in our dataset, each tree is represented with two different point clouds and two QSMs accordingly, showing their stands in 2022 and 2023 respectively. To further ensure a precise reconstruction, the outliers were pre-deleted through the statistical outlier removal (SOR) tool (Rusu and

Cousins, 2011). This step was implemented in CloudCompare, where the number of points used for mean distance estimation was set to 6. The standard deviation multiplier threshold was set to 1.

2.3 Labelling and reorganizing the dataset

In the pre-processing, the trimmed branches and the new shoots were detected in the point clouds, while topological cylinders were generated with TreeQSM. The next step was to combine these two datasets. The individual cylinders of the QSMs must be labelled as to whether they are part of an unchanged branch (not considering the girth growth), a pruned branch or a new shoot. This was achieved by using a distance threshold between points of the cylindrical axis and their nearest neighbouring point of the segmented point clouds. For our data, we examined only every cylinder's start and end point. If the sum of their mean distances to their 10 nearest neighbours with the same label (i.e., trimmed branches) was below 100 mm, this cylinder was labelled the same (see Figures 3C, F). To enhance the accuracy of the labelling, three more criteria were added based on practical rules when pruning these trees: for any cylinder labelled as part of either a new shoot or a pruned branch, its radius must be smaller than 20 mm (one-year-old shoots do not reach more than 20 mm in diameter for the trees at hand); for any cylinder labelled as part of a pruned branch, its branch hierarchical order must be larger than 1 (not the tree trunk and the primary branch); the label for trimmed branches and new shoots on one cylinder is passed on to all its children cylinders.

After labelling, the cylinders of different labels (unchanged branches, pruned branches and new shoots) are still separated in two QSMs regarding the same tree. There is no correspondence between these two QSMs, as their reconstruction processes are independent. Therefore, cylinders of the trimmed branches in one QSM must be integrated into the other QSM that contains the main tree structure and the new shoots, or reversely, cylinders of new shoots must be integrated into the QSM with the trimmed branches. This is a tricky process. While the geometric data remain the same for every cylinder, its topological data regarding the ID of the cylinder, its parent cylinder, and its child cylinder must be corrected, as well as the branch order and its position in the branch. Regarding whether to transfer cylinders of new shoots or pruned branches to the other QSM, considerations can be described as follows. The pruned branches, in general, could only be the same size or thicker than the new shoots. Consequently, cylinders of pruned branches have higher robustness in their position through cylinder fitting. As a result, the certainty of redefining their topological parent in another QSM based on their relative positions is supposed to be higher. So, for our dataset, the cylinders of pruned branches were picked out from their original QSM and integrated into the other QSM that has the new shoot cylinders (see Figure 3E). Their new parent cylinders were redefined as those whose endpoints were located closest to their starting point. Based on this, the topological data for every single cylinder in the newly merged QSM were entirely overwritten due to this change.

Finally, the total number of pruned branches and new shoots on every cylinder was counted (see Figure 3D). This became the crucial attribute for the prediction models in the next step.

2.4 Prediction with various classification models

After all the processes described above, the dataset contains 34,245 items, representing 28 table-topped plane trees. Each item corresponds to one cylinder, which contains the following attributes: tree's ID; cylinder's ID; parent cylinder's ID; child cylinder's ID in the same branch; x-y-z coordinate of the cylinder start; a normalized 3d vector of the axial direction; branch's ID; its sequence in the branch; branch order; cylinder length; cylinder radius; the number of pruned children and new children; the Boolean value if this cylinder is virtually added during QSM reconstruction; the Boolean value if this cylinder is pruned out.

The relationships between each two attributes (except for the IDs and Boolean values) are illustrated in Supplementary Figure 2. For our research purpose, the sprout location and numbers are the labels of new shoots on each cylinder. We tested classification models in machine learning to find links between these topological and geometrical attributes and the predicting target. Among these target labels, 16,183 (47.3%) cylinders were labelled "-1", meaning that they were trimmed away. These cylinders are not feeding into machine learning models. 15,348 (44.8%) cylinders have no new shoot, thus labelled with "0". 2,329 (6.8%) cylinders have one new shoot (labelled "1"). There are fewer cylinder samples, whose new shoot number is larger than "1": 321 (0.94%) cylinders have 2 new shoots; 54 (0.16%) cylinders have 3 new shoots; 7 (0.02%) cylinders have 4 new shoots; 2 cylinders have 5 new shoots; only 1 cylinder has 6 new shoots on it. Due to the extremely rare samples with a high number of new shoots, we label those cylinders that have more than 4 shoots with new shoot number 4.

Owing to the limited volume of data we acquired, the majority of the items labelled with new shoot numbers from "0" to "4" must feed into machine learning models (16,558 items representing 26 trees). Nevertheless, we reserved 2 trees (1,504 items) as an evaluation dataset. This evaluation dataset was only used to validate the results (see section 3), not to train the model. The dataset for machine learning was further divided into a training set (13,246 items) and a testing set (3,312 items, with a test size of 0.2). The testing set prevented overfitting the models to the given data.

For getting a quick overview of the performances across a wide range of classification models in machine learning on the dataset, we used lazy predict (Pandala, 2023) to run scikit-learn (Pedregosa et al., 2011) to compare 25 common classification models with their default settings, including GaussianNB, NearestCentroid and LGBMClassifier. Besides, we tested a basic Artificial Neural Network (ANN) model built with Keras (Chollet, 2015). It consisted of two hidden layers with 64 and 128 nodes, respectively (see Figure 4 left). In addition, to examine a graph neural network (GNN) model, the dataset for each tree was processed to a graph (Salama, 2021), where every cylinder item

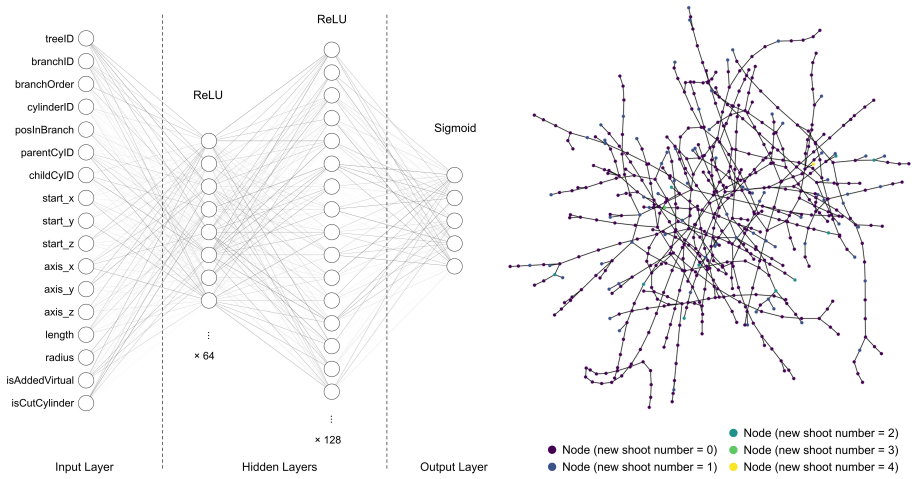


FIGURE 4
Architecture of the ANN (left) and Graph (right) of one tree used in our test.

was a node connected to its parent and children (the node connection for one tree is illustrated in Figure 4 right). These graph data were fed into a GNN model named "baseline classifier" (see Supplementary Table 2.1), including 39,512 trainable and 1,174 non-trainable params.

We tested all these classification models in two manners of labelling: binary labels that only classify cylinders if they will or will not grow new shoots; multiclass labels that categorize cylinders based on the exact number of new shoots ranging between 0 to 4.

3 Results

The accuracy, balanced accuracy, and F1 Score (weighted average F1 score for multiclass labels) of the tested models in a default setting or with a basic architecture (see section 2.4) are listed in Figure 5. Each scoring index ranges between 0 and 1. 1 is the best score, meaning that all the shoot labels are correctly predicted. On the contrary, 0 is the worst score, representing no correct prediction. In the figure, these models are shown in descending order from the left to the right according to their total scores in classifying binary labels. Among the three sub-scores, accuracy reflects an overall rate of true predictions for all labels. Our datasets are imbalanced in terms of different label numbers. Therefore, balanced accuracy, which gives equal weights to the true prediction rates for each label, is also an important indicator in evaluating their performances. The F1 score is another effective index for the imbalanced classifications but attaches more importance to true positives (predicting the cylinders with new shoots correctly), while it ignores the true negatives (predicting the

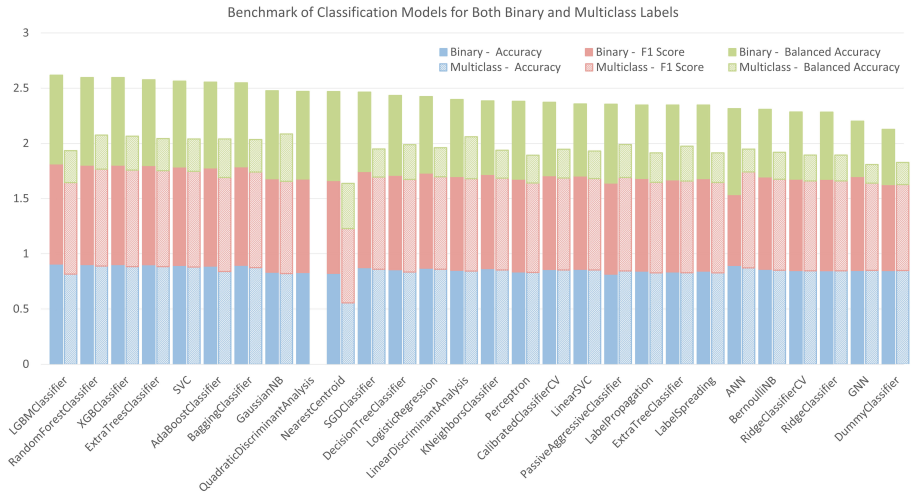


FIGURE 5
Benchmark of tested classification models for binary and multiclass labelling.

cylinders with zero shoot correctly). Based on these benchmark scores, LGBMClassifier and GaussianNB have top scores for predictions with binary and multiclass labels, respectively. The confusion matrix of the LGBMClassifier with binary labels in the testing set is shown in Supplementary Table 3.1. The confusion matrix of the GaussianNB model with multiclass labels in the testing set is shown in Supplementary Table 3.2.

To validate these two models further, we applied the trained LGBMClassifier model and GaussianNB model to the evaluation set with binary and multiclass labels respectively. The results of the evaluation are visually illustrated in Supplementary Figure 3. Their performance metrics including precision, recall, and F1 Score for every label on the validation set are shown in Supplementary Tables 3.3 and 3.4. The accuracy, balanced accuracy, and weighted F1 score of both models with the evaluation set (only 2 trees) have a maximum of around 10% difference from the scores on the benchmark.

4 Discussion

To be able to meaningfully interpret and evaluate the results, it is first necessary to discuss the specific conditions of the dataset and resulting limitations.

The following factors may impact the accuracy of the extracted geometrical data from the trees: 1) To prevent browsing the tree barks, protecting covers were installed below 2 meters around the tree trunks. This might have caused the diameter measured at trunk cylinders to be slightly overestimated. However, we assume that this has no influence on the prediction model. 2) Minor swinging of the branches by wind during the LiDAR scanning might have caused outliers or might have led to overestimating the diameter of the smaller branches. Although the point clouds were denoised through SOR filters, this does not guarantee the full deletion of these outliers and could then cause inexistent branches in the cylindrical models. 3) Aligning the same trees with different geometries in the two years has been a nonstandard manual process so far, which can cause inconsistency in change detection and identification of parent cylinders. A possible alternative to detect these changes is comparing the occupancy grids (Hirt et al., 2021).

The total number of cylinders for training the models was limited to 16,558, representing 26 trees. The percentage of the negative label "0" makes up more than 92% of the total items, causing an unbalanced rate for the number of positive samples (less than 2500 items). Unfortunately, these are all available data from the nursery.

Most importantly, the collected dataset in two consecutive years reflects the growth of these trees under almost identical environmental conditions and pruning regimes. More specifically, the temperature, water content in the soil, wind direction and speed as well as the time of pruning are all the same for these trees. This means that our method can predict the resprouting pattern of this kind of table-topped plane trees grown under the same conditions as in this study. In case of any changes in the factors mentioned above, it is unclear so far how accurate the prediction will be. For instance, the model may not predict the growth of the same trees in the following year. Horticultural experience even shows that a change in the time of pruning of only one or two weeks can have a significant impact on the growth of new shoots, especially if there is also a change in weather conditions (e.g., heat or drought immediately after pruning).To understand whether those environmental factors could also be addressed in a prediction model in the same approach, these environmental data must be collected and coupled with a larger quantity of trees. This hints at an upcoming step in this study.

Except for the barriers in data quality, its available amount, and environmental descriptors above, the following technical difficulties in this computational workflow may be worth paying attention to. 1) Merging the QSMs of one tree scanned at different times is not robust. To improve this, a reference-based cylinder fitting function should be considered. In this way, the later QSM of the tree can be built based on its previous QSMs. Then, the girth growth for each cylinder can be precisely linked from time to time following this idea. 2) For pruning and resprouting issues, positions and lengths of actual internodes are more helpful than current cylinders that contain only geometrical information and lack connections to physiological processes. Axillary buds can possibly be identified in detailed, colorful images of the tree trunks or branches. These can be used for fitting cylinders faithful between physiological nodes of the plants. 3) The LGBMClassifier and GaussianNB models are lightweight and efficient. They have shown the best performance on our relatively small dataset. If they were applied to bigger databases, their accuracy remains to be evaluated, especially in handling a higher diversity in tree ages and shapes. 4) After predicting the position of new shoots, our current model did not answer the ongoing growth of those shoots. It is possible to combine a L-system growth simulation (Boudon et al., 2012) with the QSM (see Shu et al., 2022). In this way, our model can be integrated as a tool to interrupt a natural growth through branch pruning.

Finally, our current model is only the first step in understanding resprouting patterns after one specific artificial disturbance, namely pruning of table-topped trees. Nonetheless, we are optimistic that the approach has great potential for further development and application (see e.g. Yazdi et al., 2023). The application of such a model is not limited to repeating what the gardeners can already do but goes beyond knowledge boundaries regarding the resprouting strategy of trees after disturbances. This can hopefully be achieved through gathering massive tree database (e.g. Yazdi et al., 2024). By searching this database, the "digital gardener" will likely find evidence to support its predictions in a more complex context. In agricultural automation, robots are already self-navigating through an orchard (Ye et al., 2023) and picking fruits (Meng et al., 2023; Wang et al., 2023). Following this trend, this study may offer hints about how pruning decisions could be made by the "digital gardener". For this far vision, an open-source and uniform data platform about trees [e.g., tree information modeling (Shu et al., 2022)] is required.

5 Conclusion

Resprouting patterns are vital in understanding the regeneration of trees after natural and artificial disturbances. The interrelationships are very complex, involving the primary status of hormones, the redistribution of resources, and timing issues. Until now, no single

model has addressed all these factors with a physiological approach. However, for centuries, gardeners and practitioners have been trained to prune trees based on their intuitive predictions. They are able to do so based on accumulated knowledge working with trees. In this study, we gave it a first try addressing whether computational models, especially machine learning models, could gain similar knowledge as practitioners from horticulture: what are the location and numbers of new shoots after pruning? Which model would achieve the best performance?

For this purpose, we scanned a group of annually pruned plane trees at a tree nursery with LiDAR. The detailed geometry and topology of the branches were extracted through quantitative tree models. The trimmed branches and new shoots were detected through comparison between the scans in two consecutive years. This information was finally labelled on a dataset for training multiple classification models.

We tested 25 common classification models in machine learning with default settings. Additionally, 1 ANN model and 1 GNN model with the most basic architectures were also tested. Among these models, except for two, all other models have an accuracy and an F1 score higher than 80%. For balanced accuracy, the average score of all the models was ca. 70% for binary labels; for multiclass labels, the average was 28.3%.

From the results, we can conclude that for the collected dataset, most models work well in telling the position of new shoots but are not accurate in describing the actual shoot numbers at the specific location. For the best scored models with binary labelling, the LGBMClassifier can predict the position of new shoots with an accuracy of 90.8% and a balanced accuracy of 80.3%. For predicting the exact number of the shoots, the GaussianNB Model performs the best. The accuracy is 82.1% because most cylinders should have the shoot number 0. However, the balanced accuracy is reduced to 42.9%.

The innovation of this work was to identify the tree cases in a controlled environment for studying their quantitative reactions to disturbances. It is the first study to address the resprouting pattern prediction with QSM data. To achieve this, it is highly novel to combine QSMs of different times of a tree into one. It is also of significant value to indicate a primary comparison of the performances of various machine learning models in this task.

The applicability of the current model is limited to the studied site, environmental conditions, tree species and form, and the pruning time. In the next step, a larger amount of tree data is being collected in the city of Munich to analyse how this approach can be extended to a broader scope, maybe addressing some of the environmental factors. In a further vision, a massive database of the "digital gardener" would push forward the boundaries of knowledge in understanding the resprouting strategies of trees facing natural and artificial disturbances.

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

QS: Conceptualization, Data curation, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. HY: Data curation, Investigation, Software, Validation, Writing – review & editing. TR: Funding acquisition, Resources, Supervision, Writing – review & editing. FL: Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. DFG-DACH-project funds this study under No. DFG-GZ: LU2505/2-1 AOBJ:683826 and cooperated with the DFG funded research groups under No. 437788427-RTG2679, PR 292/23-1 and RO 4283/2-1.

Acknowledgments

Great credit to Bruns Pflanzen, who provided the table-topped plane trees as studying cases. This study would not have been possible without their dedication to scientific research and support. Thanks to Luke Bohnhorst, who coordinated the use of the RIEGL LiDAR scanner and the license of RISCAN Pro with us. Finally, the deep learning tutoring materials by Amir Ali and the graph Neural Networks tutoring materials by Khalid Salama were of great help to this study.

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/fpls.2024.1297390/full#supplementary-material

References

Åkerblom, M., Raumonen, P., Mäkipää, R., and Kaasalainen, M. (2017). Automatic tree species recognition with quantitative structure models. *Remote Sens. Environ.* 191, 1–12. doi: 10.1016/j.rse.2016.12.002

Alzubi, J., Nayyar, A., and Kumar, A. (2018). Machine learning from theory to algorithms: an overview. *J. Physics: Conf. Ser.* 1142, 12012. doi: 10.1088/1742-6596/1142/1/012012

Balandier, P., Lacointe, A., Roux, X. L., Sinoquet, H., Cruiziat, P., and Dizès, S. L. (2000). SIMWAL: A structural-functional model simulating single walnut tree growth in response to climate and pruning. *Ann. For. Sci.* 57, 571–585. doi: 10.1051/forest:2000143

Boudon, F., Pradal, C., Cokelaer, T., Prusinkiewicz, P., and Godin, C. (2012). L-py: an L-system simulation framework for modeling plant architecture development based on a dynamic language. *Front. Plant Sci.* 3. doi: 10.3389/fpls.2012.00076

Bremer, M., Rutzinger, M., and Wichmann, V. (2013). Derivation of tree skeletons and error assessment using LiDAR point cloud data of varying quality. *ISPRS J. Photogrammetry Remote Sens.* 80, 39–50. doi: 10.1016/j.isprsjprs.2013.03.003

Brickell, C., and Joyce, D. (1996). *Royal Horticultural Society: Pruning & Training*. Available online at: https://www.cabdirect.org/cabdirect/abstract/19970300717.

Bucksch, A., Lindenbergh, R., and Menenti, M. (2010). SkelTre. *Visual Comput.* 26, 1283–1300. doi: 10.1007/s00371-010-0520-4

Candel-Pérez, D., Hernández-Alonso, H., Castro, F., Sangüesa-Barreda, G., Mutke, S., García-Hidalgo, M., et al. (2022). 250-Year reconstruction of pollarding events reveals sharp management changes in Iberian ash woodlands. *Trees* 36, 1909–1921. doi: 10.1007/s00468-022-02343-8

Chakraborty, D., and Elzarka, H. (2019). Advanced machine learning techniques for building performance simulation: A comparative analysis. *J. Building Perform. Simulation* 12, 193–207. doi: 10.1080/19401493.2018.1498538

Chollet, F. (2015). *Keras*. Available online at: https://keras.io/.

Dobrilovič, M. (2010). Vegetation elements in baroque gardens (The influence of foreign plants on the baroque programme). *Acta Hortic.* 881, 899–904. doi: 10.17660/ActaHortic.2010.881.151

Du, S., Lindenbergh, R., Ledoux, H., Stoter, J., and Nan, L. (2019). AdTree: accurate, detailed, and automatic modelling of laser-scanned trees. *Remote Sens.* 11 (18), 2074. doi: 10.3390/rs11182074

Fan, G., Nan, L., Dong, Y., Su, X., and Chen, F. (2020). AdQSM: A new method for estimating above-ground biomass from TLS point clouds. *Remote Sens.* 12 (18), 3089. doi: 10.3390/rs12183089

Fathi, S., Srinivasan, R., Fenner, A., and Fathi, S. (2020). Machine learning applications in urban building energy performance forecasting: A systematic review. *Renewable Sustain. Energy Rev.* 133, 110287. doi: 10.1016/j.rser.2020.110287

Girardeau-Montaut, D. (2023). Cloud-to-Cloud Distance—CloudCompareWiki *[Open-source document]. Cloud-to-Cloud Distance*. Available online at: https://www.cloudcompare.org/doc/wiki/index.php/Cloud-to-Cloud_Distance.

Gobeawan, L., Lin, E. S., Tandon, A., Yee, A. T. K., Khoo, V. H. S., Teo, S. N., et al. (2018). Modeling trees for virtual singapore: from data acquisition to citygml models. *Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci.* XLII-4/W10, 55–62. doi: 10.5194/isprs-archives-XLII-4-W10-55-2018

Gomez-Roldan, V., Fermas, S., Brewer, P. B., Puech-Pagès, V., Dun, E. A., Pillot, J.-P., et al. (2008). Strigolactone inhibition of shoot branching. *Nature* 455, 189–194. doi: 10.1038/nature07271

Grechi, I., Persello, S., Boudon, F., and Normand, F. (2022). Pruning intensity and severity affect flowering processes in the mango tree. *Acta Hortic.* 1346, 67–74. doi: 10.17660/ActaHortic.2022.1346.9

Gupta, P., Sharma, A., and Jindal, R. (2016). Scalable machine-learning algorithms for big data analytics: A comprehensive review. *WIREs Data Min. Knowledge Discovery* 6, 194–214. doi: 10.1002/widm.1194

Hackenberg, J., Morhart, C., Sheppard, J., Spiecker, H., and Disney, M. (2014). Highly accurate tree models derived from terrestrial laser scan data: A method description. *Forests* 5 (5), 1069–110. doi: 10.3390/f5051069

Hassona, S., Marszalek, W., and Sadecki, J. (2021). Time series classification and creation of 2D bifurcation diagrams in nonlinear dynamical systems using supervised machine learning methods. *Appl. Soft Computing* 113, 107874. doi: 10.1016/j.asoc.2021.107874

Hauer, R., Dawson, J., and Werner, L. (2006). "Trees and Ice Storms: The Development of Ice Storm–Resistant Urban Tree Populations (Second Edition)," in *USDA Forest Service/UNL Faculty Publications*. (Lincoln, United States: UNL Faculty Publications). Available at: https://digitalcommons.unl.edu/usdafsfacpub/295.

Hill, J. L., and Hollender, C. A. (2019). Branching out: New insights into the genetic regulation of shoot architecture in trees. *Curr. Opin. Plant Biol.* 47, 73–80. doi: 10.1016/j.pbi.2018.09.010

Hirt, P.-R., Xu, Y., Hoegner, L., and Stilla, U. (2021). Change detection of urban trees in MLS point clouds using occupancy grids. *PFG – J. Photogrammetry Remote Sens. Geoinformation Sci.* 89, 301–318. doi: 10.1007/s41064-021-00179-4

Jafari, B., Khaloo, A., and Lattanzi, D. (2017). Deformation tracking in 3D point clouds via statistical sampling of direct cloud-to-cloud distances. *J. Nondestructive Eval.* 36, 65. doi: 10.1007/s10921-017-0444-2

Kohek, Š., Guid, N., Tojnko, S., Unuk, T., and Kolmanič, S. (2015). EduAPPLE: interactive teaching tool for apple tree crown formation. *HortTechnology* 25, 238–246. doi: 10.21273/HORTTECH.25.2.238

Kotsiantis, S. (2007). Supervised machine learning: A review of classification techniques. *Informatica (Slovenia)* 31, 249–268. Available at: https://www.informatica.si/index.php/informatica/article/view/148.

Kükenbrink, D., Marty, M., Bösch, R., and Ginzler, C. (2022). Benchmarking laser scanning and terrestrial photogrammetry to extract forest inventory parameters in a complex temperate forest. *Int. J. Appl. Earth Observation Geoinformation* 113, 102999. doi: 10.1016/j.jag.2022.102999

Leica. (2023). Leica Zeno FLX100 plus | Smartantenne *[Leica Zeno FLX100 plus Smartantenne - Eine genaue, kompakte und flexible Lösung für alle]*. Available online at: https://leica-geosystems.com/de-de/products/gis-collectors/smart-antennas/leica-zeno-flx100-plus.

Li, M., Ma, L., Zong, W., Luo, C., Huang, M., and Song, Y. (2021). Design and experimental evaluation of a form trimming machine for horticultural plants. *Appl. Sci.* 11 (5), 2230. doi: 10.3390/app11052230

Li, J., Wu, H., Xiao, Z., and Lu, H. (2022). 3D modeling of laser-scanned trees based on skeleton refined extraction. *Int. J. Appl. Earth Observation Geoinformation* 112, 102943. doi: 10.1016/j.jag.2022.102943

Liu, Y., Guo, J., Benes, B., Deussen, O., Zhang, X., and Huang, H. (2021). TreePartNet: Neural decomposition of point clouds for 3D tree reconstruction. *ACM Trans. Graphics* 40 (6), 232. doi: 10.1145/3478513.3480486

Martín-Fontecha, E. S., Tarancón, C., and Cubas, P. (2018). To grow or not to grow, a power-saving program induced in dormant buds. *Curr. Opin. Plant Biol.* 41, 102–109. doi: 10.1016/j.pbi.2017.10.001

Martini, A. M. Z., Lima, R. A. F., Franco, G. A. D. C., and Rodrigues, R. R. (2008). The need for full inventories of tree modes of disturbance to improve forest dynamics comprehension: An example from a semideciduous forest in Brazil. *For. Ecol. Manage.* 255, 1479–1488. doi: 10.1016/j.foreco.2007.11.004

Matula, R., Šrámek, M., Kvasnica, J., Uherková, B., Slepička, J., Matoušková, M., et al. (2019). Pre-disturbance tree size, sprouting vigour and competition drive the survival and growth of resprouting trees. *For. Ecol. Manage.* 446, 71–79. doi: 10.1016/j.foreco.2019.05.012

Meng, F., Li, J., Zhang, Y., Qi, S., and Tang, Y. (2023). Transforming unmanned pineapple picking with spatio-temporal convolutional neural networks. *Comput. Electron. Agric.* 214, 108298. doi: 10.1016/j.compag.2023.108298

Pandala, S. R. (2023). *GitHub Master—Shankarpandala/lazypredict.* (GitHub). Available at: https://github.com/shankarpandala/lazypredict/blob/master/docs/index.rst.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* 12, 2825–2830. doi: 10.5555/1953048.2078195

Petit, S., and Watkins, C. (2003). Pollarding trees: changing attitudes to a traditional land management practice in britain 1600–1900. *Rural History* 14, 157–176. doi: 10.1017/S0956793303001018

Raumonen, P. (2022). "TreeQSM/Manual at master," in *InverseTampere/TreeQSM* (GitHub). Available at: https://github.com/InverseTampere/TreeQSM/tree/master/Manual.

Raumonen, P., Kaasalainen, M., Åkerblom, M., Kaasalainen, S., Kaartinen, H., Vastaranta, M., et al. (2013). Fast Automatic precision tree models from terrestrial laser scanner data. *Remote Sens.* 5 (2), 491–520. doi: 10.3390/rs5020491

RIEGL, L. M. S. G. (2023). *RIEGL - Produktdetail*. Available online at: http://www.riegl.com/nc/products/terrestrial-scanning/produktdetail/product/scanner/48/.

Rusinkiewicz, S., and Levoy, M. (2001). "Efficient variants of the ICP algorithm," in *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*. (Quebec City, QC, Canada: IEEE), 145–152. doi: 10.1109/IM.2001.924423

Rusu, R. B., and Cousins, S. (2011). "3D is here: point cloud library (PCL)," in *2011 IEEE International Conference on Robotics and Automation*. (Shanghai, China: IEEE), 1–4. doi: 10.1109/ICRA.2011.5980567

Salam, B. B., Barbier, F., Danieli, R., Teper-Bamnolker, P., Ziv, C., Spíchal, L., et al. (2021). Sucrose promotes stem branching through cytokinin. *Plant Physiol.* 185, 1708–1721. doi: 10.1093/plphys/kiab003

Salama, K. (2021). *Keras documentation: Node Classification with Graph Neural Networks*. Available online at: https://keras.io/examples/graph/gnn_citations/.

Sam, S., Mira, L., and Kai, S. (2022). The impact of digitalization and automation on horticultural employees – A systematic literature review and field study. *J. Rural Stud.* 95, 560–569. doi: 10.1016/j.jrurstud.2022.09.016

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.* 3, 210–229. doi: 10.1147/rd.33.0210

Saunders, W. (1898). Pruning of Trees and Other Plants. In Hill, G. W. (Eds.), *Yearbook of the United States Department of Agriculture*. (Washington: Goverment Printing Office), 151–166. Available at: https://books.google.de/books?id=FgCjQVa4eJ0C.

Saure, M. C. (1987). Summer pruning effects in apple—A review. *Scientia Hortic.* 30, 253–282. doi: 10.1016/0304-4238(87)90001-X

Schneider, A., Boudon, F., Demotes-Mainard, S., Sakr, S., Godin, C., and Bertheloot, J. (2022). Control of lateral meristems: how is sugar availability involved in the environmental control of axillary bud outgrowth? In: J. Bertheloot, J.-L. Durand and C. Godin (ed.). *From Genes to Plant Architecture: the Shoot Apical Meristem in all its States* (Poitiers, France: INRAE), 35–36.

Shu, Q., Rötzer, T., Detter, A., and Ludwig, F. (2022). Tree information modeling: A data exchange platform for tree design and management. *Forests* 13 (11), 1955. doi: 10.3390/f13111955

Simler, A. B., Metz, M. R., Frangioso, K. M., Meentemeyer, R. K., and Rizzo, D. M. (2018). Novel disturbance interactions between fire and an emerging disease impact survival and growth of resprouting trees. *Ecology* 99, 2217–2229. doi: 10.1002/ecy.2493

Singh, R. K., Bhalerao, R. P., and Maurya, J. P. (2022). When to branch: Seasonal control of shoot architecture in trees. *FEBS J.* 289, 8062–8070. doi: 10.1111/febs.16227

Sun, J., Wang, P., Li, R., Zhou, M., and Wu, Y. (2022). Fast tree skeleton extraction using voxel thinning based on tree point cloud. *Remote Sens.* 14 (11), 2558. doi: 10.3390/rs14112558

Suzuki, A. (2002). Influence of shoot architectural position on shoot growth and branching patterns in Cleyera japonica. *Tree Physiol.* 22, 885–889. doi: 10.1093/treephys/22.12.885

Taha, A. A., and Hanbury, A. (2015). An efficient algorithm for calculating the exact hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* 37 (11), 2153–2163. doi: 10.1109/TPAMI.2015.2408351

Teimoorinia, H., Toyonaga, R. D., Fabbro, S., and Bottrell, C. (2020). Comparison of multi-class and binary classification machine learning models in identifying strong gravitational lenses. *PASP.* 132, 044501. doi: 10.1088/1538-3873/ab747b

The MathWorks Inc. (2023). *MATLAB version: 9.13.0 (R2022b)* (The MathWorks Inc). Available at: https://www.mathworks.com.

Tosto, A., Evers, J. B., Anten, N. P. R., and Zuidema, P. A. (2023). Branching responses to pruning in young cocoa trees. *Scientia Horticulturae*. 322, 112439. doi: 10.1016/j.scienta.2023.112439

Wang, C., Li, C., Han, Q., Wu, F., and Zou, X. (2023). A performance analysis of a litchi picking robot system for actively removing obstructions, using an artificial intelligence algorithm. *Agronomy* 13 (11), 2795. doi: 10.3390/agronomy13112795

Wang, Z., Zhang, L., Fang, T., Mathiopoulos, P. T., Qu, H., Chen, D., et al. (2014). A structure-aware global optimization method for reconstructing 3-D tree models from terrestrial laser scanning data. *IEEE Trans. Geosci. Remote Sens.* 52, 5653–5669. doi: 10.1109/TGRS.2013.2291815

Yang, J., Wen, X., Wang, Q., Ye, J.-S., Zhang, Y., and Sun, Y. (2022). A novel algorithm based on geometric characteristics for tree branch skeleton extraction from LiDAR point cloud. *Forests* 13 (10), 1534. doi: 10.3390/f13101534

Yazdi, H., Shu, Q., and Ludwig, F. (2023). A target-driven tree planting and maintenance approach for next generation urban green infrastructure (UGI). *JoDLA – J. Digital Landscape Architecture* 8–2023, 178. doi: 10.14627/537740019

Yazdi, H., Shu, Q., Rötzer, T., Petzold, F., and Ludwig, F. (2024). A multilayered urban tree dataset of point clouds, quantitative structure and graph models. *Sci. Data* 11 (28). doi: 10.1038/s41597-023-02873-x

Ye, L., Wu, F., Zou, X., and Li, J. (2023). Path planning for mobile robots in unstructured orchard environments: An improved kinematically constrained bi-directional RRT approach. *Comput. Electron. Agric.* 215, 108453. doi: 10.1016/j.compag.2023.108453

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 11 March 2024
DOI 10.3389/fpls.2024.1342123

Check for updates

OPEN ACCESS

EDITED BY
Zhenghong Yu,
Guangdong Polytechnic of Science and Technology, China

REVIEWED BY
Milind B. Ratnaparkhe,
ICAR Indian Institute of Soybean Research, India
Honghua Jiang,
Shandong Agricultural University, China

*CORRESPONDENCE
Hanmi Zhou
zhouhm@163.com
Qi Wu
qiwu0701@syau.edu.cn

RECEIVED 21 November 2023
ACCEPTED 21 February 2024
PUBLISHED 11 March 2024

CITATION
Zhou H, Chen J, Niu X, Dai Z, Qin L, Ma L, Li J, Su Y and Wu Q (2024) Identification of leaf diseases in field crops based on improved ShuffleNetV2.
Front. Plant Sci. 15:1342123.
doi: 10.3389/fpls.2024.1342123

COPYRIGHT
© 2024 Zhou, Chen, Niu, Dai, Qin, Ma, Li, Su and Wu. 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.

Identification of leaf diseases in field crops based on improved ShuffleNetV2

Hanmi Zhou[1*], Jiageng Chen[1], Xiaoli Niu[1], Zhiguang Dai[1], Long Qin[1], Linshuang Ma[1], Jichen Li[1], Yumin Su[1] and Qi Wu[2*]

[1]College of Agricultural Engineering, Henan University of Science and Technology, Luoyang, China, [2]College of Water Resource, Shenyang Agricultural University, Shenyang, China

Rapid and accurate identification and timely protection of crop disease is of great importance for ensuring crop yields. Aiming at the problems of large model parameters of existing crop disease recognition methods and low recognition accuracy in the complex background of the field, we propose a lightweight crop leaf disease recognition model based on improved ShuffleNetV2. First, the repetition number and the number of output channels of the basic module of the ShuffleNetV2 model are redesigned to reduce the model parameters to make the model more lightweight while ensuring the accuracy of the model. Second, the residual structure is introduced in the basic feature extraction module to solve the gradient vanishing problem and enable the model to learn more complex feature representations. Then, parallel paths were added to the mechanism of the efficient channel attention (ECA) module, and the weights of different paths were adaptively updated by learnable parameters, and then the efficient dual channel attention (EDCA) module was proposed, which was embedded into the ShuffleNetV2 to improve the cross-channel interaction capability of the model. Finally, a multi-scale shallow feature extraction module and a multi-scale deep feature extraction module were introduced to improve the model's ability to extract lesions at different scales. Based on the above improvements, a lightweight crop leaf disease recognition model REM-ShuffleNetV2 was proposed. Experiments results show that the accuracy and F1 score of the REM-ShuffleNetV2 model on the self-constructed field crop leaf disease dataset are 96.72% and 96.62%, which are 3.88% and 4.37% higher than that of the ShuffleNetV2 model; and the number of model parameters is 4.40M, which is 9.65% less than that of the original model. Compared with classic networks such as DenseNet121, EfficientNet, and MobileNetV3, the REM-ShuffleNetV2 model not only has higher recognition accuracy but also has fewer model parameters. The REM-ShuffleNetV2 model proposed in this study can achieve accurate identification of crop leaf disease in complex field backgrounds, and the model is small, which is convenient to deploy to the mobile end, and provides a reference for intelligent diagnosis of crop leaf disease.

KEYWORDS

complex background, crop leaf disease, ShuffleNetV2, EDCA module, residual structure

1 Introduction

Various diseases in the process of crop growth will significantly reduce the yield and quality of agricultural products and seriously restrict agricultural production. To improve agricultural production efficiency, timely detection and early prevention of crop diseases are crucial (Hassan et al., 2021; Wang and Wang, 2021). At present, crop disease identification mainly relies on manual diagnosis, however the wide variety of crop diseases and the similarity of symptoms of some of them lead to a time-consuming and laborious diagnostic process (Barbedo, 2016). Image processing and machine vision can adapt to complex and changeable natural scenes and lay the foundation for accurate identification and diagnosis of crop disease (Zhang et al., 2014; Hossain et al., 2021; Ye et al., 2021). Therefore, computer vision and image processing strategies are utilized to design an intelligent recognition algorithm that can diagnose crop diseases quickly, inexpensively, and accurately, which is of great practical significance for the establishment of disease prediction mechanisms for timely prevention and control.

Since the 1980s, researchers have started to identify crop diseases using machine learning and image processing methods, proposing many traditional methods for image recognition of crop diseases (Camargo and Smith, 2009; Ma et al., 2017; Zhang et al., 2020). Tian et al. (2016) proposed a recognition method for eggplant brown streak disease based on spot characteristics, using the H component of the HSI color space to extract the feature parameters of the spot area and selecting the feature parameters to form a classification feature vector for classification by principal component analysis, which achieved better experimental results. Zhang and Zhang (2014) used region growing segmentation algorithm to segment disease spot images in diseased maize leaves and reorganized them into one-dimensional vectors, and used a nearest neighbor classifier to identify the disease categories with good recognition results. These traditional methods require manual design of features such as color, texture, and edge gradient of disease images for recognition. However, manually designed features require expensive resource conditions and specialized knowledge and are susceptible to subjectivity. In addition, the inability to efficiently segment leaves and corresponding disease images under complex background conditions has led to the inability of these methods to meet the needs of modern agriculture for accurate identification of crop disease.

In recent years, with the rapid development of deep learning techniques and the enhancement of computer processing power, crop leaf disease recognition methods based on convolutional neural networks (CNNs) have become a research focus of many researchers (Huang et al., 2021; Bao et al., 2022; Du et al., 2023; Praveen et al., 2023). Sun et al. (2021) embedded the coordinate attention mechanism in the MobileNetV2 model, and then performed fusion and extraction operations on feature maps of different sizes. The recognition accuracy of the improved model for a variety of crop leaf diseases was 92.20%. Rangarajan et al. (2018) used the strategy of fine-tuning and transfer learning for AlexNet and VGG16 to propose two fast converging models, which obtained 97.29% and 97.49% recognition rates on the tomato dataset. Gao et al. (2023) proposed an Apple Leaf Disease Recognition Model (BAM-Net) that uses an aggregated coordinate attention mechanism to enhance the network's focus on disease features, introduces a multi-scale feature refinement module to improve the network's ability to discriminate between similar disease features, which achieved an accuracy of 95.64% on the test set. Peng et al. (2022) introduced the SimAM module on the ShuffleNetv2 model to enhance the effective extraction of important features and used the activation function Hardswish to reduce the number of network model parameters, which resulted in a recognition accuracy of 84.9% on lychee pests and diseases. Bhagat et al. (2023) introduced local binary pattern for feature fusion based on the VGG-16 model and used random forest method for classification, which effectively improved the robustness of the model and achieved an accuracy of 99.75% on the sweet pepper leaf dataset. Agarwal et al. (2020) proposed a simplified convolutional neural network model that was tested on the tomato leaf dataset and the experimental results showed that the proposed model has better results than traditional machine learning methods. The above studies have proved the feasibility of CNNs in crop leaf disease recognition, but there are also problems such as a large number of network parameters, a large amount of calculation, and complex model, which make the model difficult to carry and move.

To solve the problem of mobile deployment of deep learning models, some researchers have proposed methods such as knowledge distillation and model pruning, aiming to improve the performance of network models and reduce the number of model parameters. Peng and Li (2023) proposed a plant leaf disease recognition model RLDNet based on improved MobileNetV2. The model used the reparameterized inverted residual module to improve the inference speed. The DepthShrinker pruning method is used to reduce the space occupation. The recognition accuracy of the RLDNet model on the PlantVillage dataset under simple background is 99.53%, and the number of parameters is 0.65 M. Liu et al. (2023) used the ResNet model as the baseline model, introduced a multi-teacher joint distillation strategy to train the model, and utilized model pruning to reduce the number of model parameters. After pruning the model by 90%, the model achieved up to 97.78% accuracy on the PlantVillage dataset, while after pruning the model by 70%, the model achieved up to 91.94% accuracy on the Apple Leaf Disease dataset in a complex context. Wen et al. (2023) used ShuffleNetV2 as the base network, introduced the efficient channel attention mechanism with the silu activation function for structural improvement, and also combined the knowledge distillation technique to train the model. The improved model achieved 95.21% accuracy in recognizing 11 diseases of two crops in a complex environment. However, although the above methods make the crop leaf disease recognition model lightweight, the effect of disease recognition in real scenes needs to be improved.

Based on the above problems, this study constructed a variety of crop disease datasets contained in the field context, and then used ShuffleNetV2-1.0 network as the baseline model, fine-tuned the model parameters, and introduced the efficient dual channel attention (EDCA) module, the multi-scale feature fusion module, and residual structure connection strategy. We propose a field crop leaf disease recognition model-REM-ShuffleNetV2 based on

improved ShuffleNetV2. This model can effectively extract the subtle features of crop leaf diseases and improve the accuracy of crop disease classification in the field. Meanwhile, the model has the advantages of small size and few parameters, which can provide a reference for subsequent related research. The main innovations of this paper are as follows:

1. A lightweight CNN model REM-ShuffleNetV2 is proposed for the automatic identification of leaf diseases in field crops on mobile devices.
2. The number of repetitions and output channels of the basic module of the ShuffleNetV2 model are fine-tuned to reduce the model parameters and make the model lightweight.
3. The EDCA module is embedded in the basic feature extraction module, which enhances the model's ability to extract effective feature information in crop disease, and introduces residual structure to alleviate the problem of information loss and gradient loss in the model.
4. A multi-scale shallow feature extraction module and a multi-scale deep feature extraction module are designed to enable the model to capture feature information at different scales, thus improving the model's perceptual and expressive capabilities.

2 Datasets

2.1 Data acquisition

In this study, the dataset used contains 17 categories of diseased leaf images of six crops (apple, soybean, maize, strawberry, sugarcane, and wheat) and healthy leaf images of five crops (apple, soybean, maize, strawberry, and sugarcane), totaling 22 categories and 8,408 sample images from the field collection, the official website of Kaggle(https://www.kaggle.com/), and the website of Baidu Fly Paddle(https://aistudio.baidu.com/), and the sample images were all taken in a field background Photographed (Muhab and Ercan, 2022). Disease types include apple alternaria leaf spot, bean angular leaf spot, maize northern leaf blight, strawberry calciumdeficieny, sugarcane red rot, wheat powdery mildew, etc. Some sample images are shown in Figure 1.

2.2 Data set segmentation and preprocessing

The original dataset is randomly divided into a training set and a test set in a ratio of 8:2 (Liu and Cui, 2023), where the training set has 6732 images and the test set has 1676 images. To increase the diversity of crop disease datasets, and enhance the generalization ability and robustness of the model, this study performs data enhancement on the training set (Shorten and Khoshgoftaar, 2019). Data enhancement follows the principle of increasing the number of samples while keeping the sample features unchanged to better reflect the real background. In this study, two image enhancement techniques were used: 1) Brightness enhancement and attenuation: used to simulate different lighting conditions in real field background; 2) Rotation and flip: used to simulate the shooting of the recognition device at different angles. Finally, a sufficient and balanced training set with 22217 images is obtained by the augmentation technique. Detailed sample information is shown in Table 1.

2.3 The process of disease identification

The overall process of crop leaf disease identification is shown in Figure 2. Firstly, the disease image data were collected through multiple channels and the useless images were manually removed. Secondly, the constructed dataset is preprocessed and divided into

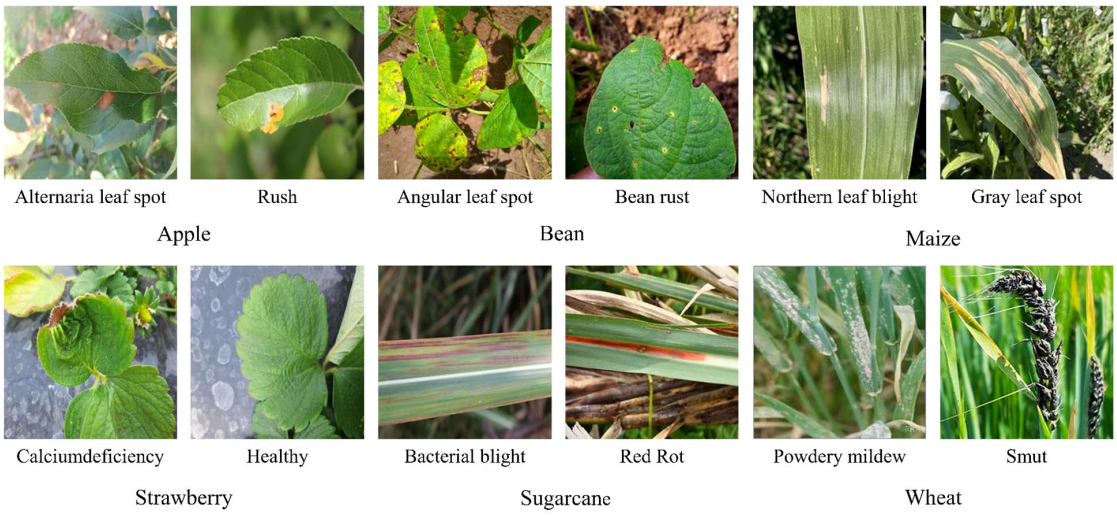


FIGURE 1
Diseased images of crops in a field background.

TABLE 1 Detailed sample information on the dataset.

Crops	Types of diseases	Number	Before	After	Test set
Apple	Alternaria leaf spot	A1	219	1095	54
	Grey spot	A2	131	1048	32
	Health	A3	516	1032	129
	Mosaic	A4	137	1096	34
	Powdery mildew	A5	549	1098	137
	Rust	A6	447	894	111
	Scab	A7	477	954	119
Bean	Angular leaf spot	B1	264	1056	66
	Bean rust	B2	264	1056	66
	Healthy	B3	264	1056	66
Maize	Gray Leaf Spot	M1	398	1194	99
	Health	M2	265	1060	66
	Northern leaf blight	M3	419	1257	104
	Northern leaf spot	M4	441	882	110
Strawberry	Calciumdeficieny	S1	378	1134	94
	Healthy	S2	369	1107	92
Sugarcane	Bacterial blight	SU1	80	640	20
	Healthy	SU2	80	640	20
	Red rot	SU3	80	640	20
Wheat	Powdery mildew	W1	208	1040	51
	Smut	W2	330	990	82
	Rust	W3	416	1248	104
Total	–	–	6732	22217	1676

"Before" represents the original training set; "After" represents the augmented training set.

training and testing sets in 8:2 ratio, and the original dataset is expanded by data enhancement to increase the diversity to improve the generalization ability of the trained model. Finally, the data-enhanced dataset is used to train the REM-ShuffleNetV2 model and the model weights with the best performance during training are saved. Based on the above trained REM-ShuffleNetV2 model, the images in the test set are used to get the prediction categories of the test samples for crop disease recognition on leaves. If more disease image data is subsequently collected, all can follow this process to retrain the model to improve the performance.

3 Crop leaf disease recognition model

3.1 ShuffleNetv2 model

With the rapid development of convolutional neural networks in the field of computer vision, although the traditional convolutional neural networks have good accuracy, their large number of model parameters is difficult to adapt to today's mobile devices with limited computational resources (Liu et al., 2017). ShuffleNetV2 is an extremely efficient lightweight convolutional neural network for mobile devices proposed by Ma et al. (2018). The network introduces the concept of group convolution which divides the input and output channels into multiple groups and performs convolution operations within each group. This design enables the network to parallelize processing efficiently and significantly reduce the computational cost. By rearranging the feature channels, information from different channels can be mixed and exchanged, leading to better representation learning and reducing the overall complexity of the network. The basic feature extraction module of ShuffleNetV2 is shown in Figure 3.

3.2 EDCA module

The crop disease samples in the dataset constructed in this study were taken in a field environment with complex background information. The attention mechanism adjusts the weight of the

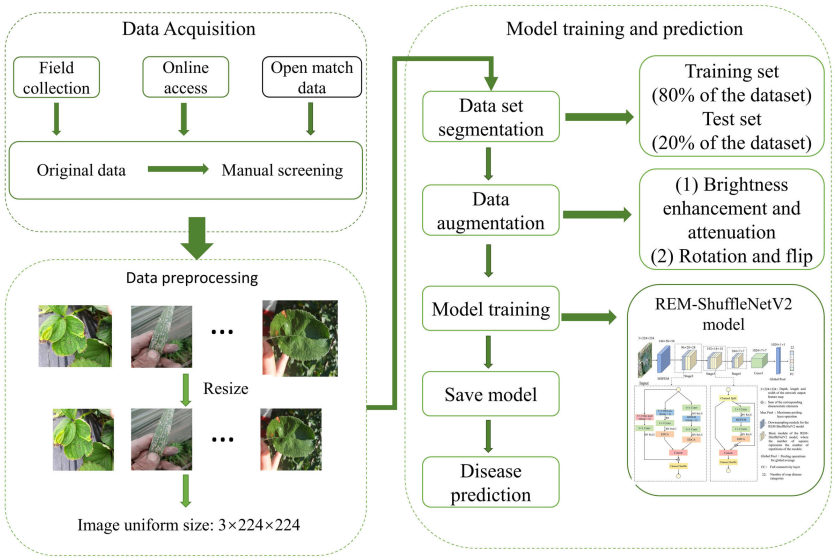


FIGURE 2
The overall process of disease identification.

input feature map to suppress redundant background information and enhance the feature representation of the foreground disease in the image, thereby improving the recognition performance of the model (Huang et al., 2023). SE (Squeeze and Excitation) module uses global average pooling to aggregate global information, and then captures nonlinear cross-channel interactions by compressing channels for dimensionality reduction, but this approach is not conducive to learning inter-channel dependencies (Glorot et al., 2011). The ECA (Efficient Channel Attention) module uses one-dimensional convolution to realize cross-channel interactions and

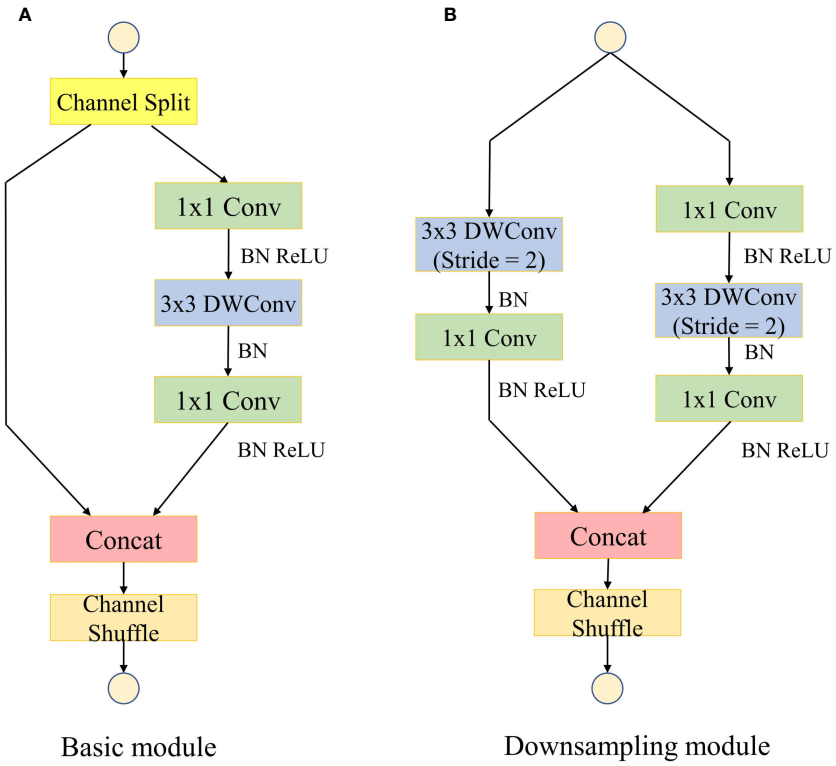


FIGURE 3
Basic feature extraction module for the ShuffleNetV2 model. "Conv" represents standard convolution; "BN" represents batch normalization; "ReLU" represents activation function; "Concat" represents channel splicing. **(A)** Basic module. **(B)** Downsampling module.

learns inter-channel dependencies while keeping the channel dimensions unchanged, and the model requires fewer parameters and less computation to introduce the ECA module compared to the SE module (Wang et al., 2020). To further optimize the global information extraction capability of the ECA module, inspired by the SRM (Style-based Recalibration Module) module (Lee et al., 2019), this study proposes an EDCA (Efficient Dual Channel Attention) module, and its structure is shown in Figure 4.

Suppose X is the input feature, and the size of the feature map is H×W×C, where H represents the height of the feature map, W represents the width of the feature map, and C represents the number of channels of the feature map. The EDCA module processes the input using average pooling (AvgPool) and standard deviation pooling (StdPool) to compress it into 1×1×C feature maps, respectively, and generates weights for each channel by one-dimensional convolution of size K. The average value and standard deviation are calculated as shown in (Equations 1, 2):

$$A_c = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}X_c(h,w) \tag{1}$$

$$S_c = \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(X_c(h,w)-A_c)^2} \tag{2}$$

In (Equations 1, 2), Ac and Sc represent the average value and standard deviation of each element in the channel.

The convolution kernel size K can be adaptively determined by nonlinear mapping of the channel dimensions, and the adaptation function is defined as shown in (Equation 3):

$$K = \psi(C) = \left|\frac{log_2(C)}{\gamma} + \frac{b}{\gamma}\right|_{odd} \tag{3}$$

In (Equation 3), C represents the input feature channel dimensions, |x|odd represents the closest singularity to x, γ and b are used to change the ratio between the number of channels C and the convolution kernel size, and are taken to be γ = 2 and b = 1 according to empirical values taken from the literature. Then, the elements of the feature maps obtained by the two paths are added together, and the weight ω of each channel is obtained by the Sigmoid function. At last, the weights ω are multiplied by the original input feature map. The calculation of the weights ω is shown in (Equation 4):

$$\omega = \sigma(C1D_k(y_1) + C1D_k(y_2)) \tag{4}$$

In (Equation 4), represents the Sigmoid activation function, C1D represents the one-dimensional convolution, K represents the one-dimensional convolution kernel size, y1 represents the feature map output by the average pooling path, y2 represents the feature map output by the standard deviation pooling path.

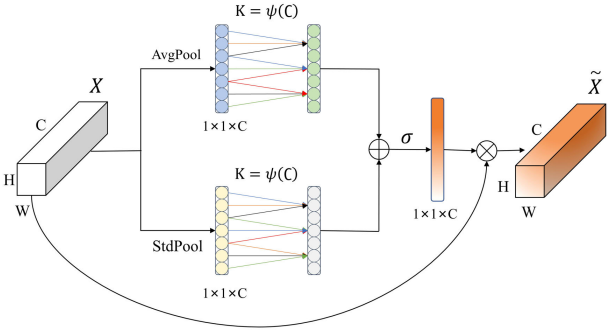


FIGURE 4
EDCA Module.

3.3 Multi-scale feature extraction module

In the convolutional neural networks, the low-level convolutions mainly extract simple features such as color, texture, and edge of images, which usually have strong expressive power in local regions of images Yang et al., 2022), while the features extracted by high-level convolutions are abstract, global, and have global expressive power (Li et al., 2020). In the ShuffleNetV2 model, a 3×3 convolutional layer and a maximum pooling layer are used to extract low-level convolutional features. However, this method extracts fewer features, and the receptive field is fixed. This leads to the fact that low-level convolution cannot adequately capture the subtle feature differences of different size spots in crop leaf diseases (Shah et al., 2017). Therefore, this study designed a multi-scale shallow feature extraction module (Figure 5A) with a combination of a maximum pooling layer and multiple 3×3 convolutional layers to improve the response of the shallow network to features of different granularity. Meanwhile, a multi-scale deep feature extraction module (Figure 5B) with the combination of 3×3 convolutional layers and 5×5 convolutional layers was designed to further improve the global feature extraction capability of the model.

3.4 Crop leaf disease recognition model REM-ShuffleNetV2

ShuffleNetV2 model adopts lightweight design strategies such as depthwise convolution, channel random rearrangement, etc., which has less parameters and computation. However, the early lesions of crop leaf diseases are sparsely distributed, and the lesions tend to exhibit small area, inconspicuous features, and different morphologies, resulting in a lower overall recognition accuracy of the ShuffleNetV2 model. To further improve the accuracy of the model, this study optimized the ShuffleNetV2 model by first changing the number of repetitions of the basic modules in the Stage2, Stage3, and Stage4 phases of the model to [2, 3, 2], and fine-tuning the number of output channels to reduce the number of parameters in the model. Then, the residual structure is introduced into the basic feature extraction module of the ShuffleNetV2 model. The residual structure can increase the network learning path while preserving the original features, so that the network can pass the shallow information directly to the deep layer, solve the problem of gradient disappearance or gradient explosion that occurs in the process of model training, thereby improving the expression ability of the model (Le et al., 2023). In the residual structure of the downsampling module, the maximum pooling layer was used to

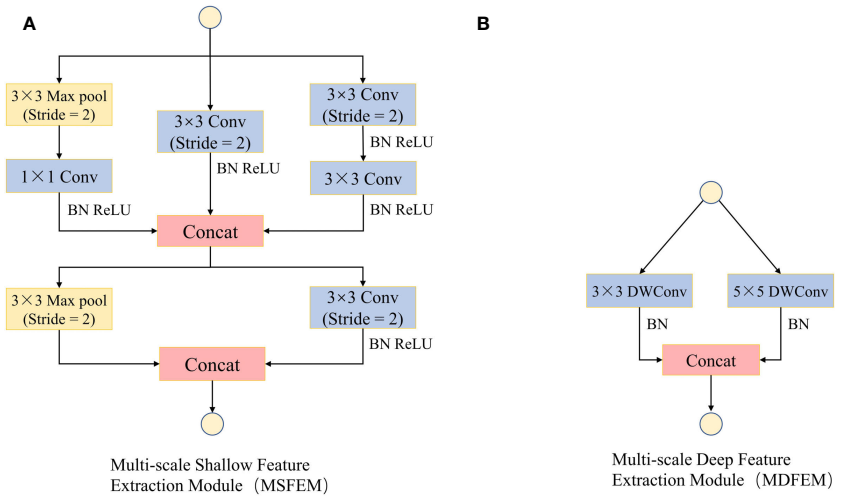


FIGURE 5
Multi-scale feature extraction module (MFEM). **(A)** Multi-scale Shallow Feature Extraction Module (MSFEM). **(B)** Multi-scale Deep Feature Extraction Module (MDFEM).

complete the downsampling, and the number of channels was adjusted by 1×1 convolution, to ensure that the output number of channels was consistent. The EDCA module is introduced after the pointwise convolution at the tail of the basic feature extraction module, so that the model can pay targeted attention to the disease spot features in the input data, to improve the model's ability to extract effective feature information. Finally, a multi-scale feature extraction module is introduced to enhance the model's ability to extract shallow semantic information and deep semantic information. Combining the above improvement approaches, this study proposes the high-precision and low-consumption network model REM-ShuffleNetV2, as shown in Figure 6.

4 Results and analysis

4.1 Experimental environment setup

The experiments were conducted using a desktop computer as the processing platform, the operating system was Windows 10, and the Pytorch framework was used, the experimental environment was constructed in the Anaconda3 software, and the program was written in Python 3.8, the CUDA version was 11.1, and the Torch version was 1.8.0. Hardware: The processor is Intel Pentium G4560, the running memory is 16G, the graphics card is NVIDIA GeForce RTX3050, and the video memory is 8G.

Considering the hardware performance of the equipment and the training effect, the batch training method was used to divide the training and testing process into multiple batches, each batch contained 32 images, and the number of iterations was set to 60. The loss function uses cross-entropy loss and the classification layer uses Softmax function. The model was trained using an SGD optimizer with a momentum parameter of 0.9 and a weight decay parameter of 0.0005. The initial learning rate was 0.01, which was tuned using a cosine annealing decay strategy, with a total number of steps in a cycle of 60, and a lower value of 1e-9 for the learning rate.

4.2 Evaluation metrics

To evaluate the performance of the REM-ShuffleNetV2 network, this paper uses model size and number of parameters as the evaluation criteria for model complexity, and precision P, recall R, F1 score, and accuracy A on the test set as the evaluation indexes for model performance. The above four performance indicators are calculated as shown in Equations 5–8.

$$P = \frac{TP}{TP + FP} \tag{5}$$

$$R = \frac{TP}{TP + FN} \tag{6}$$

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$F_1 = \frac{2 * P * R}{P + R} \tag{8}$$

Where TP is the result of correctly predicting positive classification; FP is the result of incorrectly prediction of positive classification; TN is the result of correctly predicting negative classification; FN is the result of incorrectly predicting negative classification.

4.3 ShuffleNetV2-1.0 model parameter tuning

To obtain the optimal parameters of the ShuffleNetV2-1.0 model, this study adjusted the number of basic modules and the

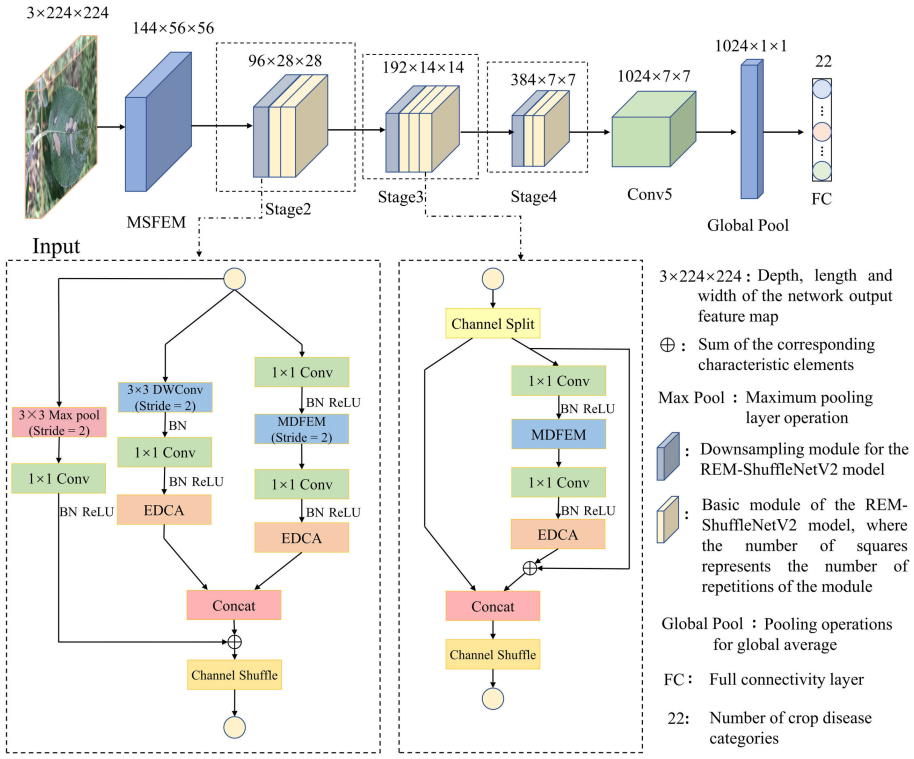


FIGURE 6
Overall model structure diagram.

number of output channels in the Stage2, Stage3, and Stage4 phases, designed five different parameters and conducted experiments, and the experimental results are shown in Table 2. Under the condition of the constant number of output channels, the best training results of the model are obtained when the number of basic modules in Stage2, Stage3, and Stage4 is [2, 3, 2], and based on this, the best model recognition results with the accuracy of 93.50% were obtained when the number of output channels of the model was [116, 232, 464, 1024]. However, with the number of output channels set to [96, 192, 384, 1024], the accuracy of the model was only 0.18% lower than the best case, but the size of the model was reduced by 22%. To consider the accuracy rate and model size, this study sets the number of basic modules in Stage2, Stage3, and Stage4 to [2, 3, 2], and the number of output channels to [96, 192, 384, 1024], and under this parameter, the accuracy rate of the model was improved by 0.48% compared with that of the original model, and the size of the model was reduced by 1.89MB. The next optimization experiments were carried out under this parameter.

4.4 Effects of different down sampling methods in residual structure on model performance

To study the effect of different down sampling methods in the residual structure of the down sampling module on the performance of the model, this study conducted comparative experiments using the completed down sampling methods of the maximally pooled layer (RM), the average pooled layer (RA), and the 3 × 3 convolutional layer (RC). The results are shown in Table 3,

TABLE 2 ShuffleNetV2-1.0 Parameter Tuning.

Number	Repeat	Output channels	Accuracy/%	Model size/MB
0	[3, 7, 3]	[116, 232, 464, 1024]	92.84	5.03
1	[3, 3, 3]	[116, 232, 464, 1024]	93.20	4.56
2	[2, 3, 2]	[116, 232, 464, 1024]	93.50	4.12
3	[1, 3, 1]	[116, 232, 464, 1024]	93.08	3.63
4	[2, 3, 2]	[96, 192, 384, 1024]	93.32	3.14
5	[2, 3, 2]	[96, 192, 384, 768]	92.96	2.74

TABLE 3 Experimental results for different downsampling methods in residual structure.

Model	Precision /%	Recall /%	F1 score /%	Accuracy /%	Parameters /M	Model size/MB
ShuffleNetV2	92.27	92.42	92.53	93.32	3.03	3.14
ShuffleNetV2-RM	95.00	94.31	94.55	94.99	3.39	5.17
ShuffleNetV2-RA	93.62	92.97	93.14	93.74	3.39	5.17
ShuffleNetV2-RC	92.12	91.55	91.71	92.84	6.28	20.99

using RM and RA to complete downsampling in residual structure improves the performance of the model, this is because the pooling layer retains the main feature information of the image while completing downsampling (Saeedan et al., 2018). Among them, the best results achieved by using RM to complete the downsampling, the F1 score and accuracy of the model increased by 2.02% and 1.67% compared with the original model, this is mainly because RM, by selecting the maximum value, can select the feature activation value with the strongest response and discard the other weaker responses, realizing the downsampling of retaining the important information (He et al., 2022). The use of RC to accomplish downsampling was the least effective, with the number of parameters and model size increasing by 3.25M and 17.84MB, and the F1 score and accuracy decreasing by 0.82% and 0.48%.

4.5 Effects of different attention mechanisms on model performance

To verify the effectiveness of the EDCA module proposed in this study, comparative experiments are conducted with the SE module, the original ECA module, and the SRM module, respectively. Table 4 shows that compared to the ShuffleNetV2 model, the model recognition accuracies with the introduction of the SE module, ECA module, SRM module, and EDCA module increased by 0.30%, 0.54%, 0.71%, and 0.89%, respectively; and the F1 scores increased by 0.19%, 0.17%, 0.44%, and 0.68%, suggesting that the introduction of the attention mechanism helps in the recognition of leaf diseases in crops. Meanwhile, the introduction of the EDCA module compared to the original ECA module improved the F1 score and accuracy by 0.51% and 0.35%, respectively. In addition, compared with other attention mechanism modules, the EDCA module achieves the optimal recognition effect with the number of parameters and model size basically unchanged.

Heatmap can intuitively show whether the network learns the key features or not through the degree of color change, this paper visualizes the feature map after the introduction of the attention mechanism in the ShuffleNetV2 model in the form of a heatmap (Figure 7), in which the more the color tends to be in deep red, indicating that the model is more responsive in that region. As is shown in Figure 7, compared with the ShuffleNetV2 model, the model incorporating the attention mechanism can better notice the feature regions related to crop disease leaves and has a stronger ability to recognize the feature information of the crop disease. Meanwhile, the introduction of the EDCA module can extract the feature information of the diseased area more accurately than other attention mechanisms, effectively avoiding the interference of non-important features such as the background environment, which further proves the effectiveness of the EDCA module.

4.6 Effect of MSFEM module on model performance

To better extract shallow feature information, different network models are designed with different network structures. As shown in Figure 8, Stem-A is the shallow feature extraction module of the ShuffleNetV2 model, which consists of a 3×3 convolutional layer with a step size of 2 and a 3×3 maximum pooling with a step size of 2. Stem-B is the shallow feature extraction module of the ResNet model, which consists of a 7×7 convolutional layer with a step size of 2 and a 3×3 maximum pooling with a step size of 2. Stem-C is the shallow feature extraction module of the Inception-ResnetV2 model, which uses a stack of three 3×3 convolutional layers instead of 7×7 convolutional layers, and combines the 3×3 convolutional layers with the maximum pooling layer through a branch structure (Szegedy et al., 2017). To verify the effectiveness of the MSFEM module, a comparison experiment was conducted and

TABLE 4 Experimental results of introducing different attention mechanisms into the model.

Model	F1 score/%	Accuracy/%	Parameters/M	Model size/MB
ShuffleNetV2	94.55	94.99	3.39	5.17
ShuffleNetV2+SE	94.74	95.29	3.49	5.28
ShuffleNetV2+ECA	94.72	95.53	3.39	5.18
ShuffleNetV2+SRM	94.99	95.70	3.41	5.23
ShuffleNetV2+EDCA	95.23	95.88	3.39	5.18

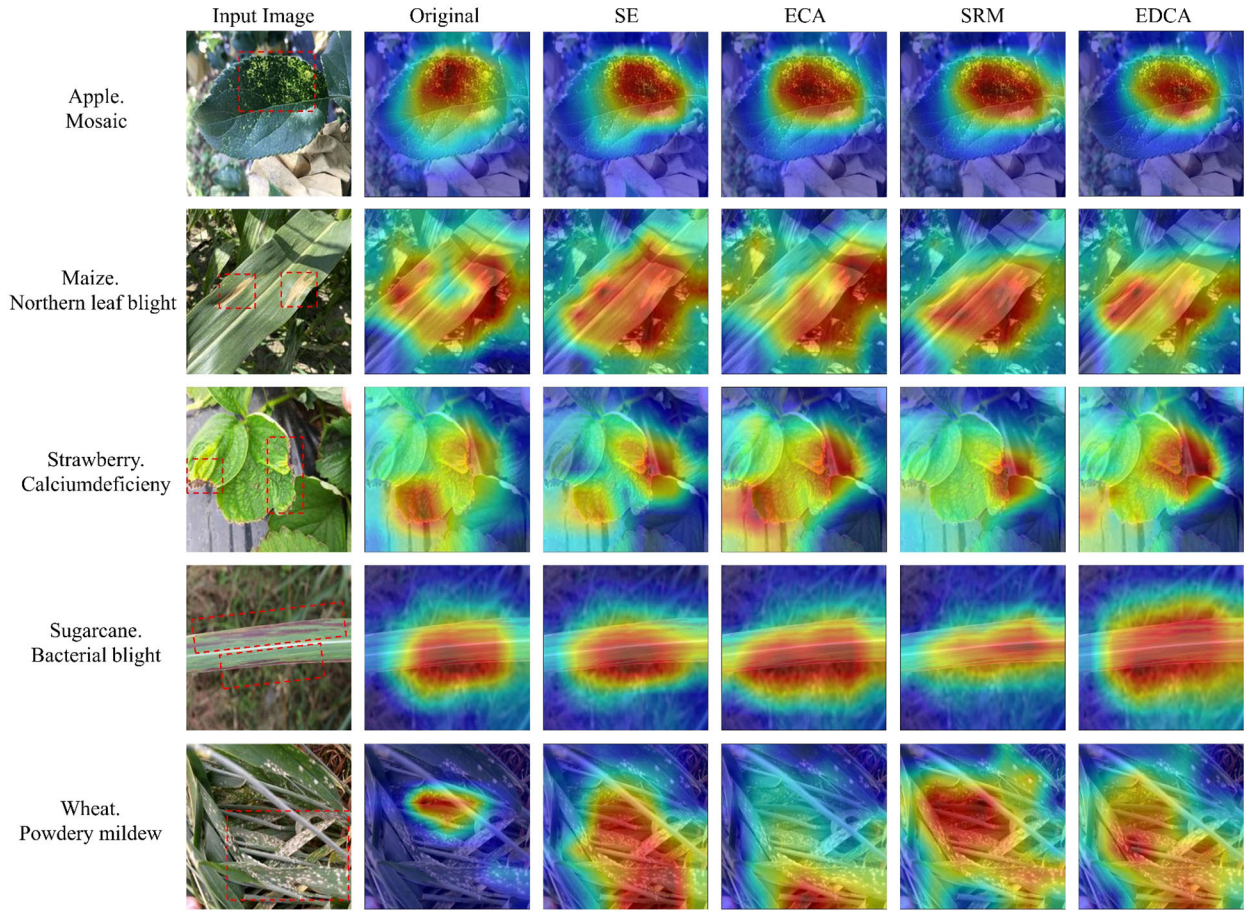


FIGURE 7
Comparison of the heatmap for different attention mechanisms. The red boxes in the original image indicate the main areas of disease in the crop.

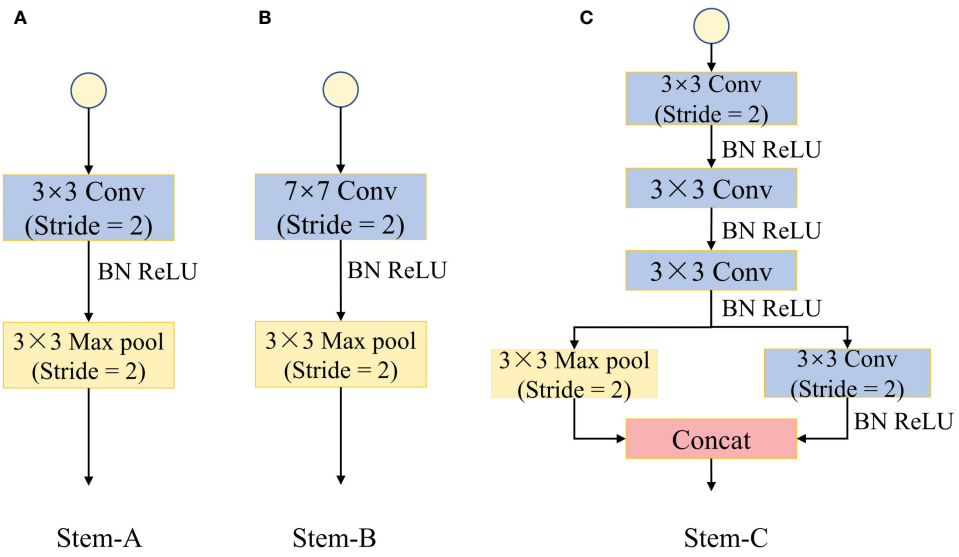


FIGURE 8
Stem module structure. **(A)** Stem-A. **(B)** Stem-B. **(C)** Stem-C.

TABLE 5 Comparison of experimental results of different shallow feature extraction modules.

Model	F1 score/%	Accuracy/%	Parameters/M	Model size/MB
ShuffleNetV2-Stem-A	92.53	93.32	3.03	3.14
ShuffleNetV2-Stem-B	91.87	92.94	3.04	3.15
ShuffleNetV2-Stem-C	93.39	93.85	3.10	3.22
ShuffleNetV2-MSFEM	93.54	94.09	3.28	3.40

the results are shown in Table 5. The introduced Stem-B module has basically the same number of parameters and model size compared to the original model (ShuffleNetV2-Stem-A), but the F1 score and accuracy are reduced by 0.66% and 0.38%. The introduction of the Stem-C module increased the model's nonlinear capability and receptive fields, and the model's F1 score and accuracy improved by 0.86% and 0.53%, respectively. The introduction of the MSFEM module increased the number of parameters and model size by 0.25M and 0.26MB, but the F1 score and accuracy improved by 1.01% and 0.77%, respectively. Taken together, the test with the introduction of the MSFEM module was the most effective.

4.7 Ablation experiment with the REM-ShuffleNetV2 model

To explore the performance enhancement of the ShuffleNetV2 model brought about by the improved approach of using architecture tuning, residual structure connection, EDCA module, and Multiscale Feature Fusion Module (MFEM), ablation experiments are conducted and the results are shown in Table 6. After data enhancement, the F1 score and accuracy of the ShuffleNetV2-1.0 model improved by 6.83% and 5.01%, respectively, without increased model parameters. After parameter tuning, the F1 score of the model was improved by 0.28% and the accuracy by 0.48%, while the number of parameters and model size were reduced by 1.84M and 1.89MB. When the residual structure method is introduced into the basic feature extraction module of the model, the F1 score and accuracy of the model increased by 2.02% and 1.67%, respectively, but the number of parameters and model size increased by 0.39M and 2.03MB. The introduction of the EDCA module improves the F1 score and accuracy of the model by 0.68% and 0.89% while keeping the number of parameters constant. With the introduction of the multi-scale feature fusion module, the F1 score and accuracy of the model increased by 1.39% and 0.84%, while the number of parameters and model size increased by 1.01M and 1.05MB, respectively. Finally, the F1 score and accuracy of the REM-ShuffleNetV2 lightweight model proposed in this study were 96.62% and 96.72%, which were 4.37% and 3.86% higher than the original model, the number of covariates was 4.40M which was 0.47M less than that of the original model, and the size of the model was 6.23MB which was 1.20MB more than that of the original model.

To observe the variation of performance metrics of ShuffleNetV2 model and REM-ShuffleNetV2 model on different crop diseases, the precision P, recall R, F1 score, and accuracy A of the models were visualized for different crops. As shown in Figure 9, the recognition effect of the ShuffleNetV2 model on apple disease, soybean disease, and wheat disease was poor, this is because they have more types of diseases and high similarity of lesion characteristics, which leads to recognition difficulties. The recognition effect of ShuffleNetV2 model on maize disease, strawberry disease, and sugarcane disease was better, this is because they are easy to differentiate due to their fewer types of diseases and distinct disease characteristics. REM-ShuffleNetV2 improved crop disease recognition to varying degrees. On the more difficult to recognize apple, bean, and wheat diseases, the average F1 score and average accuracy improved by 7.52% and

TABLE 6 Ablation experiment with the REM-ShuffleNetV2 model.

Model	Data enhancement	Parameter tuning	Residual structure	EDCA module	Multi-scale feature fusion module	F1 score/ %	Accuracy /%	Parameters /M	Model size/ MB
ShuffleNetV2	–	–	–	–	–	85.42	87.83	4.87	5.03
	√	-	–	–	–	92.25	92.84	4.87	5.03
	√	√	–	–	–	92.53	93.32	3.03	3.14
	√	√	√		–	94.55	94.99	3.39	5.17
	√	√	√	√	–	95.23	95.88	3.39	5.17
REM-ShuffleNetV2	√	√	√	√	√	96.62	96.72	4.40	6.23

"-" means not to use the improvement factor, "√" means to use the improvement factor.

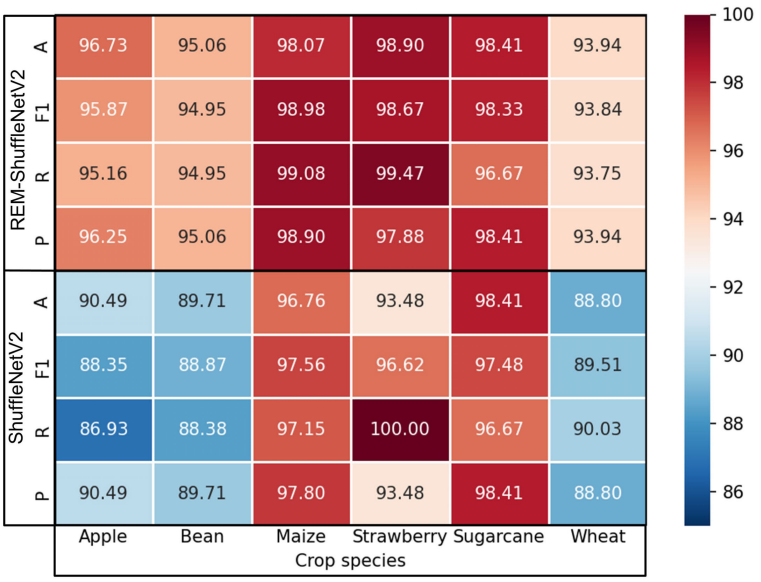


FIGURE 9
Performance metrics of the model before and after improvement on individual crops.

6.24%, 6.08% and 5.35%, 4.33% and 5.14%, respectively, compared to the original model. For the easily recognized maize disease and strawberry disease, the average F1 score and average accuracy improved by 1.42% and 1.31%, 2.05% and 5.42%, respectively, compared with the original model. For sugarcane diseases, the average precision, average recall, and average accuracy of the REM-ShuffleNetV2 model were the same as those of the original model, but the average F1 score was improved by 0.85%.

4.8 Different network comparison experiments

To further verify the effectiveness of the REM-ShuffleNetV2 model, this paper compared it with the DenseNet121 (Huang et al., 2017), EfficientNet (Tan and Le, 2019), MobileNetV3 (Howard et al., 2019), MobileVit (Mehta and Rastegari, 2021) and RepVGG (Ding et al., 2021) models under the same test conditions. The change curves of accuracy and loss value of different models are shown in Figure 10.

As can be seen in Figure 10A, after 60 iterations, the accuracy of each model in crop disease tends to stabilize, which indicates that the performance of the model has been fully demonstrated. REM-ShuffleNetV2 is the fastest converging model among these models. When iterating to the 5th round, the accuracy of the REM-ShuffleNetV2 model had already reached 90%. As the iteration proceeds, the accuracy of the model reaches 96% at round 20 and begins to converge. In contrast, the training curves of the remaining models behave similarly. After 15 rounds of iterations, these models all achieve 90% accuracy and begin to converge after 30 rounds. In the later stages of training, the REM-ShuffleNetV2 model exhibits higher accuracy with less fluctuation. This shows that REM-ShuffleNetV2 had stronger robustness and faster convergence on the crop leaf disease test set. Figure 10B shows that the loss value of

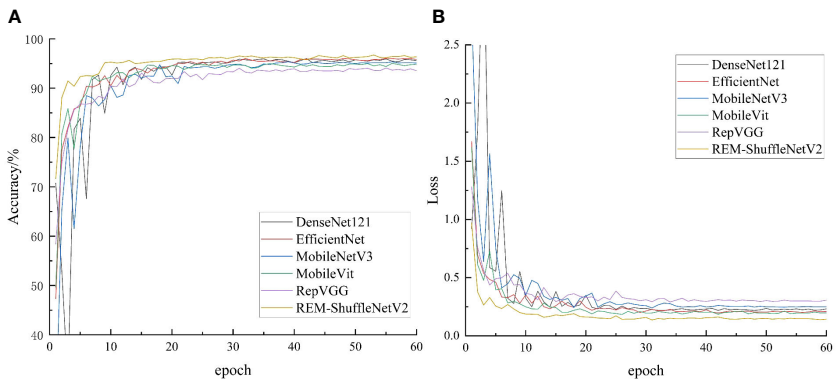


FIGURE 10
Accuracy and loss of comparative network model. **(A)** Accuracy; **(B)** Loss value.

REM-ShuffleNetV2 decreases the fastest and obviously, and at 20 rounds of iteration, the loss value basically stabilizes, and the network loss value maintains around 0.146. From the perspective of loss-value convergence, the REM-ShuffleNetV2 model is ideally trained. Other measures of the model are shown in Table 7.

As shown in Table 7, compared to the conventional models DenseNet121 and RepVGG, the REM-ShuffleNetV2 lightweight model had higher accuracy and F1 scores, and the number of parameters was significantly reduced. Compared with the lightweight convolutional networks EfficientNet, MobileNetV3, and MobileVit, the number of parameters of the REM-ShuffleNetV2 model were only 28.57%, 27.28% and 59.38% of those of EfficientNet, MobileNetV3 and MobileVit, but the model's F1 scores and accuracy are 1.45% and 0.66%, 1.93% and 1.43%, 3.01% and 1.67% higher than them respectively. In summary, the REM-ShuffleNetV2 model achieves good performance in terms of performance and complexity.

4.9 Analysis of model robustness performance

To further verify the anti-interference ability of the REM-ShuffleNetV2 model, we performed a variety of treatments on the test set, including adding Gaussian noise, performing a rotation process, and adjusting the luminance to simulate more realistic environmental conditions (as shown in Figure 11). These treatments help to evaluate the performance of the models in the face of complex, variable environments and thus provide a more complete picture of their robustness and reliability. The classification accuracy of each model under different treatments is shown in Table 8.

Table 8 shows that the classification accuracy of each model generally decreases more significantly when Gaussian noise and 30-degree rotation treatments are added. Under Gaussian noise processing, the recognition effect of the REM-ShuffleNetV2 model is significantly worse than that of the DenseNet121 model and the EfficientNet model; while under the brightness reduction processing, the recognition effect of the REM-ShuffleNetV2 model is slightly lower than that of the MobileVit model. However, under other conditions of processing, the recognition effect of the REM-ShuffleNetV2 model is better than the other models. Taken together, REM-ShuffleNetV2 still shows good recognition results under different treatments, showing good robustness.

4.10 Confusion matrix for different models

The confusion matrix is usually used as an evaluation metric for machine learning classification models, which can demonstrate the number of observations that are misclassified and right-classified by the model, thus assessing the performance of the model (Bi et al., 2023). In the dataset used in this experiment, the types of apple leaf diseases are the most numerous, and different apple leaf diseases only have slight differences in a certain localization, which is characterized by "high within-class variance and low between-class variance", therefore, the confusion matrix of apple leaf diseases was used to present the results, as shown in Figure 12.

From Figure 12, Alternaria leaf spot and gray spot are easily confused because of their high spot similarity, while scab and powdery mildew are easily confused with healthy leaf because their early spot characteristics are not obvious and basically indistinguishable from those of healthy leaf, which leads to misclassification in the model. The REM-ShuffleNetV2 model performed well in the identification of confusing apple leaf diseases with a number of recognition errors of 28, which was comparable to the Efficientnet model. Compared with the DenseNet121, MobileNetV3, MobileVit, and RepVGG models, the REM-ShuffleNetV2 reduced 6, 5, 7, and 10 recognition errors, respectively.

5 Discussion

Advanced convolutional neural networks are often designed to be deep and wide to learn patterns of features from different objects. However, in the crop leaf disease images used in this paper, the disease features are similar and scattered, and no obvious patterns exist to be learned. Therefore, blindly stacking the number of network layers and increasing the model width may overfit useless feature information without improving the performance of the model. On the contrary, doing so may increase the number of parameters and computational effort of the model, thus affecting the efficiency and usefulness of the model (Peng and Li, 2023). In this

TABLE 7 Performance comparison results of different models.

Model	Precision/%	Recall/%	F1 score /%	Accuracy/%	Parameters/M	Model size/MB
DenseNet121	95.69	94.99	95.30	96.00	26.61	27.21
EfficientNet	95.67	94.82	95.17	96.06	15.40	15.69
MobileNetV3	95.01	94.49	94.69	95.29	16.14	16.34
MobileVit	93.84	93.66	93.61	95.05	7.41	7.56
RepVGG	94.11	93.03	93.41	94.09	29.97	30.18
REM-ShuffleNetV2	96.85	96.48	96.62	96.72	4.40	6.23

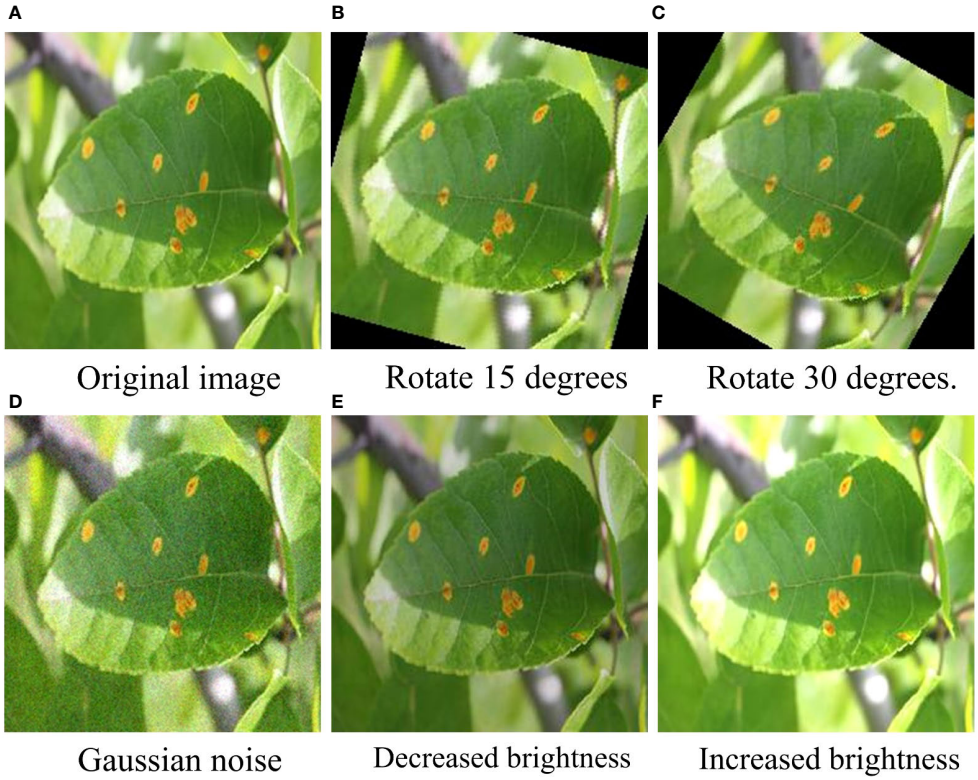


FIGURE 11
Example plots under different treatments. **(A)** Original image. **(B)** Rotate 15 degrees. **(C)** Rotate 30 degrees. **(D)** Gaussian noise. **(E)** Decreased brightness. **(F)** Increased brightness.

study, it was found that properly reducing the parameters of the model did not degrade the model performance, but rather improved it. This indicates that appropriately reducing the number of parameters of the model helps the model learn features better. Therefore, reducing the number of parameters of the model appropriately for a specific task and dataset may be an effective strategy to help improve the performance and generalization of the model.

In the task of image classification, the region of interest is often distributed in multiple regions of the image, and more global information and higher-level feature information are needed to better recognize the target. The smaller the receptive field is, the smaller the range of the original image to which it corresponds, which means that it contains features that tend to be more localized and detailed, and the high-level semantic information used to deal with the complex task is difficult to be captured by the network; the larger the receptive field is, the larger the range of the original image to which it corresponds, which means that it contains more global and higher semantic level features. In the real environment, crop diseases have problems such as different sizes of spots and a wide range of disease distribution. In this paper, a multi-scale feature extraction module is introduced to enhance the model's ability to extract feature information at different scales and to solve the problem of losing small feature information due to downsampling. To further improve the model performance, this paper draws on the idea of ResNet and introduces a residual

TABLE 8 Classification accuracy of each model under different treatments.

Model	A	B	C	D	E	F
DenseNet121	96.00	90.69	85.20	94.63	90.45	92.36
EfficientNet	96.06	90.45	83.53	93.32	90.69	94.27
MobileNetV3	95.29	80.85	72.85	88.42	88.07	93.50
MobileVit	95.05	88.60	82.70	89.68	91.83	93.20
RepVGG	94.09	72.85	59.55	88.96	88.60	88.31
REM-ShuffleNetV2	96.72	91.53	86.87	89.68	91.64	94.87

A, B, C, D, E, and F in the table correspond to the different treatments of a, b, c, d, e, and f in Figure 11, respectively. The data in the table are in %.

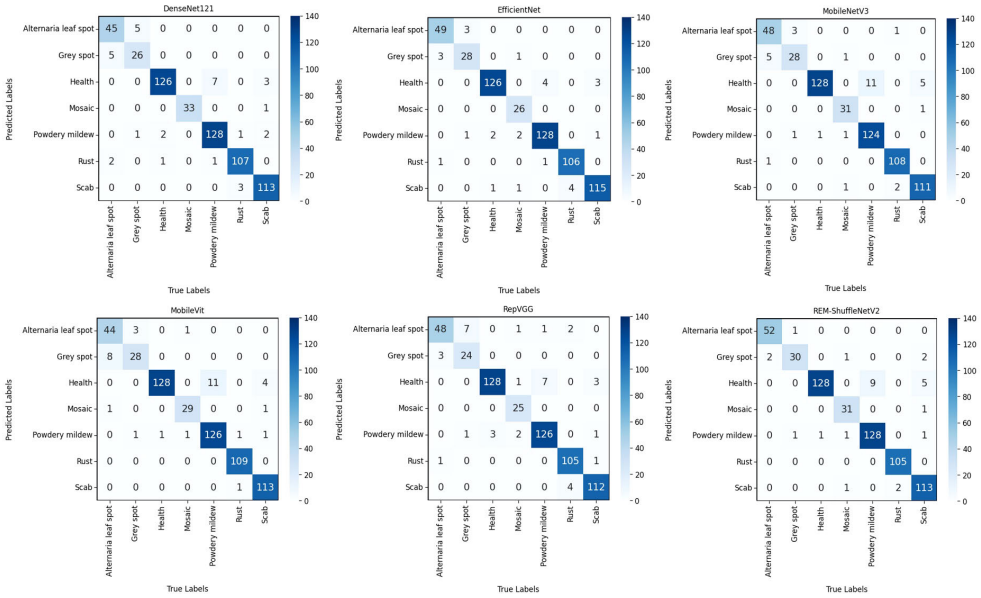


FIGURE 12
Confusion matrix of different models.

structure to overcome the problems of gradient vanishing and gradient explosion during network training, to better fit the data.

Attentional mechanisms are often used to improve the performance of models by better aggregating information about the features of the network model for the region of interest and reducing the influence of extraneous background (Sun et al., 2022; Liao et al., 2023). However, different attention mechanisms work differently and have different impacts on model performance. Compared with other attention mechanisms, the introduction of the EDCA module designed in this paper can effectively improve the performance of the ShuffleNetV2 model for crop leaf disease recognition. This is because the attention module uses two different pooling layers to couple the global information and a local cross-channel interaction strategy without dimensionality reduction to obtain more accurate attention information by aggregating the cross-channel information with a one-dimensional convolutional layer.

Although the study has achieved some results, there are still some limitations. Firstly, the sample images used in the experiment were taken under real environments on sunny or cloudy days, and although realistic factors were taken into account to a certain extent, further in-depth research is needed to fully reflect the performance under various environmental conditions. Secondly, due to the limitation of shooting conditions, the types of disease samples collected are limited, which limits the application range of the model to a certain extent. In future work, we will collect more plant disease data from real scenarios, covering different types, parts and developmental stages of the disease, and develop more efficient and accurate deep learning models to be able to differentiate between more types of crop disease. In addition, we try to deploy the model to cell phones to help farmers find diseases on plants in time so that they can take appropriate control measures to prevent the spread of diseases. In addition, we also plan to deploy it into field management robots for real-time monitoring of crop diseases. This will help professionals understand the type, distribution and severity of diseases and develop more effective disease management strategies.

6 Conclusions

Aiming at the problems of low recognition accuracy and complex model structure of existing models, this paper proposes a lightweight crop leaf disease recognition model REM-ShuffleNetV2. First, we build a field crop disease dataset, which contains 22 categories of 6 crops with a total of 8408 sample images. To reduce the complexity of the model, architectural adjustments were made to the ShuffleNetV2 model. The residual structure was introduced in the basic feature extraction module, which solved the problem of the model's gradient disappearing during the training process and improved the convergence speed of the model. To improve the model's ability to extract effective features in complex backgrounds, we used the EDCA module to filter out the complex interference information in the samples. Meanwhile, we also introduced the MSFEM module and MDFEM module designed in this paper to improve the model's ability to extract feature information at different scales. Finally, the REM-ShuffleNetV2 model achieved 96.72% recognition accuracy on the crop leaf disease test set, which increased by 3.86% compared to the ShuffleNetV2 model.

In order to further evaluate the performance of the REM-ShuffleNetV2 model, we conducted comparison experiments with the DenseNet121, ResNet18, MobileNetV2, and GhostNet models. The experimental results show that the recognition accuracies of the REM-ShuffleNetV2 model were 0.72%, 1.67%, 2.09%, and 11.52% higher than these models, while the model structure was more streamlined. In addition, the superiority of the REM-ShuffleNetV2 model in fine-grained classification is further demonstrated by the analysis of the confusion matrix.

Data availability statement

The dataset and code in this study can be accessed by contacting the corresponding author.

Author contributions

HZ: Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Funding acquisition. JC: Methodology, Data curation, Formal analysis, Writing – original draft. XN: Project administration, Writing – review & editing. ZD: Supervision, Visualization, Writing – review & editing. LQ: Software, Investigation, Writing – review & editing. LM: Supervision, Writing – original draft. JL: Software, Validation, Writing – original draft. YS: Investigation, Data curation, Writing – original draft. QW: Writing – review & editing, Project administration, Supervision.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was funded by the National Natural Science Foundation of China (52379039, 51909079), the Scientific and Technological Project of Henan Province, China (212102110035), and the Young Backbone Teacher Project of Henan University of Science and Technology (13450001).

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

Agarwal, M., Gupta, S. K., and Biswas, K. K. (2020). Development of Efficient CNN model for Tomato crop disease identification. *Sustain. Computing: Inf. Syst.* 28, 100407. doi: 10.1016/j.suscom.2020.100407

Bao, W., Cheng, T., Zhou, X. G., Guo, W., Wang, Y. Y., Zhang, X., et al. (2022). An improved DenseNet model to classify the damage caused by cotton aphid. *Comput. Electron. Agric.* 203, 107485. doi: 10.1016/j.compag.2022.107485

Barbedo, A. G. J. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst. Eng.* 144, 52–60. doi: 10.1016/j.biosystemseng.2016.01.017

Bhagat, M., Kumar, D., and Kumar, S. (2023). Bell pepper leaf disease classification with LBP and VGG-16 based fused features and RF classifier. *Int. J. Inf. Technol.* 15, 465–475. doi: 10.1007/s41870-022-01136-z

Bi, C., Xu, S., Hu, N., Zhang, S., Zhu, Z., and Yu, H. (2023). Identification method of corn leaf disease based on improved mobilenetv3 model. *Agronomy* 13, 300. doi: 10.3390/agronomy13020300

Camargo, A., and Smith, J. S. (2009). Image pattern classification for the identification of disease causing agents in plants. *Comput. Electron. Agric.* 66, 121–125. doi: 10.1016/j.compag.2009.01.003

Ding, X. H., Zhang, X. Y., Ma, N. N., Han, J. G., Ding, G. G., and Sun, J. (2021). "Repvgg: Making vgg-style convnets great again," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Vol. 2021. 13733–13742.

Du, H. S., Zhang, C. L., An, W. H., Zhou, Y., Zhang, Z., and Hao, X. X. (2023). Crop disease recognition based on multi-layer information fusion and saliency feature enhancement. *Trans. Chin. Soc. Agric. Machinery* 54, 214–222. doi: 10.6041/j.issn.1000-1298.2023.07.021

Gao, Y., Cao, Z., Cai, W., Gong, G. F., Zhou, G. X., and Li, L. J. (2023). Apple leaf disease identification in complex background based on BAM-net. *Agronomy* 13, 1240. doi: 10.3390/agronomy13051240

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*. 315–323 (JMLR Workshop and Conference Proceedings).

Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., and Jasińska, E. (2021). Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics* 10, 1388. doi: 10.3390/electronics10121388

He, D. J., Wang, P., Niu, T., Miao, Y. R., and Zhao, Y. R. (2022). Classification model of grape downy mildew disease degree in field based on improved residual network. *Trans. Chin. Soc. Agric. Machinery* 53, 235–243. doi: 10.6041/j.issn.1000-1298.2022.01.026

Hossain, S. M. M., Deb, K., Dhar, P. K., Dhar, P. K., and Koshiba, T. (2021). Plant leaf disease recognition using depth-wise separable convolution-based models. *Symmetry* 13, 511. doi: 10.3390/sym13030511

Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M. X., et al. (2019). "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, Vol. 2019. 1314–1324.

Huang, G., Liu, Z., Maaten, L., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4700–4708.

Huang, L. S., Luo, Y.W., Yang, X. D., Yang, G. J., and Wang, D. Y. (2021). Crop disease recognition based on attention mechanism and multi-scale residual network. *Trans. Chin. Soc. Agric. Machinery* 52, 264–271. doi: 710.6041/j.issn.1000-1298.2021.10.027

Huang, L. W., Zheng, L., Huang, Y., Qian, B., and Guan, F. F. (2023). Strawberry disease recognition method based on multi-scale convolution and channel domain enhancement. *Jiangsu Agric. Sci.* 51, 202–210. doi: 10.15889/j.issn.1002-1302.2023.10.028

Le, Y., Yu, X. Y., Zhang, S. P., Long, H. B., Zhang, H. H., Xu, S., et al. (2023). GoogLeNet based on residual network and attention mechanism identification of rice leaf diseases. *Comput. Electron. Agric.* 204, 107543. doi: 10.1016/j.compag.2022.107543

Lee, H. J., Kim, H. E., and Nam, H. (2019). "Srm: A style-based recalibration module for convolutional neural networks," in *Proceedings of the IEEE/CVF International conference on computer vision*. 1854–1862.

Li, Z. B., Yang, Y. B., Li, Y., Guo, R. H., Yang, J. Q., and Yue, J. (2020). A solanaceae disease recognition model based on SE-Inception. *Comput. Electron. Agric.* 178, 105792. doi: 10.1016/j.compag.2020.105792

Liao, Y. J., Yang, L., Shao, P., and Yu, X. Y. (2023). Improved identification of leaf diseases and pest infestations on rice by means of coordinate attention mechanism-based residual network. *Fujian J. Agric. Sci.* 38, 1220–1229. doi: 10.19303/j.issn.1008-0384.2023.10.011

Liu, X. L., and Cui, Y. Y. (2023). An improved method for apple leaf disease identification based on lightweight network MobileNeXt. *Jiangsu Agric. Sci.* 51, 185–193. doi: 10.15889/j.issn.1002-1302.2023.10.026

Liu, W., Wang, Z., Liu, X., Zeng, N. Y., Liu, Y. R., and Alsaadi, F. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing* 234, 11–26. doi: 10.1016/j.neucom.2016.12.038

Liu, Y. Y., Wang, D. K., Wu, L., Huang, D. C., and Zhu, L. (2023). A light-weight model for plant disease identification based on model pruning and knowledge distillation. *Acta Agriculturae Zhejiangensis* 35, 2250–2264. doi: 10.3969/j.issn.1004-1524.20221193

Ma, J., Du, K., Zhang, L., Zheng, F., Chu, J., and Sun, Z. (2017). A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. *Comput. Electron. Agric.* 142, 110–117. doi: 10.1016/j.compag.2017.08.023

Ma, N., Zhang, X., Zheng, H. T., and Sun, J. (2018). "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*. 116–131.

Mehta, S., and Rastegari, M. (2021). Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint* arXiv:2110.02178. doi: 10.48550/arXiv.2110.02178

Muhab, H., and Ercan, A. (2022). Tipburn disorder detection in strawberry leaves using convolutional neural networks and particle swarm optimization. *Multimedia Tools Appl.* 81 (8), 11795–11822. doi: 10.1007/s11042-022-12759-6

Peng, H. X., He, H. J., Gao, Z. M., Tian, X. G., Deng, Q. T., and Xian, C. L. (2022). Litchi diseases and insect pests identification method based on improved shuffleNetV2. *Trans. Chin. Soc. Agric. Machinery* 53, 290–300. doi: 10.6041/j.issn.1000-1298.2022.12.028

Peng, Y. H., and Li, S. Q. (2023). Recognizing crop leaf diseases using reparameterized MobileNetV2. *Trans. Chin. Soc. Agric. Eng.* 39, 132–140. doi: 10.11975/j.issn.1002-6819.202304241

Praveen, S. P., Nakka, R., Chokka, A., Thatha, V. N., Vellela, S. S., and Sirisha, U. (2023). A novel classification approach for grape leaf disease detection based on different attention deep learning techniques. *Int. J. Advanced Comput. Sci. Appl.* 14 (6). doi: 10.14569/IJACSA.2023.01406128

Rangarajan, A. K., Purushothaman, R., and Ramesh, A. (2018). Tomato crop disease classification using pre-trained deep learning algorithm. *Proc. Comput. Sci.* 133, 1040–1047. doi: 10.1016/j.procs.2018.07.070

Saeedan, F., Weber, N., Goesele, M., and Roth, S. (2018). "Detail-preserving pooling in deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9108–9116.

Shah, M. P., Singha, S., and Awate, S. P. (2017). "Leaf classification using marginalized shape context and shape+ texture dual-path deep convolutional neural network," in *2017 IEEE International conference on image processing (ICIP)*. 860–864 (IEEE).

Shorten, C., and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *J. big Data* 6, 1–48. doi: 10.1186/s40537-019-0197-0

Sun, L. Q., Wang, X. L., Wang, B. N., Wang, J. Y., and Meng, X. Y. (2022). Identification method of fish satiation level based on resNet-CA. *Trans. Chin. Soc. Agric. Machinery* 53, 219–225+277. doi: 10.6041/j.issn.1000-1298.2022.S2.025

Sun, J., Zhu, W. D., Luo, Y. Q., Shen, J. F., Chen, Y. D., and Zhou, X. (2021). Recognizing the diseases of crop leaves in fields using improved Mobilenet-V2. *Trans. Chin. Soc. Agric. Eng.* 37, 161–169. doi: 10.15302/J-SSCAE-2021.02.022

Szegedy, C., Ioffe, S., and Vanhoucke, V. (2017). "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI conference on artificial intelligence*, Vol. 31.

Tan, M., and Le, Q. (2019). "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. 6105–6114 (PMLR).

Tian, K., Zhang., L. K., Xiong, M. D., Huang, Z. H., and Li, J. H. (2016). Recognition of phomopsis vexans in solanum melongena based on leaf disease spot features. *Trans. Chin. Soc. Agric. Eng.* 32, 184–189. doi: 10.11975/j.issn.1002-6819.2016.z1.026

Wang, D. F., and Wang, J. (2021). Crop disease classification with transfer learning and residual networks. *Trans. Chin. Soc. Agric. Eng.* 37, 199–207. doi: 10.11975/j.issn.1002-6819.2021.04.024

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11534–11542.

Wen, Z. F., Pu, Z., Cheng, X., and Zhao, Y. J. (2023). Recognizing crop diseases in fields based on knowledge distillation and essNet. *Shandong Agric. Sci.* 55, 154–163. doi: 10.14083/j.issn.1001-4942.2023.05.023

Yang, D., Du, Y., Yao, H., and Bao, L. Y. (2022). Image semantic segmentation with hierarchical feature fusion based on deep neural network. *Connection Sci.* 34, 1772–1784. doi: 10.1080/09540091.2022.2082384

Ye, Z. H., Zhao, M. X., and Jia, L. (2021). Image recognition of crop diseases in complex background. *Trans. Chin. Soc. Agric. Machinery* 52, 118–124+147. doi: 10.6041/j.issn.1000-1298.2021.S0.015

Zhang, J. H., Kong, F. T., Li, Z. M., Wu, J. Z., Chen, W., Wang, S. W., et al. (2014). Recognition of honey pomelo leaf diseases based on optimal binary tree support vector machine. *Trans. Chin. Soc. Agric. Eng.* 30, 222–231. doi: 10.3969/j.issn.1002-6819.2014.19.027

Zhang, Y., Li, Q. X., and Wu, H. R. (2020). Rapid recognition model of tomato leaf diseases based on kernel mutual subspace method. *Smart Agric.* 2, 86–97. doi: 10.12133/j.smartag.2020.2.3.202009-SA001

Zhang, S. W., and Zhang, C. L. (2014). Maize disease recognition based on local discriminant algorithm. *Trans. Chin. Soc. Agric. Eng.* 30, 167–172. doi: 10.3969/j.issn.1002-6819.2014.11.021

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 04 April 2024
DOI 10.3389/fpls.2024.1334362

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Hiroshi Mineno,
Shizuoka University, Japan
Guowei Dai,
Chinese Academy of Agricultural Sciences, China

*CORRESPONDENCE
Yuzuko Utsumi
yuzuko@omu.ac.jp

RECEIVED 07 November 2023
ACCEPTED 21 February 2024
PUBLISHED 04 April 2024

CITATION
Nakatani T, Utsumi Y, Fujimoto K, Iwamura M and Kise K (2024) Image recognition-based petal arrangement estimation.
Front. Plant Sci. 15:1334362.
doi: 10.3389/fpls.2024.1334362

COPYRIGHT
© 2024 Nakatani, Utsumi, Fujimoto, Iwamura and Kise. 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.

Image recognition-based petal arrangement estimation

Tomoya Nakatani[1], Yuzuko Utsumi[1]*, Koichi Fujimoto[2], Masakazu Iwamura[1] and Koichi Kise[1]

[1]Graduate School of Informatics, Osaka Metropolitan University, Sakai, Japan, [2]Graduate School of Integrated Sciences for Life, Hiroshima University, Higashi-Hiroshima, Japan

Flowers exhibit morphological diversity in the number and positional arrangement of their floral organs, such as petals. The petal arrangements of blooming flowers are represented by the overlap position relation between neighboring petals, an indicator of the floral developmental process; however, only specialists are capable of the petal arrangement identification. Therefore, we propose a method to support the estimation of the arrangement of the perianth organs, including petals and tepals, using image recognition techniques. The problem for realizing the method is that it is not possible to prepare a large number of image datasets: we cannot apply the latest machine learning based image processing methods, which require a large number of images. Therefore, we describe the tepal arrangement as a sequence of interior-exterior patterns of tepal overlap in the image, and estimate the tepal arrangement by matching the pattern with the known patterns. We also use methods that require less or no training data to implement the method: the fine-tuned YOLO v5 model for flower detection, GrubCut for flower segmentation, the Harris corner detector for tepal overlap detection, MAML-based interior-exterior estimation, and circular permutation matching for tepal arrangement estimation. Experimental results showed good accuracy when flower detection, segmentation, overlap location estimation, interior-exterior estimation, and circle permutation matching-based tepal arrangement estimation were evaluated independently. However, the accuracy decreased when they were integrated. Therefore, we developed a user interface for manual correction of the position of overlap estimation and interior-exterior pattern estimation, which ensures the quality of tepal arrangement estimation.

KEYWORDS

plant measurement, tepal arrangement, segmentation, meta-learning, circular permutation matching

1 Introduction

Flowers are of highly diverse shapes in their body plants, for instance, in the number of perianth organs such as petals and sepals, and in the symmetry of the arrangements of the organs (Endress, 1999; Tucker, 1999; Smyth, 2018; Spencer and Kim, 2018). Among these characteristics, the arrangement symmetry changes from species to species, where close species may differ and distant species may not. Hence, the flower development process may transcend the difference in species. The formation of flower organs during flower development has been studied using mathematical models via computer simulation to understand the developmental bases of the flower shape diversity (Nakagawa et al., 2020). To verify such flower development models, comparing the positional arrangement of organs in blooming flowers with models is useful.

This study aims to examine the positional arrangement of petal-like perianth organs (tepals). The development order of tepals can be estimated from the arrangement of neighboring tepals by studying a flower not fully bloomed (Figure 1). Exterior tepals are typically initiated earlier during the floral development in many clades of plant species, including *Anemone* and *Arabidopsis* (Smyth et al., 1990; Ren et al., 2010). At present, tepal arrangements are identified manually (Endress, 2010; Ronse De Craene, 2010; Vislobokov et al., 2014; Kitazawa and Fujimoto, 2018; Kitazawa and Fujimoto, 2020). The visual examination of a large number of flower images is extremely labor intensive; moreover, judging these images unless one is a specialist is difficult. Thus, this study aims to automate this task to reduce the burden on researchers.

We use image data for several *Anemone* species and cultivars that were collected for the development of the flower shapes model. Every image in this data was manually captured when the flowers were blooming and shows how the tepals are arranged. For this reason, the number of flower images that can be used for machine learning is small, unlike large-scale datasets such as ImageNet (Deng et al., 2009) or Microsoft Common Objects in Context (MS COCO) (Lin et al., 2014) that are often used in image recognition. In addition to the small number of images included in these data, the lack of diversity in the shapes of the tepals is a problem when these images are used. The appearance of the tepals is quite different for individual flowers (diversity), and due to the small number of pictures (small quantity), there are not many images showing flowers that have a similar appearance (Figure 2). Therefore, it is difficult to build an end-to-end classifier that takes images as input and then outputs tepal arrangement.

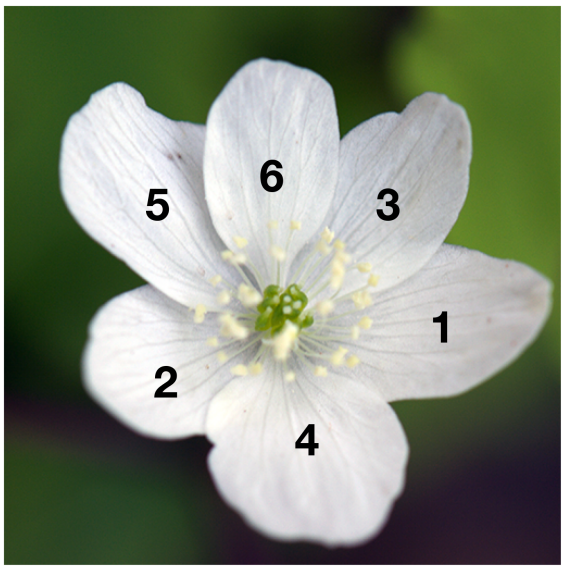


FIGURE 1
Example image of *Anemone* flower; index on each tepal (perianth organ) denotes the positioning, where exterior and interior organs take lower and higher indices, respectively.

Therefore, we propose estimating the tepal arrangement from the interior-exterior pattern of the overlap between tepals. This method detects the overlapping parts of the tepals, extracts the interior-exterior patterns of the tepals, and then identifies the tepal arrangements through circular permutation matching. Considering that there is only a small amount of training data, flower detection is performed using the You Only Look Once (YOLO)[1] algorithm, which is capable of object detection after fine-tuning on a small number of training images. Moreover, the recognition of flower regions and the detection of overlapping parts are conducted by segmentation using GrabCut (Rother et al., 2004), which does not require training, and corner detection using the Harris corner detector (Harris and Stephens, 1988). Additionally, meta-learning, which is a machine learning framework that enables effective learning using a small amount of data, is used for the identification of interior-exterior relationships of the tepals.

Circular permutation matching, which is used to estimate the tepal arrangement, cannot estimate correctly if even one result for the overlapping part or the interior-exterior relationship is incorrect. Also, due to the property of pattern recognition, the detection of overlapping parts and the recognition of the interior-exterior relationship are not perfect. Thus, the accuracy of the estimation of tepal arrangement decreases due to these errors. Therefore, we attempt to improve the estimation accuracy using manual correction of the detection and recognition results. The result of the study showed that with total automation, the estimation accuracy of tepal arrangement was 0.275. However, after manual correction, the estimation accuracy increased to 0.711.

2 Related work

Here, we introduce studies in which image processing was used for flower image observation in plant measurement and studies related to tepal overlapping order estimation. As a flower is a familiar object, it is one of the commonly used image subjects in the fields of computer vision and pattern recognition. ImageNet (Deng et al., 2009), which is a representative common-object image dataset, also contains a category for flowers. Moreover, species recognition by flower appearance has been studied because flowers

1 https://github.com/ultralytics/yolov5.



FIGURE 2
Floral shape and color diversity in *Anemone* species, *A. pulsatilla* (left) and *A. x hybrida* (center, right).

have diverse shapes and colors. Oxford flower datasets (Nilsback and Zisserman, 2006, 2008), are well-known and widely used in the field of computer vision. In this datasets, there are two categories; one comprises 17 types of images, where each type contains 80 images; the other comprises 102 types of images, where each type contains 40 to 258 images. Many image recognition studies have used the Oxford flower datasets (Nilsback and Zisserman, 2006, 2008; Fernando et al., 2012a, b, 2014; Hu et al., 2014; Mabrouk et al., 2014; Xie et al., 2014; Yang et al., 2014; Wang et al., 2020). Additionally, segmentation has also been carried out using the datasets (Nilsback and Zisserman, 2010; Chai et al., 2011; Chai et al., 2012). Moreover, a method that performs recognition through the production of a unique dataset (Guru et al., 2011) has been proposed, as has an interactive segmentation method (Zou and Nagy, 2004). There are studies on flower recognition that do not analyze flower structure.

In the field of agriculture, flower detection in images is being studied for the automation of farm work and the prediction of crop yield. In indoor environments such as greenhouses, flower detection methods for the automation of tomato flower picking (Rahim and Mineno, 2020) and the automatic pollination of raspberries and blackberries (Ohi et al., 2018) have been studied. Furthermore, flower detection in crops grown outdoors is being done with deep learning, which offers object detection that is robust in any lighting conditions. Flower detection from images is currently in use for crop prediction and the automation of tasks such as picking flowers and pollinating strawberries (Chen et al., 2019; Lin et al., 2020), apples (Dias et al., 2018; Wu et al., 2020), grapes (Millan et al., 2017; Liu et al., 2018), and kiwis (Williams et al., 2020). However, these are methods for flower location detection that do not analyze flower structure.

Some studies modeled the shape details of flower organs such as petals. A method that reproduces the detailed shapes of flowers, including their petals, from computed tomography images (Ijiri et al., 2014), methods that chronologically model the flower blooming process (Ijiri et al., 2008; Yan et al., 2014), and a method that reconstruct shapes of flowers in actual images using the RGB-D image data and petal model of color and shape that was prepared in advance (Zhang et al., 2014) have been developed. These models were built to draw flowers using computer graphics, and they require the measurement data for three-dimensional (3D) shapes and the application of 3D models. Thus, they are not suitable for analyzing the shapes of flowers that bloom outdoors.

3 Materials and methods

3.1 Dataset

In this study, we used 3,942 flower images of several *Anemone* species: *A. flaccida, A. hepatica*, *A. x hybrida, A. nikoensis, A. pulsatilla, and A. soyensis*. These images were photographed in locations with Shiga, Kyoto, Hyogo, Okayama, and Hiroshima Prefecture, using an Olympus TG-5 digital camera and a Nikon D5200 SLR camera. The image resolutions were 568 × 3,712, 4,928 × 3,264, 4,000 × 3,000, and 4,000 × 2,672 pixels; all images were saved in JPEG format. The size of the images was normalized to 1,000 × 662 pixels for the experiment. The images used in this study can be found in the data repository: 10.6084/m9.figshare.25323112.

Bounding boxes to indicate the position of the flowers were attached to all of these images. Moreover, the position of tepal overlap in the bounding boxes, and interior-exterior relationship of the tepals were studied and manually labeled. As the tepal arrangement of each pair of neighboring tepals is uniquely determined by their interior-exterior pattern, the arrangement of tepals was correctly labeled according to the interior-exterior relationship of the overlapping tepals. Moreover, labels to indicate the flower region at the pixel level were manually attached inside the bounding boxes in 420 images.

3.2 Tepal arrangement

Tepal arrangement represents the number of tepals and their interior-exterior relationship; it is determined by the tepal development order. The interior-exterior relations of tepals can be represented as a circular permutation by attaching a class label to each tepal. There are three types of class labels (Figure 3): I (Internal), where a tepal is above both the tepals on its sides; E (External), where a tepal is below both the tepals on its sides; and A (Alternating), where a tepal is above the neighboring tepal on one side and below on the other. The tepal arrangements represented as

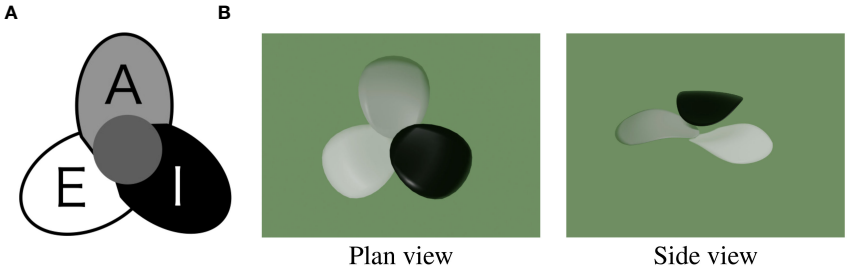


FIGURE 3
Class label I, E and A indicating the interior-exterior relations of tepals. **(A)** 2D diagram **(B)** 3D visualization[2].

circular permutations that match with each other after rotation or reflection transformation are treated as the same arrangement. Only some of all the conceivable arrangements occur in nature, and one-three types have been identified for each tepal number.

In this study, *Anemone* species and cultivars having 4–9 tepals are studied. Figure 4 shows a diagram of a model that includes 13 of the tepal arrangement types frequently observed in flowers with these tepal counts. The color of each tepal in this diagram indicates its class label; black, gray, and white represent I, A, and E, respectively. Moreover, the letter shown in the lower part of the diagram indicates the number of tepals (A–F correspond to 4–9, respectively), and the number indicates the tepal arrangement order for each tepal number.

3.3 Estimation of tepal arrangement

Figure 5 shows the process flow for the proposed method. Tepal arrangement, which is the object to be estimated, can be uniquely determined from the interior-exterior relations between all of the overlapping tepals. Thus, the overlapping parts of the tepals and their interior-exterior relations are estimated after the flower region is obtained through flower detection and segmentation. Finally, the tepal arrangement is estimated using the estimated interior-exterior relations. In the following section, we describe the details of flower detection, segmentation, tepal overlap detection, the estimation of interior-exterior relations, and the tepal arrangement estimation.

3.3.1 Flower detection

For the preprocessing, the proposed method carried out flower detection in the original image to obtain a rectangle that indicated the position of the flower (the bounding box). The proposed method used YOLO ver. 5 [3] for flower detection. YOLO v5 has shown good accuracy in common-object detection, and it is expected to be highly accurate in flower detection after fine-tuning, even if the training dataset is small. YOLO v5 simultaneously detects and classifies objects using a convolutional neural network, and the original model could detect and classify 80 objects. As the proposed method only needs to detect the flower region, we fine-tune the model using flower images.

3.3.2 Flower region segmentation

Because the estimation of the tepal interior-exterior relationship is affected by the image background, the flower region is extracted from the detected image. As the flowers have a color that is different from their background, the proposed method utilizes GrabCut (Rother et al., 2004), which is a color-based segmentation technique, to segment the flower region. GrabCut employs a Gaussian mixture model to represent the color distribution of the foreground and background, and it performs the segmentation using this model. Because the model is estimated individually for each image, no prior training is required. The color distribution model is constructed based on a bounding box that encompasses the user-defined foreground region. As the bounding box defined in 3.3.1 encompasses the flower region, we use this bounding box to automatically construct the model and segment the flower region instead of using the user-specified rectangle.

3.3.3 Overlapping tepal detection

Next, the points at which the tepals overlap are detected. The points are detected by using the Harris corner detector (Harris and Stephens, 1988). As the corner detector is not perfect, the detected points contain false positives (Figure 6, left image). This image shows that when the region around a detected point is examined, the overlapping point is found to be concave, whereas false positives are convex, as shown in the image on the right side of Figure 6. Therefore, the proposed method detects the overlapping points by using this convex/concave characteristic as follows:

1. Draw a circle around the detected point, as shown in green in the close-up image of a corner on the right side of Figure 6, and obtain two points that intersect with the contour of the tepal (red points in the figure).
2. Draw a straight line that connects the two obtained points (red broken line, Figure 6). If the entire line is above the foreground, the point is considered to be a false positive and is removed.
3. When the entire line is above the background, it is determined to be an overlapping point.

2 3D movie is available at https://youtu.be/clYmer-nvO4.

3 https://github.com/ultralytics/yolov5.

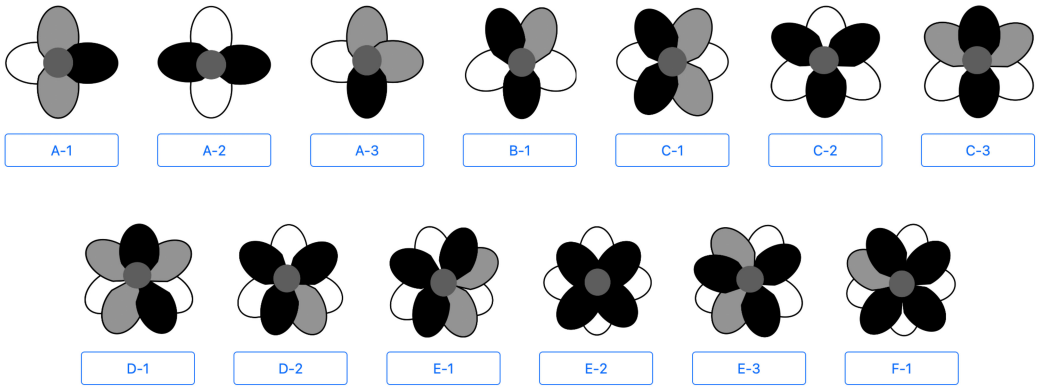


FIGURE 4
Diagram of tepal arrangement model.

3.3.4 Recognition of interior-exterior relationship between tepals

The recognition of the interior-exterior relation of tepals is solved as a binary classification problem for images of tepal overlapping regions. The two classes are the left-tepal-forward class (L) and the right-tepal-forward class (R) (Figure 7). The images of tepal overlapping regions (the patches) are cropped from the segmented images around the overlapping points.

Each flower in the images used in this study has tepals of a different color and shape from the other flowers. Therefore, the appearance of the cropped patches is diverse. Moreover, the number of patches that can be cropped from one segmented image is 6 or 7 at most. This number of patches is insufficient for training a classification model; therefore, the model cannot achieve sufficient classification accuracy. Thus, this study employs synthesized petal overlapping patches (hereafter referred to as synthetic patches) and meta-learning to achieve sufficient accuracy. Synthetic patches are used to compensate for the small number of cropped patches and to generate an adequate number of patches for training. Meta-learning is used to prevent a reduction in classification accuracy due to differences in the appearance of the flowers.

3.3.4.1 Generate synthetic patches

Figure 8 shows how the synthetic patches are generated through image processing. First, we draw a circle with the center as the position of overlap of the tepals, as shown in red on the left side of Figure 8. Next, we crop two patches, which are shown as the green squares centered at the intersections of the red circle and the contour on the left side of Figure 8. These patches are called the original patches. The radius of the red circle must be set so that the cropped region does not contain the overlapping parts of the tepals. Lastly, by superimposing these two patches, a synthetic patch that simulates the overlapping of the tepals is generated. As it is obvious whether the synthetic patches belong to L or R classes, the correct class label can automatically be applied to the synthetic patches; this cannot be done for the cropped patches.

3.3.4.2 Meta-learning

Next, model-agnostic meta-learning (MAML) (Finn et al., 2017), which is one of the meta-learning methods, is used to classify the cropped patches. Meta-learning is a framework that optimizes the learning of new tasks through learning various tasks. The object of optimization is different for each method. In MAML (Finn et al., 2017), the optimization is done by learning the initial model of a network that is capable of accommodating many different tasks.

Figure 9 shows the meta-learning framework used in this study. The tasks used for learning the model are called the meta-training, and the target tasks are called the meta-testing. Each task consists of a support set and a test set. The support set includes the training

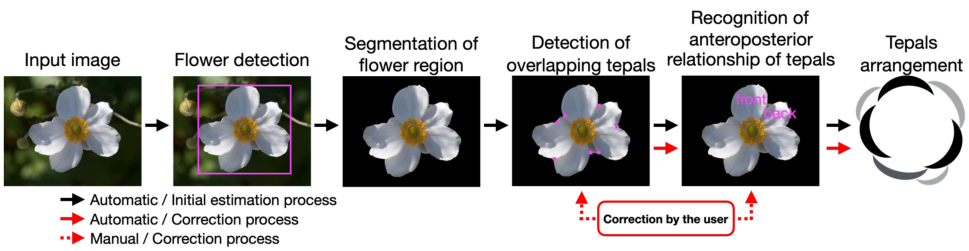


FIGURE 5
Overview of proposed method.

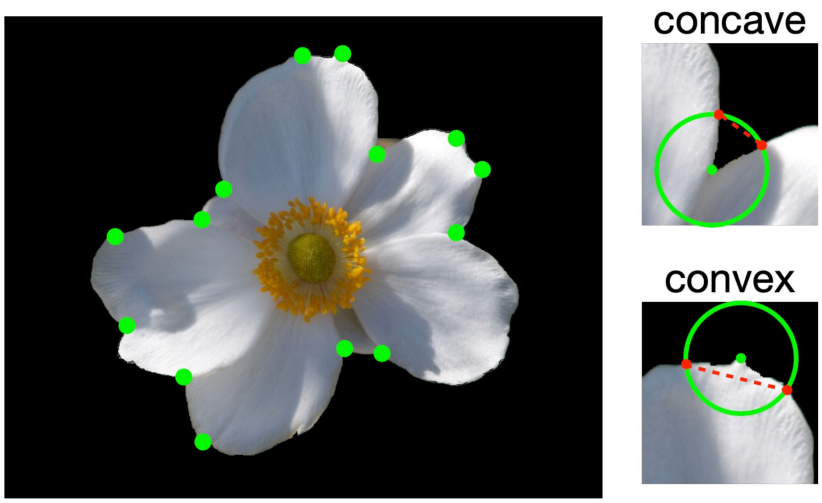


FIGURE 6
Detected corners (green points) and convex/concave corners in surrounding boundaries.

data, which are used for training the model, and the query set contains the test data for evaluating the model. In MAML, the model is trained on the support set within a meta-training task to derive parameters that are appropriate for the task. Next, the performance of the model is evaluated using a query set in the meta-training task and the model parameters are updated based on the error of the test data. By iterating this process multiple times for each task in the meta-training, the model is trained to rapidly adapt to the various tasks. Finally, the model is evaluated using meta-testing. The model parameters that were trained by meta-training are updated using the support set for the meta-testing, and then the performance of the updated model is evaluated using the query set for the meta-testing.

In this study, each task consists of patches generated from a single flower, and synthetic patches are used for the support set, while the query set is the original patches for the interior-exterior estimation, as shown in Figure 9. In the training process, the model is trained using synthetic patches from a support set of a meta-training task and evaluated using original patches from a query set of a meta-training task. The process is iterated for the number of meta-training tasks. Then, in the estimation process, the parameters of the model are optimized for a meta-testing task using the support set of the task, and the estimation is performed on the query set of the task with the optimized parameters.

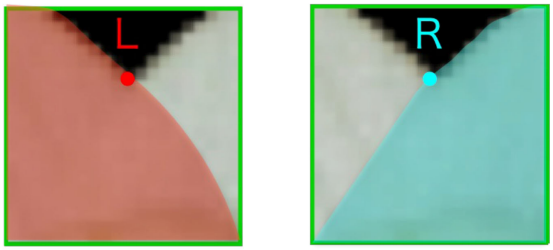


FIGURE 7
Examples from left-(L) (left) and right-tepal-forward class (R) (right).

3.3.5 Tepal arrangement estimation

The tepal arrangement estimation is performed using the arrangement from the interior-exterior relation recognition results. The estimation consists of three steps.

First, we obtain the class labels (L, R) of the tepal overlapping as a circular permutation. We set the center of gravity as the origin O (Figure 10) and consider the angle between the red line, OX, in the horizontal rightward direction from the origin and the line connecting the overlapping points (light blue) and the origin. The circle permutation is obtained by arranging the class labels in the order of increasing angle at each overlapping point.

Next, the class labels representing the interior-exterior relation of a pair of tepals are converted to class labels for tepals (Figure 3). When the circular permutation of L and R labels representing the interior-exterior relation is clockwise, a sequence of two letters, such as LR, can be converted into one of the I, E, or A labels (Figure 11). Following this conversion rule, the permutation represented by L and R is converted to a permutation of tepal class labels indicated by I, E, and A.

Finally, the tepal arrangement being estimated is compared with the 13 known types of tepal arrangements (Figure 4). Concretely, the class label permutations represented by I, E, and A are

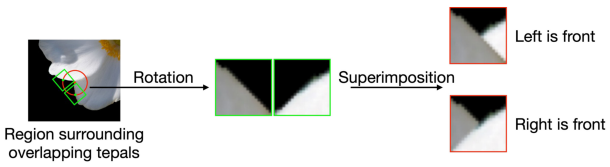


FIGURE 8
Procedure for generating synthetic patches.

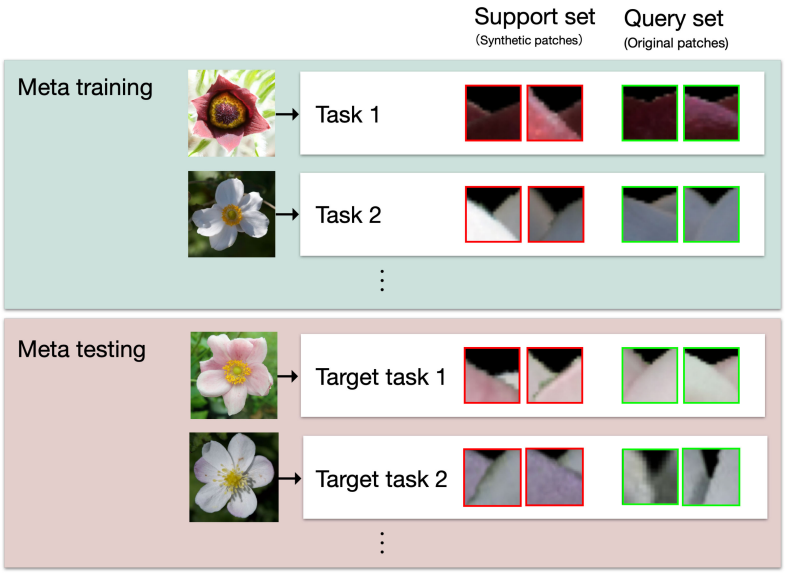


FIGURE 9
The meta-learning framework for this study. Each task consists of patches generated or cropped from a single flower. Support sets, which are used to train or optimize the model parameters, consist of synthetic patches. Query sets, which are used to evaluate the model, consist of original patches. In the training process, meta-training tasks are used to train and evaluate the model. In the evaluation process, meta-testing tasks are used for model optimization and evaluation.

compared with the class label permutations extracted from the tepal arrangement candidates. Edit distance (Navarro, 2001) is used as the metric for the estimation. The edit distance is the minimum number of single-character edits (insertions, deletions, or substitutions) needed to change one string into another. Then, the tepal arrangement that gives the shortest edit distance is regarded as the estimation result.

As the class label permutations are circular permutations, considering the rotation and flip transformations in computing the edit distance between two permutations is necessary. Therefore, the edit distance is calculated by rotating the circular permutation. Additionally, the edit distance is calculated after a flip transformation is carried out.

4 Experiments

To evaluate the accuracy of the proposed method, we executed experiments using the original dataset introduced in Section 3.1. In the experiments, the accuracy of each intermediate process was evaluated. This is because the accuracy of petal arrangement estimation is heavily dependent on the accuracy of the intermediate processing steps. Next, the accuracy of the tepal arrangement estimation was evaluated after all the processing steps were combined. Lastly, the accuracy of the case was evaluated when the results for the tepal overlap positions and for the recognition of their interior-exterior relationship in the intermediate processes were manually corrected. The following sections explain each experimental condition and describe the results. The research ethics committee of the Graduate School of Informatics, Osaka Metropolitan University approved the experiments on manual correction.

4.1 Experimental conditions

The conditions used to evaluate the accuracy of each intermediate process are first explained. For flower detection, 200 images from the original dataset were used for training, and 3,742 images were used for testing. We used the original implementation of the YOLO v5(m) model[4], which is pretrained on MS COCO. As the size of the input image for the YOLO v5(m) model is 256×256 pixels, the images were scaled down to 256×256 pixels using bilinear interpolation. As described in Section 3.1, the images used in the experiment were 1,000× 662-pixel landscape images. Therefore, in order not to change the aspect ratio of the images, the image was scaled down to 256 × 169 pixels. Then, black pixels were added to the scaled image to make the image size fit to the input of YOLO v5 (m). In fine-tuning, following the original YOLO v5(m) implementation, blur, random size cropping, and binary histogram equalization were applied to the training data with 1% probability to augment the data. The augmentation was performed using the function included in the original implementation. Training was performed by changing the number of training epochs and batch size of the original implementation to 500 and 16, respectively. The implementation was based on the deep learning framework Pytorch 1.9.1 on Python 3.7.7. Intersection

4 https://github.com/ultralytics/yolov5.

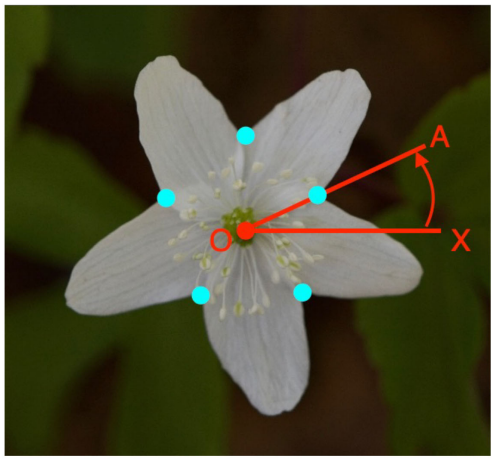


FIGURE 10
Acquisition of a circular permutation of overlapping tepal classes.

over Union (IoU), which is commonly used in object detection, was used as the evaluation metric for flower detection. When the detection result region for a given object (pixel set) is set to A, and the correct region is set to B, the IoU value is calculated using the following Equation (1):

$$\mathrm{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

As the flower detection result is indicated by the bounding box, the IoU value was calculated using the bounding boxes of the detection result and the correct region.

Flower segmentation was evaluated on 220 images to which flower regions were manually assigned. The 220 images were different from the 200 images used for fine-tuning the flower detection method. The size of the images was 1,000×662 pixels. Segmentation was conducted by setting the flower region inside the bounding box as the foreground. We used opencv-python 4.5.3.56, which is the computer vision library OpenCV[5] for Python, which was used for implementing segmentation. We used the function grabCut, which implements GrabCut. The initialization of grabCut, which indicates the format of the foreground settings, was set to the rectangles. The rectangles containing the flowers, which were determined manually, were given to the function. The number of algorithm iterations was set to 5. IoU was used to evaluate the accuracy, as was the case for the flower detection. As the detection result and the correct region are presented as flower regions at the pixel level, these regions were used to calculate the IoU value.

To evaluate the tepal overlap detection, 3,742 images that were manually labeled with the tepal overlap positions were used. The images used for evaluation were the same as those used for flower detection evaluation. The size of the images was 1,000 × 662 pixels. The flower segmentation method described in Section 3.3.2 was applied to the flower images, and the tepal overlap detection was performed on the segmented flower regions in the images. As for the flower segmentation, opencv-python 4.5.3.56 was used for the implementation. We used the cornerHarris function, which implements the Harris corner detector. The size of the circle radius used to determine the overlap area was 15 pixels. The F1 score (F_1), which is the harmonic mean of the precision ratio and the recall ratio, was used in the accuracy evaluation. The F1 score is calculated using the following Equation (2):

$$F_1 = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} \tag{2}$$

For the interior-exterior estimation of the tepals, 200 images were used to determine the initial parameters (meta-training), and 220 images were used for the testing (meta-testing). As the interior-exterior estimation method requires the use of images from which only flower regions are cropped, it is necessary to prepare images from which flower regions are correctly cropped for training and testing. Therefore, we used 200 and 220 manually flower-segmented images for training and testing, respectively. The 200 images used for training were the same as those used for fine-tuning the flower detection method.

Patches of size 20 × 20 pixels, whose centers were the manually labeled positions of the tepal overlap, were cropped for the meta-training and meta-testing for query sets. Note that each patch was cropped so that one side was parallel to the straight line connecting the two intersections of the circle whose center was the point of overlap and the contour, as shown in the concave image in Figure 6. In addition, the cropped patches were aligned so that the contour part of the tepal was on top of each cropped patch. From a single flower, 4 to 16 patches were cropped, and a total of 1,575 original patches were cropped. The synthesis patches were generated following in Section 3.3.4.1 in each flower image. The radius of the red circle shown in Figure 8 for cropping the patches for synthesis was set to 10 and 15 pixels. Two patches of 20×20 pixels were cropped from each point of intersection of the red circle and the contour of the tepals. If the number of cropped patches was less than 50, we cropped patches around the intersections, adding perturbations, until the number of patches was increased to 50. After the cropping was done for all the overlapping points in a flower image, two types of patches were cropped, 25 patches each: contour arcs that slope upward from left to right, and contour arcs that slope upward from right to left. From each cropped patch type, one patch was selected and superimposed so that the contours of the tepals were intersected at the top to create two synthetic images: left or right in front. Synthetic patches were generated for all combinations of the cropped patches. From a single flower image, 1250 synthetic patches were generated.

For the meta-training, 200,000 tasks were prepared, and 20 tasks were prepared for the meta-testing. To construct a meta-training task, we randomly selected one image out of 200 training flower images, and then we randomly sampled 20 synthetic and 2 original patches generated from the selected flower image for the support and query sets, respectively. We iterated the above process 200,000 times, then generated 200,000 tasks for meta-training. It was acceptable for the same flower image to be selected during

5 https://opencv.org.

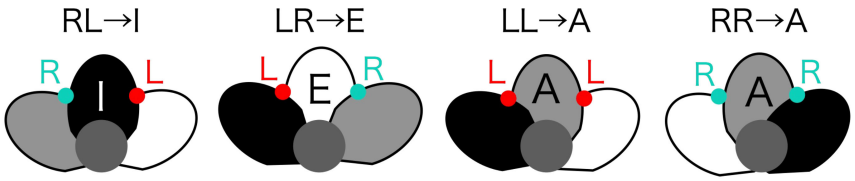


FIGURE 11
Rule for converting tepal overlapping labels to tepal labels.

task generation. This is because even if the same flower image was selected, we selected the patches in such way that the combination of the patches is different from the previously generated tasks. The meta-testing tasks were generated from 220 flower images for the test. Each flower image generated a task. Each task consists of a support set of 10 synthetic patches for each class, for a total of 20 patches, and a query set of 2 original patches, one for each class.

We used the same network model as for the MAML (Finn et al., 2017). We used the implementation with Pytorch, which is a deep learning framework written in Python[6]. The code was written in Python 3, and TensorFlow v1.0+, which is a Python library for machine learning, was used to implement the model. The model consisted of 4 concatenated blocks, including batch normalization, ReLU, and 2×2 max-pooling after a 3×3 convolutional layer. As the input to the model was 84×84 pixels, we upsampled the patches from 20×20 to 84×84 pixels using bilinear interpolation. For training, no pretraining was executed, and the model was trained from scratch. The meta train iteration, meta batch size, meta learning rate, update batch size, and update learning rate were set to 200,000, 20, 0.001, 5, and 0.01, respectively. No augmentation was performed during the training. Recognition accuracy, which is the ratio of correct interior-exterior relations to the total overlap positions, was used as the evaluation metric.

We also executed an experiment that integrated all the intermediate processes and estimated the tepal arrangement in the input images. A total of 3,742 flower images were used for the evaluation; each image contained one flower. The size of the images was 1000 × 662 pixels. For each process, unless otherwise noted, the model and parameter settings were the same as for each intermediate process evaluation. Detection results, which were shown in rectangles, were given as the foreground of the flower segmentation process. After segmentation, overlapping tepals detection was performed on the segmentation results. The interior-exterior estimation was performed on the detected overlapping points. For the petal interior-exterior estimation, the same meta-training tasks that were used to evaluate the intermediate process were used, and the meta-testing tasks were generated using the 3,742 flower images, which were used for flower detection evaluation. The tepal arrangements were estimated using the results of interior-exterior estimation. In the tepal arrangement estimation, the edit distance cost of insert, delete and replace was set to 1. Recognition accuracy, which is the ratio of correct to the total number of flowers, was used as the evaluation metric.

To evaluate manual correction, 15 of the 3,742 images were used. The size of the images was 1000 × 662 pixels. There were 9 participants in the experiment. All participants were graduate and undergraduate students majoring in computer science and were not botany experts. The experiment was conducted using a web application we developed for the experiment. Figure 12 shows an example of what the application displays to the user in a web browser when it asks the user to modify intersections or estimation results. The web application presents an image of a flower, the position of tepal overlap detection results, and the interior-exterior estimation relationship of the petals. One can correct the detection and estimation results by mouse operation through the web application. We asked the participants to find the incorrect overlap detection results and interior-exterior relation estimation results and correct the results by mouse operation. Then, we estimate the tepal arrangements using the corrected results. The application was developed using a Python web framework Django 4.1.2 [7] to display images and estimation results in a browser and to collect user mouse events. The tepal arrangement estimation results from the images were computed in the same experimental environment as the integrated experiments. Recognition accuracy, which is the ratio of correctly estimated arrangements to the total test numbers of all the experiment participants, was used for the accuracy evaluation.

We used a GPU server to evaluate the deep learning-based intermediate processes: fine-tuning and evaluation of YOLO v5 for flower detection and training and evaluation of MAML for tepal interiorexterior estimation. The GPU server was equipped with NVIDIA TITIAN RTX and 24 GB of memory. We used a CPU server for the evaluation of flower segmentation, tepal overlap detection, and tepal arrangement estimation. We also used the CPU server for the integrated experiment and the web application server for the manual correction experiment. The CPU server was equipped with an Opteron 6348 and 512 GB of memory.

6 https://github.com/dragen1860/MAML-Pytorch.

7 https://www.djangoproject.com.

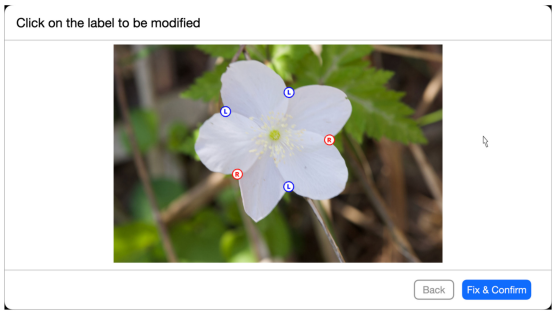


FIGURE 12
An example of the application display used in the manual correction evaluation experiment. The circles superimposed on the image indicate the detected overlap points, and their colors and letters indicate the results of the interior-exterior estimation, with the blue circled letter L and the red circled letter R on the left and on the right. If the user clicks on a circle in the estimation result, the estimation result is modified.

4.2 Result

Table 1 shows the results of the accuracy evaluation of the intermediate process. The accuracy was determined to be 0.275 after all the intermediate processes were integrated, and tepal arrangement was estimated from the input images. When manual correction was applied, the accuracy was 0.711. The average time for manual correction for one flower was 48 seconds.

5 Discussion

Table 1 shows that flower detection and segmentation accuracy was particularly high relative to the accuracy of each intermediate process. IoU metric is sensitive to misalignment. Figure 13 shows an example of a flower detection result when the lowest IoU was shown, 0.906. Figure 13 shows that the segmentation is almost perfect, as the detection result overlapped the ground truth. However, a small deviation from the ground truth decreased the IoU value. In PASCAL VOC challenge 3[8], a common-object recognition competition, object detection was deemed to be successful when IoU was 0.5. As IoU for flower detection in this study was 0.939, the flower detection was quite accurate.

As the foreground and background in the most of the flower images used for this experiment were different, as shown in Figure 2, flower detection was a relatively easy task. In addition, IoU value for segmentation was high at 0.974. Similar to detection, this is assumed to be because of the difference between the background and flower colors, making the segmentation task easy. However, several images failed segmentation because the background has the similar color to the foreground, as shown in Figure 14. GrabCut used color information to determine the foreground and background using color information.

When using Grabcut, segmentation fails if the background has a similar color to the foreground. To correctly segment such images, deep learning based segmentation methods can be effective.

Figure 15 shows examples of the result of tepal overlap detection. This result contains errors: false positives and a false negative. In Figure 15A, the tepal indentations were incorrectly detected as overlaps. As the overlap detection method considers the concave points as overlaps, the overlap detection method failed on the flower images whose tepals have indentations. In Figure 15B, the overlap detection method missed the overlap point, shown by the blue circle. The Harris corner detector failed to detect the bluecircled corner. As the overlap detection method used in this study was a rule-based method, the detection failed when an exception occurred. To improve the accuracy, learning-based detection could be introduced.

The accuracy of the interior-exterior estimation was not satisfactory enough to automate the tepal arrangement estimation. Figure 16 shows an example with incorrect interior-exterior estimation results. In the enlarged upper two patches in Figure 16, the shading of the tepal folds would affect the estimation. In the lower patch, the disappearing contour of the tepal in front would affect the estimation. These changes in the appearance of the patches are caused by the lighting conditions when the images were taken. The accuracy can be improved by keeping the lighting conditions constant. However, since the flower images used in this study were taken outdoors, it is difficult to take all the images under constant lighting conditions.

In the integrated experiments, the accuracy was 0.275. This is significantly less accurate than any of the intermediate processes. To clarify the reason for the low accuracy, we theoretically calculated the accuracy when intermediate processes are integrated using the accuracy of the independent intermediate process evaluations. Since the average number of tepals in the flower images used in this study was 6.3, we consider an image of a flower with 6 tepals. The flower has 6 tepal overlapping points. Given the high accuracy of flower detection and segmentation in the experiments, we assume that they are always successful. The experimental results show that the precision and estimation accuracy for the detection of the overlapping point detection and the interior-exterior estimation are 0.889 and 0.849, respectively. For the sake of simplicity, we assume that the overlapping point detection and the interior-exterior estimation are independent. Then, the accuracy that all overlapping point detection and interior-exterior estimation will be successful for the flower is calculated by the 6 powers of the product of the detection and estimation as follows:

$$(0.889 * 0.849)^6 = 0.18486585651.$$

The calculated accuracy is lower than the accuracy of the integrated experiment, 0.275. This is probably due to the fact that even though the overlapping point detection or the interior-exterior estimation failed, some of the flowers succeeded in the circular permutation matching of the tepal order estimation process. However, the improvement is not sufficient.

To improve the accuracy of the tepal arrangement estimation, it is essential to improve the accuracy of overlapping point detection and interior-exterior estimation. Overlap point detection could be

8 http://host.robots.ox.ac.uk/pascal/VOC/.

TABLE 1 Evaluation of intermediate process.

Intermediate process	Number of training images	Number of test images	Evaluation metric	Score
Flower detection	200	3742	IoU	0.939
Segmentation	–	220	IoU	0.974
Tepal overlap detection	–	3742	F1 score	0.870
Tepal interior-exterior relationship recognition	200	220	accuracy	0.849

improved by using a learning-based detection method. However, it would be difficult to improve the accuracy of interior-exterior estimation because the lighting condition when the images were taken is one of the causes of the lower accuracy of interior-exterior estimation, and the lighting condition is not constant in the outdoor environment. Therefore, improving interior-exterior relation estimation is not feasible. Another possible cause of the low accuracy is the division of the tepal arrangement estimation into overlapping point detection and interior-exterior estimation. As calculated above, each task can be assumed to be independent, so even a slight decrease in the accuracy of one task will significantly affect the accuracy when the results are integrated. A possible solution to this problem is to estimate the tepal arrangement directly from the flower image without splitting the task.

The manual correction experiment showed that manual correction could significantly improve the arrangement estimation, and the tepal arrangements can be estimated by amateurs not used to observing flowers. When the number of tepals is less than 10, as in this study, experts can estimate the tepal arrangement in a few seconds without the proposed method. Although it takes more time than with the experts, the amateurs can estimate the tepal arrangement without any flower observation experience using the proposed method with manual correction. Amateurs can estimate the tepal arrangement instead of the experts, and the experts can reduce the effort of estimating the tepal arrangement. As anyone who can estimate the tepal interior-exterior relationship can be assigned to the tepal arrangement, the estimation work can be parallelized by multiple people. Although the time for estimating the tepal arrangement by the proposed method with manual correction is longer than that of the experts, parallelization can improve the efficiency of the estimation work.

Additionally, results from manually corrected intermediate processes could be used as the ground truth data for training. If enough manually corrected data are collected, replacing the current rule-based method used for tepal interior-exterior estimation with a learning-based method would be possible. As a result, the accuracy of the estimation of tepal interior-exterior relationship would improve. In addition, as manual correction makes it easy to assign the correct tepal arrangement to the input flower image, preparing a large number of flower images with tepal arrangements would be smooth. Next, constructing an end-to-end network that performs image input and placement estimation would be possible. In the proposed method, tepal arrangement estimation was performed by combining several independent intermediate methods due to the difficulty of preparing training data to train an end-to-end network, leading to a decrease in accuracy. If an end-to-end network trained with sufficient training data and built by the proposed method with manual correction could be constructed, it would be possible to achieve highly accurate tepal arrangement estimation automatically.

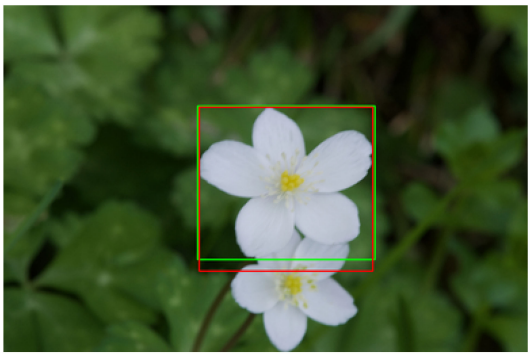


FIGURE 13
The example of the flower detection result with the lowest IoU, 0.906. The red and green rectangles show the detection result and ground truth.



FIGURE 14
The example of the flower segmentation result with the IoU was 0.761. The light red area shows the segmented area as the foreground. The background area, which was flowers overlapping with the foreground flower, was also determined to be foreground.

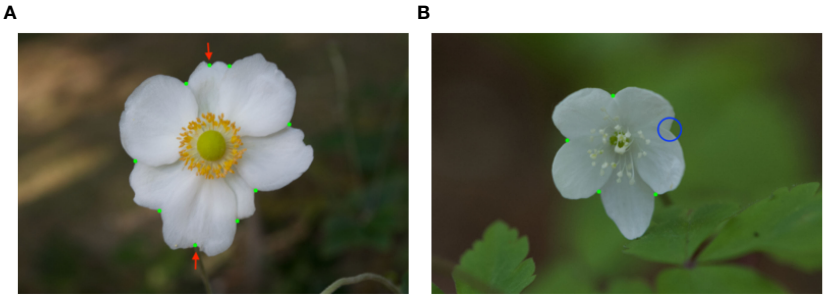


FIGURE 15
The examples of tepal overlap detection results. The green dots show the detected tepal overlap. **(A)** includes false positives, shown by the green dots with the red arrows. **(B)** includes false negative, shown by the blue circle.



FIGURE 16
The example of failed for interior-exterior estimation. The red circles and patches show the tepal overlap points and the patches that failed to the interior-exterior estimation.

6 Conclusion

This study focused on tepal arrangement, an indicator of the floral developmental process. Currently, specialists classify tepal arrangements in flower images manually. This requires a tremendous amount of labor. Therefore, using the image recognition method to estimate tepal arrangement from flower images was proposed to reduce specialists' workload. As it was difficult to collect a large amount of data, a segmentation method not requiring learning and a method that could accurately recognize the interiorexterior relationships of tepals with a small amount of training data were used. Our experiment showed that, although flower detection and segmentation could be performed accurately, tepal overlap detection and the estimation of the tepal interior-exterior relationship were inaccurate, which resulted in low accuracy in the tepal arrangement estimation. However, the accuracy improved significantly when the tepal overlap detection and interior-exterior relationship recognition results were manually corrected. The results showed that manual correction can be used to assign the ground truth of a tepal arrangement to a large number of images. Thus, an end-to-end network expected to be highly accurate could be realized by using a large number of images with manually assigned tepal arrangements.

Data availability statement

The original contributions presented in the study are publicly available. This data can be found here: https://drive.google.com/drive/u/2/folders/1IpJrhKur7POvEDyzyQl37fhSwBZvuRwZ.

Ethics statement

The studies involving humans were approved by The research ethics committee of the Graduate School of Informatics, Osaka Metropolitan University. The studies were conducted in accordance with the local legislation and institutional requirements. The participants provided their written informed consent to participate in this study.

Author contributions

TN: Software, Validation, Visualization, Writing – original draft. YU: Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Writing – original draft, Writing – review & editing. KF: Conceptualization, Data curation, Supervision, Writing – review & editing. MI: Conceptualization, Investigation, Methodology, Supervision, Writing – review & editing. KK: Resources, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This study was supported by JSPS Grants-in-Aid for Scientific Research JP20H05423 and JP22H04732.

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

Chai, Y., Lempitsky, V., and Zisserman, A. (2011). "BiCoS: A bi-level co-segmentation method for image classification," in *Proceedings of 2011 International Conference on Computer Vision* (Berlin: Springer). 2579–2586. doi: 10.1109/ICCV.2011.6126546

Chai, Y., Rahtu, E., Lempitsky, V., Van Gool, L., and Zisserman, A. (2012). "TriCoS: A tri-level classdiscriminative co-segmentation method for image classification," in *Proceedings of European Conference on Computer Vision*. 794–807. doi: 10.1007/978-3-642-33718-557

Chen, Y., Lee, W. S., Gan, H., Peres, N., Fraisse, C., Zhang, Y., et al. (2019). Strawberry yield prediction based on a deep neural network using high-resolution aerial orthoimages. *Remote Sens.* 11, 1–21. doi: 10.3390/rs11131584

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Li, F.-F. (2009). "ImageNet: A large-scale hierarchical image database," in *Proceedings of 2009 IEEE Conference on Computer Vision and Pattern Recognition* (New York: IEEE). 248–255. doi: 10.1109/CVPR.2009.5206848

Dias, P. A., Tabb, A., and Medeiros, H. (2018). Multispecies fruit flower detection using a refined semantic segmentation network. *IEEE Robot. Automat. Lett.* 3, 3003–3010. doi: 10.1109/LRA.2018.2849498

Endress, P. K. (1999). Symmetry in flowers: diversity and evolution. *Int. J. Plant Sci.* 160, S3–S23. doi: 10.1086/314211

Endress, P. K. (2010). Flower structure and trends of evolution in eudicots and their major subclades1. *Ann. Missouri Botanic. Garden* 97, 541–583. doi: 10.3417/2009139

Fernando, B., Fromont, E., Muselet, D., and Sebban, M. (2012a). "Discriminative feature fusion for image classification," in *Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition* (New York: IEEE). 3434–3441. doi: 10.1109/CVPR.2012.6248084

Fernando, B., Fromont, E., and Tuytelaars, T. (2012b). "Effective use of frequent itemset mining for image classification," in *Proceedings of European Conference on Computer Vision* (Berlin: Springer). 214–227. doi: 10.1007/978-3-642-33718-516

Fernando, B., Fromont, E., and Tuytelaars, T. (2014). Mining mid-level features for image classification. *Int. J. Comput. Vision* 108, 186–203. doi: 10.1007/s11263-014-0700-1

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* (Sydney: PMLR). 1126–1135.

Guru, D. S., Sharath Kumar, Y. H., and Manjunath, S. (2011). Textural features in flower classification. *Math. Comput. Model.* 54, 1030–1036. doi: 10.1016/j.mcm.2010.11.032

Harris, C., and Stephens, M. (1988). "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference* (Manchester: AlvetyVisionClub). 147–151.

Hu, W., Hu, R., Xie, N., Ling, H., and Maybank, S. (2014). Image classification using multiscale information fusion based on saliency driven nonlinear diffusion filtering. *IEEE Trans. Image Process.* 23, 1513–1526. doi: 10.1109/TIP.2014.2303639

Ijiri, T., Yokoo, M., Kawabata, S., and Igarashi, T. (2008). "Surface-based growth simulation for opening flowers," in *Proceedings of Graphics Interface 2008* (NewYork: ACM). 227–234.

Ijiri, T., Yoshizawa, S., Yokota, H., and Igarashi, T. (2014). Flower modeling via x-ray computed tomography. *ACM Trans. Graphics* 33, 1–10. doi: 10.1145/2601097.2601124

Kitazawa, M. S., and Fujimoto, K. (2018). Spiral phyllotaxis underlies constrained variation in *Anemone* (*Ranunculaceae*) tepal arrangement. *J. Plant Res.* 131, 459–468. doi: 10.1007/s10265-018-1025-x

Kitazawa, M. S., and Fujimoto, K. (2020). Perianth phyllotaxis is polymorphic in the basal eudicot *anemone* and *eranthis* species. *Front. Ecol. Evol.* 8. doi: 10.3389/fevo.2020.00070

Lin, P., Lee, W. S., Chen, Y. M., Peres, N., and Fraisse, C. (2020). A deep-level region-based visual representation architecture for detecting strawberry flowers in an outdoor field. *Precis. Agric.* 21, 387–402. doi: 10.1007/s11119-019-09673-7

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft COCO: common objects in context. *Proc. Eur. Conf. Comput. Vision.*, 740–755. doi: 10.1007/978-3-319-10602-148

Liu, S., Li, X., Wu, H., Xin, B., Tang, J., Petrie, P. R., et al. (2018). A robust automated flower estimation system for grape vines. *Biosyst. Eng.* 172, 110–123. doi: 10.1016/j.biosystemseng.2018.05.009

Mabrouk, A. B., Najjar, A., and Zagrouba, E. (2014). "Image flower recognition based on a new method for color feature extraction," in *Proceedings of 2014 International Conference on Computer Vision Theory and Applications (VISAPP)* (Set`ubal: Science and Technology Publications). 201–206. doi: 10.5220/0004636302010206

Millan, B., Aquino, A., Diago, M. P., and Tardaguila, J. (2017). Image analysis-based modelling for flower number estimation in grapevine. *J. Sci. Food Agric.* 97, 784–792. doi: 10.1002/jsfa.7797

Nakagawa, A., Kitazawa, M. S., and Fujimoto, K. (2020). A design principle for floral organ number and arrangement in flowers with bilateral symmetry. *Development* 147, 1–10. doi: 10.1242/dev.182907

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Comput. Surv.* 33, 31–88. doi: 10.1145/375360.375365

Nilsback, M. E., and Zisserman, A. (2006). "A visual vocabulary for flower classification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (New York: IEEE). 1447–1454. doi: 10.1109/CVPR.2006.42

Nilsback, M.-E., and Zisserman, A. (2008). "Automated flower classification over a large number of classes," in *Proceedings of 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing* (New York: IEEE). 722–729. doi: 10.1109/ICVGIP.2008.47

Nilsback, M. E., and Zisserman, A. (2010). Delving deeper into the whorl of flower segmentation. *Image Vision Comput.* 28, 1049–1062. doi: 10.1016/j.imavis.2009.10.001

Ohi, N., Lassak, K., Watson, R., Strader, J., Du, Y., Yang, C., et al. (2018). "Design of an autonomous precision pollination robot," in *Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (New York: IEEE). 7711–7718. doi: 10.1109/IROS.2018.8594444

Rahim, U. F., and Mineno, H. (2020). Tomato flower detection and counting in greenhouses using faster region-based convolutional neural network. *J. Image Graphics* 8, 107–113. doi: 10.18178/joig.8.4.107-113

Ren, Y., Chang, H.-L., and Endress, P. K. (2010). Floral development in *anemoneae* (*Ranunculaceae*). *Botanic. J. Linn. Soc.* 162, 77–100. doi: 10.1111/j.1095-8339.2009.01017.x

Ronse De Craene, L. P. (2010). *Floral Diagrams: An Aid to Understanding Flower Morphology and Evolution* (Cambridge: Cambridge University Press).

Rother, C., Kolmogorov, V., and Blake, A. (2004). "GrabCut: interactive foreground extraction using iterated graph cuts," in *Proceedings of ACM SIGGRAPH 2004*, New York, New York, USA (New York: ACM). 309. doi: 10.1145/1186562.1015720

Smyth, D. R. (2018). Evolution and genetic control of the floral ground plan. *New Phytol.* 220, 70–86. doi: 10.1111/nph.15282

Smyth, D. R., Bowman, J. L., and Meyerowitz, E. M. (1990). Early flower development in arabidopsis. *Plant Cell* 2, 755–767. doi: 10.2307/3869174

Spencer, V., and Kim, M. (2018). Re"CYC"ling molecular regulators in the evolution and development of flower symmetry. *Semin. Cell Dev. Biol.* 79, 16–26. doi: 10.1016/j.semcdb.2017.08

Tucker, S. C. (1999). Evolutionary lability of symmetry in early floral development. *Int. J. Plant Sci.* 160, S25–S39. doi: 10.1086/314212

Vislobokov, N., Sokoloff, D., Degtjareva, G., Valiejo-Roman, C., Kuznetsov, A., and Nuraliev, M. S. (2014). Aspidistra paucitepala (Asparagaceae), a new species with occurrence of the lowest tepal number in flowers of Asparagales. *Phytotaxa* 161, 270–282. doi: 10.11646/phytotaxa.161.4.2

Wang, Z., Wang, K., Wang, X., and Pan, S. (2020). "A convolutional neural network ensemble for flower image classification," in *Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition* (New York: ACM). 225–230. doi: 10.1145/3436369.3437427

Williams, H., Nejati, M., Hussein, S., Penhall, N., Lim, J. Y., Jones, M. H., et al. (2020). Autonomous pollination of individual kiwifruit flowers: Toward a robotic kiwifruit pollinator. *J. Field Robot.* 37, 246–262. doi: 10.1002/rob.21861

Wu, D., Lv, S., Jiang, M., and Song, H. (2020). Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments. *Comput. Electron. Agric.* 178, 105742. doi: 10.1016/j.compag.2020.105742

Xie, L., Tian, Q., Wang, M., and Zhang, B. (2014). Spatial pooling of heterogeneous features for image classification. *IEEE Trans. Image Process.* 23, 1994–2008. doi: 10.1109/TIP.2014.2310117

Yan, F., Gong, M., Cohen-Or, D., Deussen, O., and Chen, B. (2014). Flower reconstruction from a single photo. *Comput. Graphics Forum* 33, 439–447. doi: 10.1111/cgf.12332

Yang, M., Zhang, L., Feng, X., and Zhang, D. (2014). Sparse representation based fisher discrimination dictionary learning for image classification. *Int. J. Comput. Vision* 109, 209–232. doi: 10.1007/s11263-014-0722-8

Zhang, C., Ye, M., Fu, B., and Yang, R. (2014). "Data-driven flower petal modeling with botany priors," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (New York: IEEE). 636–643. doi: 10.1109/CVPR.2014.87

Zou, J., and Nagy, G. (2004). "Evaluation of model-based interactive flower recognition," in *Proceedings of the 17th International Conference on Pattern Recognition* (New York: IEEE). 311–314. doi: 10.1109/ICPR.2004.1334185

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 16 April 2024
DOI 10.3389/fpls.2024.1376915

Check for updates

OPEN ACCESS

EDITED BY
Zhenghong Yu,
Guangdong Polytechnic of Science and Technology, China

REVIEWED BY
Peixian Zhuang,
Tsinghua University, China
Jing Yao,
Chinese Academy of Sciences (CAS), China

*CORRESPONDENCE
Ling Zhou
zhouling@hist.edu.cn

RECEIVED 26 January 2024
ACCEPTED 27 March 2024
PUBLISHED 16 April 2024

CITATION
Li X, Zhai M, Zheng L, Zhou L, Xie X, Zhao W and Zhang W (2024) Efficient residual network using hyperspectral images for corn variety identification.
Front. Plant Sci. 15:1376915.
doi: 10.3389/fpls.2024.1376915

COPYRIGHT
© 2024 Li, Zhai, Zheng, Zhou, Xie, Zhao and Zhang. 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.

Efficient residual network using hyperspectral images for corn variety identification

Xueyong Li[1], Mingjia Zhai[1], Liyuan Zheng[2], Ling Zhou[2*], Xiwang Xie[3], Wenyi Zhao[4] and Weidong Zhang[2]

[1]School of Computer Science and Technology, Henan Institute of Science and Technology, Xinxiang, China, [2]School of Information Engineering, Henan Institute of Science and Technology, Xinxiang, China, [3]School of Information Science and Technology, Dalian Maritime University, Dalian, China, [4]School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China

Corn seeds are an essential element in agricultural production, and accurate identification of their varieties and quality is crucial for planting management, variety improvement, and agricultural product quality control. However, more than traditional manual classification methods are needed to meet the needs of intelligent agriculture. With the rapid development of deep learning methods in the computer field, we propose an efficient residual network named ERNet to identify hyperspectral corn seeds. First, we use linear discriminant analysis to perform dimensionality reduction processing on hyperspectral corn seed images so that the images can be smoothly input into the network. Second, we use effective residual blocks to extract fine-grained features from images. Lastly, we detect and categorize the hyperspectral corn seed images using the classifier softmax. ERNet performs exceptionally well compared to other deep learning techniques and conventional methods. With 98.36% accuracy rate, the result is a valuable reference for classification studies, including hyperspectral corn seed pictures.

KEYWORDS

crop variety, hyperspectral image, channel attention, linear discriminant analysis, deep learning

1 Introduction

The cultivation of maize holds significant economic importance as a pivotal crop. As automation technology becomes increasingly prevalent in the agricultural sector, a growing need for automated classification and identification of corn seeds is needed. Accurately identifying corn seeds is vital for effective planting management, variety enhancement, and quality control of agricultural products (ElMasry et al., 2019). However, traditional manual classification methods can be inefficient and require substantial human resources. In the agricultural field, hyperspectral imaging technology has found extensive application

(Zhang et al., 2022). Hyperspectral images offer multi-band spectral data and capture more comprehensive plant information than traditional RGB images (Wang et al., 2019; Ahmad et al., 2021). Therefore, hyperspectral imaging technology is widely employed in non-destructive testing of crop seed varieties, quality assessment, and vigor analysis (Ma et al., 2020; Zhang et al., 2023a; Zhang et al., 2024a). Nevertheless, the high-dimensional nature of hyperspectral data, complex features, noise, and variations in illumination poses challenges for traditional image processing and classification techniques in recognizing hyperspectral corn seed images (Zhang et al., 2021a; Ghaderizadeh et al., 2022; Huang et al., 2022). Hence, this article aims to enhance corn seed hyperspectral image recognition accuracy and efficiency using the efficient residual network (ERNet).

ERNet is an image classification and recognition model based on deep learning. First, preprocessing and feature extraction were performed on hyperspectral image data of different varieties and qualities of corn seeds. Next, the ERNet model is constructed, trained, and optimized to learn the image's feature representation and classification decision. Finally, the performance and effectiveness of the proposed method will be evaluated, compared, and analyzed with traditional image classification methods.

ERNet enhances model performance and efficiency by incorporating residual connections and lightweight attention mechanism. It leverages collaborative learning strategies among different modules to effectively exploit coarse-grained, fine-grained, and abstract-level features. By fully utilizing the feature extraction capabilities of deep networks, ERNet overcomes the challenges of gradient disappearance and information loss in deep networks, enabling improved learning and image feature extraction. Compared to traditional deep convolutional neural networks, ERNet offers advantages such as reduced parameter count, high computational efficiency, and suitability for processing high-dimensional image data. The critical contributions of the proposed ERNet model in this study can be summarized as:

- We propose ERNet, an efficient residual network specifically designed for identifying corn varieties using hyperspectral data. ERNet leverages the power of residual connections and lightweight attention mechanism to address issues like gradient disappearance and reducing information loss commonly encountered in deep networks. As a result, it dramatically enhances the model's performance and efficiency, leading to more accurate and efficient corn variety identification.
- We introduce two efficient residual modules: identity block-ECA (IBE) and convolutional block-ECA (CBE). These modules incorporate a lightweight efficient channel attention (ECA) mechanism into traditional identity and convolutional residual modules. The ECA aims to enhance the network's accuracy and sensitivity in feature extraction and analysis without altering the convolution operation process or feature map size. This integration significantly improves ERNet's ability to recognize fine-grained features in hyperspectral corn seeds.
- We implemented effective cropping to optimize the utilization of ERNet in extracting finegrained features from hyperspectral corn seed images. This involved removing redundant backgrounds and enlarging the original image features. By employing this approach, we enhanced ERNet's ability to extract detailed and precise features from the images.

The initial section will outline the characteristics of hyperspectral images and emphasize the significance of corn seed identification. The principles and advantages of efficient residual networks will be elaborated upon. The subsequent section will explain this article's research objectives and methods, encompassing data collection and preprocessing, network model construction and training, and other relevant aspects. Finally, the study's significance and anticipated results will be presented.

2 Related works

Extensive research has been conducted by scholars in seed classification, utilizing various methods categorized into traditional, machine learning, and deep learning approaches. The following provides an overview and summary of these research efforts.

Traditional methods have been attempted to be applied in seed recognition and hyperspectral image classification. Gan et al. (Gan et al., 2018) introduced a hyperspectral image classifier based on multi-feature kernel sparse representation. The features were transformed into a nonlinear low dimensional kernel space by employing kernel principal component analysis, enabling the handling of highly nonlinear distributions in hyperspectral image data. Experimental results demonstrated remarkable performance in hyperspectral image classification tasks. Hu et al. (Hu et al., 2020) showcased a promising technology that combined multispectral imaging and multivariate analysis. They utilized the LDA model to achieve 90% accuracy in alfalfa seed classification and SVM to achieve 91.67% accuracy in mycobacterium needle seed classification. Furthermore, Chen et al. (Chen et al., 2023) utilized the interior point hollowing algorithm to extract the outlines of sugarcane images on the MATLAB platform. They compared the effects of five classic edge detection operators on the same original sugarcane image and found the Canny operator to be the most suitable and effective. Li et al. (Li et al., 2023) proposed a method that combined terahertz time-domain spectroscopy (THz-TDS) imaging technology with the K-Means image segmentation method to detect the internal quality of pumpkin seeds accurately. Their approach achieved efficient results, with average detection errors of approximately 6.27% and 4.27% for single-frequency images at spatial resolutions of 0.4 mm and 0.2 mm, respectively. Ahmed et al. (Ahmed et al., 2020) conducted a study using X-ray imaging technology to investigate three watermelon varieties' internal parameters (endosperm and air space). They evaluated traditional machine learning and deep learning methods and recognized X-ray imaging as promising.

These studies employed diverse hyperspectral image classification and feature extraction methods to accomplish seed identification. Nonetheless, traditional methods often focus on specific problems and datasets, which may limit their models' and algorithms' adaptability and generalization capabilities, warranting further improvement.

Machine learning methods provide solutions for algorithms and models to handle tasks such as seed recognition and hyperspectral image classification (Okwuashi and Ndehedehe, 2020; Chen et al., 2021a). Traditional crop seed classification and identification methods based on machine learning typically involve extracting features such as color, shape, texture, and others from images. These features are then used with classifiers like support vector machines (SVM) and artificial neural networks for classification purposes (Gao and Lim, 2019; Flores et al., 2021). For instance, Koklu et al. (Koklu and Ozkan, 2020). developed a computer vision system to differentiate seven dry bean varieties with similar characteristics. They employed image segmentation and feature extraction techniques, resulting in 16 features. By comparing the classification accuracy using 10-fold cross validation against four other methods, they found that the SVM classification model achieved the highest accuracy for bean variety classification. Su et al. (Su et al., 2020) utilized the KNCCRT integration framework and the random subspace (RS) concept to enhance diversity by randomly selecting features. They incorporated shape-adaptive (SA) neighborhood constraints within the RS integration framework to integrate spatial information. The method's effectiveness was verified through experiments on three real hyperspectral datasets. In addition, Khatri et al. (Khatri et al., 2022) employed machine learning methods to classify wheat seeds based on seven physical characteristics. They observed 92% 94%, and 92% accuracy for KNN, decision tree, and naive bayes classifiers, respectively. An ensemble classifier based on hard voting achieved a maximum accuracy of 95% for decision-making. Zhang et al. (Zhang et al., 2020) utilized a random forest classifier along with multispectral data from Landsat 8 and Gaofen-1 (GF-1), field sample data, and panchromatic data from Gaofen-2 (GF-2). They calculated a time-series vegetation index from the data's textural features and developed an RF classifier method for identifying corn seed fields. By inputting high-resolution remote sensing image features into this RF classifier, they successfully distinguished between two planting modes (seed and ordinary) and different types of corn varieties (selfing and hybrid), enabling the identification and mapping of extensive corn seed fields. Lastly, Ruslan et al. (Ruslan et al., 2022) proposed image processing and machine learning techniques were utilized to investigate the identification of weedy rice seeds. The researchers demonstrated that features extracted from RGB images, including color, morphology, and texture, exhibited higher sensitivity and accuracy compared to monochrome images.

In summary, researchers employ machine learning methods for crop seed identification as machine learning technology advances. These methods effectively identify crop seeds by extracting image features and utilizing various classifiers for classification. However, traditional machine learning methods often have high algorithm complexity and computational resource demands. This limitation hinders their real-time performance and scalability in practical applications.

Deep learning methods have made significant advancements and found widespread applications in agriculture. Researchers have utilized various methods to enhance the accuracy of image classification (Ding et al., 2020; Ding et al., 2023). These methods include the use of hybrid convolutional networks (Chen et al., 2020; Zhao et al., 2022a; Zhao et al., 2022b), innovative networks (Sun et al., 2023; Zhang et al., 2023b; Zhang et al., 2024b), improving image resolution (Paoletti et al., 2018; Liang et al., 2022), underwater image enhancement using different methods (Li et al., 2019; Li et al., 2021), multimodal deep learning models (Yao et al., 2023) and combining convolutional neural networks with hyperspectral images (Cao et al., 2020; Zheng et al., 2020; Xi et al., 2022; Yao et al., 2022). Deep learning methods address the limitations of traditional approaches by automatically learning feature representations from raw data, eliminating the need for manual feature design. They offer distinct advantages when dealing with complex and large-scale datasets.

With the ongoing development of deep learning, there is an increasing focus on applying deep learning techniques to seed classification tasks to enhance classification accuracy and robustness. For instance, Sellami et al. (Sellami et al., 2019) presented a novel approach for hyperspectral image (HSI) classification by integrating adaptive dimensionality reduction (ADR) and a semi-supervised three-dimensional convolutional neural network (3-DCNN). Their method effectively utilizes the deep spectral and spatial features extracted by convolutional encoder-decoders, substantially enhancing HSI classification accuracy. Zhang et al. (Zhang et al., 2021b) proposed a spectralspatial fractal residual convolutional neural network incorporating data balance enhancement. This method addresses the challenges posed by limited sample sizes and imbalanced categories, ultimately improving classification performance. Ahila et al. (Ahila Priyadharshini et al., 2019) developed a deep convolutional neural network based on an improved LeNet architecture to classify corn leaf diseases. By training their model on the PlantVillage dataset, they successfully classified it into four categories (three diseases and one healthy category) with an accuracy of 97.89%. Waheed et al. (Waheed et al., 2020) proposed an optimized dense convolutional network architecture for identifying and classifying corn leaf diseases. Their approach achieved an accuracy of 98.06% in accurately identifying and classifying these diseases. Furthermore, Javanmardi et al. (Javanmardi et al., 2021) proposed a novel method utilizing deep convolutional neural networks (CNN) as feature extractors. They employed multiple classifiers to classify the extracted features. Their findings demonstrated that the model trained on features extracted by CNN exhibited superior accuracy in classifying corn seed varieties, with the CNN-ANN classifier performing exceptionally well. Zhang et al. (Zhang et al., 2024a) proposed GACNet, a framework for wheat variety recognition. The framework includes semi-supervised generative adversarial networks for data augmentation and incorporates cross-conscious attention networks for variety recognition. GACNet achieves excellent classification performance through cross-learning of cascaded 3D

and 2D convolutions. Li et al. (Guohou Li et al., 2024) used a hybrid convolutional neural network based on the attention mechanism to identify varieties of hyperspectral wheat, and applied a multivariate scattering correction method to attenuate spectral differences of the same variety due to differences in scattering levels. At the same time, principal component analysis was used to reduce the unwanted spectral bands of the three-dimensional data, and the classification accuracy of this method reached 97.92%.

Deep learning technology shows excellent potential in crop seed classification tasks. These studies provide new ideas and methods for the field of seed classification.

3 Methodology

The Figure 1 illustrates the overall architecture of ERNet, designed for hyperspectral corn seed image classification. ERNet's input stage receives standardized hyperspectral maize seed images. Subsequently, the hyperspectral images undergo dimensionality reduction using the linear discriminant analysis (LDA) module. The LDA module aims to extract discriminative features by maximizing inter-class mean differences and minimizing intra-class variance. ERNet incorporates an effective residual block called the E-R module, efficiently eliminating redundant data features and addressing uneven feature extraction issues. Finally, the extracted feature information is transformed into a fully connected feature vector, and the classification result is obtained in probability form using the Softmax function. This architecture empowers ERNet to process hyperspectral corn seed images effectively, extract discriminative features, and deliver accurate classification results. The process encompasses input processing, dimensionality reduction, feature extraction, and classification output, providing a professional and effective solution for hyperspectral image-based seed classification tasks.

3.1 Network framework

The Figure 1 provides a detailed structure of ERNet, encompassing the LDA module, the efficient residual module, and the fully connected module. The LDA module framework is responsible for reducing dimensionality on hyperspectral images. The Efficient Residual module is employed to compact the network and extract precise features from hyperspectral corn seeds. Lastly, the fully connected module receives the extracted feature vector as input and employs a softmax classifier to compute class probabilities for the final classification result. Moreover, Table 1 provides a comprehensive overview of each module within the ERNet model, highlighting their respective details.

3.2 Linear discriminant analysis module

Linear discriminant analysis (LDA) is a dimensionality reduction algorithm that leverages discriminant information within a given sample set. It constructs an intra-class scatter matrix to capture the variations among similar data samples and an inter-class scatter matrix to represent the differences between dissimilar data samples (Blei et al., 2003). By identifying an optimal projection direction, LDA is designed to minimize the intra-class scatter of similar data while maximizing the inter-class scatter of dissimilar data, thus achieving optimal separability among samples (Jia et al., 2021). Specifically, LDA transforms the sample data into a feature space using linear transformations, ensuring that samples of the same pattern type are closer to each other. In contrast, samples of different patterns are pushed farther apart. This mapping enables the extraction of discriminative features, which can serve as more informative inputs for subsequent classification tasks.

Let's assume we have S training samples comprising M different pattern types, where the number of samples in each class is denoted

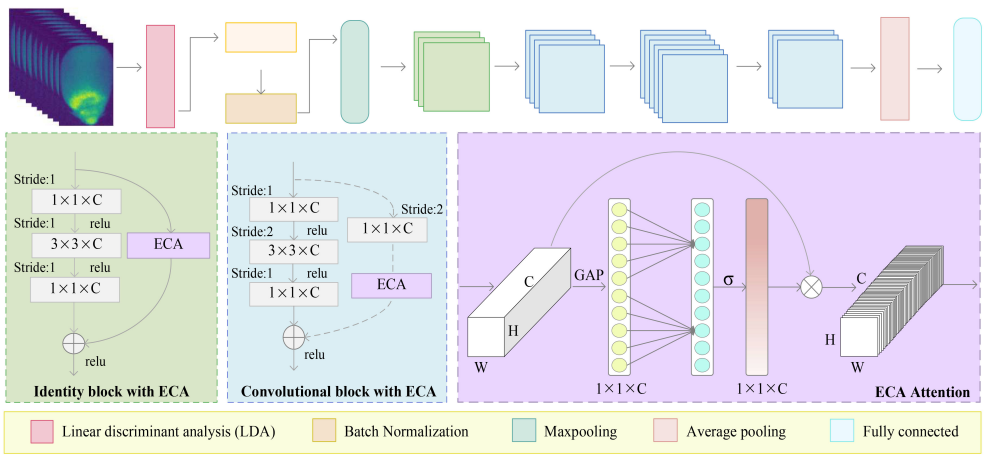


FIGURE 1
The flowchart of the ERNet method involves several steps. Initially, hyperspectral corn seed images undergo dimensionality reduction using LDA. Next, the images are subjected to convolutional operations for initial feature extraction. The texture features obtained are then refined using the efficient residual module to enhance their quality for the corn seed identification task.

TABLE 1 Details of each module of ERNet.

Layers (type)	Input Size	Output Size	Repeat	Parameter
Input	128 × 224 × 224	128 × 224 × 224	1	0
LDA	128 × 224 × 224	3 × 224 × 224	1	1280
Conv	3 × 224 × 224	64 × 112 × 112	1	9536
MaxPooling_1	64 × 112 × 112	64 × 56 × 56	1	0
IBE	64 × 56 × 56	256 × 56 × 56	3	232201
CBE_1	256 × 56 × 56	512 × 28 × 28	4	1086911
CBE_2	512 × 28 × 28	1024 × 14 × 14	6	7098422
CBE_3	1024 × 14 × 14	2048 × 7 × 7	3	14964763
Averagepooling	2048 × 7 × 7	2048 × 1 × 1	1	0
Fullyconnected	2048 × 1 × 1	2048 × 1 × 1	1	20490
Total trainable parameter:23413603				

as $\mathbf{S}_i(i = 1,2,\cdots, M)$. Class M is represented by $x_i = \left\{x_{i_1}, x_i, \cdots, x_{i_{S_i}}\right\}$, $x_{ij}(i = 1, 2, \cdots, M; j = 1, 2, \cdots, S_i)$ is an n dimensional vector.

Consequently, we can compute the mean vector for each pattern type as Equation (1):

$$v_i = \frac{1}{S_i}\sum_{j=1}^{S_i} x_{ij}, \tag{1}$$

the total sample mean vector is Equation (2):

$$v = \frac{1}{M}\sum_{i=1}^{M} v_i\,. \tag{2}$$

The intra-class scattering matrix $\mathbf{T}_W$ and inter-class scattering matrix $\mathbf{T}_B$ are respectively expressed as Equations (3) and (4):

$$T_W = \sum_{i=1}^{M}\sum_{j=1}^{S_i}(x_{ij} - v_i)(x_{ij} - v_i)^T, \tag{3}$$

$$T_B = \sum_{i=1}^{M}(v_i - v)(v_i - v)^T, \tag{4}$$

for any n-dimensional vector a, the function $f = \frac{a^T T_B a}{a^T T_W a}$ can be calculated. The function measures the linear separability between different pattern types by evaluating the ratio of the differences between dissimilar categories to the differences between similar types. A larger f value indicates a stronger linear separability, implying a higher discriminative power in distinguishing between different modes.

LDA effectively reduces data dimensionality while preserving the discriminative information between categories. By incorporating the LDA module, the classification performance in hyperspectral image seed classification can be enhanced, and redundant features can be minimized.

3.3 Efficient channel attention module

Studies have revealed that the channel attention mechanism effectively enhances the performance of neural networks (Shi et al., 2022). However, existing attention modules often exhibit complexity, which can lead to the problem of model overfitting. To tackle this problem, Wang et al (Wang et al., 2020) proposed a lightweight and versatile module called efficient channel attention (ECA). This study incorporates the ECA module into ERNet to assign channel weights to capture crucial features of hyperspectral corn seeds. Introducing the ECA module into ERNet enhances network performance and augments the ability to represent important features of hyperspectral corn seeds.

The Figure 2 demonstrates the operational principle of the ECA channel attention mechanism. Global average pooling (GAP) is initially applied to the original input image to extract its features. This process involves averaging the features across each channel. Subsequently, the ECA module facilitates local cross-channel interactions through a rapid one-dimensional convolution operation, employing a kernel size denoted as k. Determining the convolution kernel's size, k is adaptively achieved by leveraging a function that the number of input channels C. Following this, the sigmoid function is employed to assign weight proportions to each channel. These weights represent the significance of each channel in feature representation. Finally, the original input features are element-wise multiplied by the channel weights, resulting in a feature representation incorporating channel attention. Through

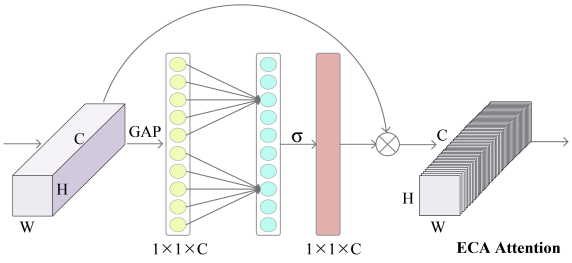


FIGURE 2
Feature refinement process of the efficient channel attention modules.

these operations, the ECA module enables the network to prioritize crucial channels and extract discriminative image features.

In the ECA attention mechanism, the first step is to transform the dimension of the feature map obtained after the residual network and pooling at each scale. The feature map, initially in the shape of [H, W, C], is transformed into a vector of [1, 1, C]. Subsequently, the adaptive one-dimensional convolution kernel size, denoted as k, is calculated based on the channel number C of the feature map. Calculated as in Equation (5):

$$k = \varphi(C) = \left| \frac{\log_2 C}{\gamma} + \frac{b}{\gamma} \right|_{odd}, \quad (5)$$

where γ takes value of 2, b takes value of 1, and odd takes odd number. The adaptive convolution kernel size, denoted as k, is calculated based on these values. The calculated k is then used for the one-dimensional convolution operation, which is applied to each channel of the feature map. The purpose of this operation is to capture the interactive information and reduce the degree of information loss between channels. Subsequently, the weights of each channel in the feature map are determined using the sigmoid function. The resulting consequences are then normalized, and the original input feature map is multiplied element-wise with the normalized weights to obtain the weighted feature map. This operation enables the network to prioritize essential channels, enhancing the features' representation capabilities.

3.4 Efficient residual module

The shortcut connections have been introduced into the residual network to facilitate optimization. A shortcut connection is a network structure that spans one or more layers and forms a residual learning unit by adding the input directly to the output. As depicted in the Figure 3, assuming the model input is denoted as x, and the original mapping as $R(x)$, the core idea of residual learning is to design the network as $R(x) = f(x) + x$, where $f(x)$ represents the residual mapping. A residual map $f(x) + x$ is obtained by adding the residual map to the input. Although both mappings achieve the same expression effect, the residual map $f(x)$ scale is relatively minor. Fitting $f(x)$ is much simpler than doing the entire $R(x)$ map. Replacing all the original mappings $R(x)$ in the model with the residual mapping $f(x) + x$, reduces the difficulty of model fitting. The shortcut connections enable the network to learn the residual part more efficiently without excessively emphasizing the original mapping. This design more accessible training and optimization of the network, thereby enhancing the model's performance and generalization ability.

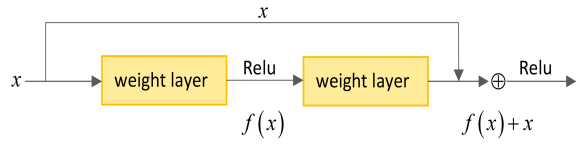


FIGURE 3
Basic unit legend of residual network.

In the context of the residual network, the output of each residual learning unit is denoted as y_i, while the input is represented as x_i. The mapping relationship within the residual learning unit can be expressed as Equation (6):

$$y_i = f[x_i + H(x_i, w_i)], \quad (6)$$

where the activation function f is applied within the residual learning unit. The rectified linear unit (ReLU) and the sigmoid function commonly use activation functions in neural networks. The term $H(x_i, w_i)$ represents the residual, where w_i represents the convolution kernel.

In the context of hyperspectral corn seed images, each image can be represented as a matrix M consisting of multiple column vectors. Matrix multiplication corresponds to a transformation, where a vector undergoes operations such as rotation or scaling to yield a new vector. When a matrix solely performs scaling or scales one or more vectors without introducing a rotational effect, these vectors are referred to as eigenvectors of the matrix, and the scaled value is known as the eigenvalue. Using methods such as gradient descent, the eigenvectors and eigenvalues of the matrix can be reversely fitted. These eigenvectors and eigenvalues are the characteristic information of hyperspectral corn seed images. We can classify and identify ideas by extracting and utilizing this feature information. The above transformation is formulated as Equation (7):

$$M\ (X) = M\ (x) * T\ (y), \quad (7)$$

where $M(X)$ represents the matrix obtained after scaling transformation, $M(x)$ denotes the original matrix, and x represents the column vector of the original matrix. $T(y)$ represents a scaling matrix, where y signifies the scaling ratio applied to the column vector x within the matrix M.

The Figure 4 illustrates the efficient residual (ER) module, constructed by combining the IBE and CBE units. The number of stacks is determined through multiple tests, with the IBE module being stacked three times and the CBE module being repeated three times. Do 4, 6, and 3 stacks, respectively. Specifically, the basic unit comprises a sequence of cascaded operations, including convolution, batch normalization, activation function, convolution, batch normalization, activation function, convolution, batch normalization, and an ECA attention module. The pixel-by-pixel addition operation is employed within the basic unit. Additionally, short-circuit connections are incorporated within the basic unit to mitigate gradient vanishing issues and prevent network degradation.

The basic unit within the ER module enhances the network's representation capabilities and improves the training process. It achieves this through cascaded convolution and batch normalization operations. The convolution operations aid in extracting essential image feature information and expand the network's depth and receptive field, enhancing the model's expressive ability. The batch normalization operation accelerates training and enhances the model's robustness. Moreover, the activation functions introduce nonlinearity, allowing the network to capture complex relationships within the data.

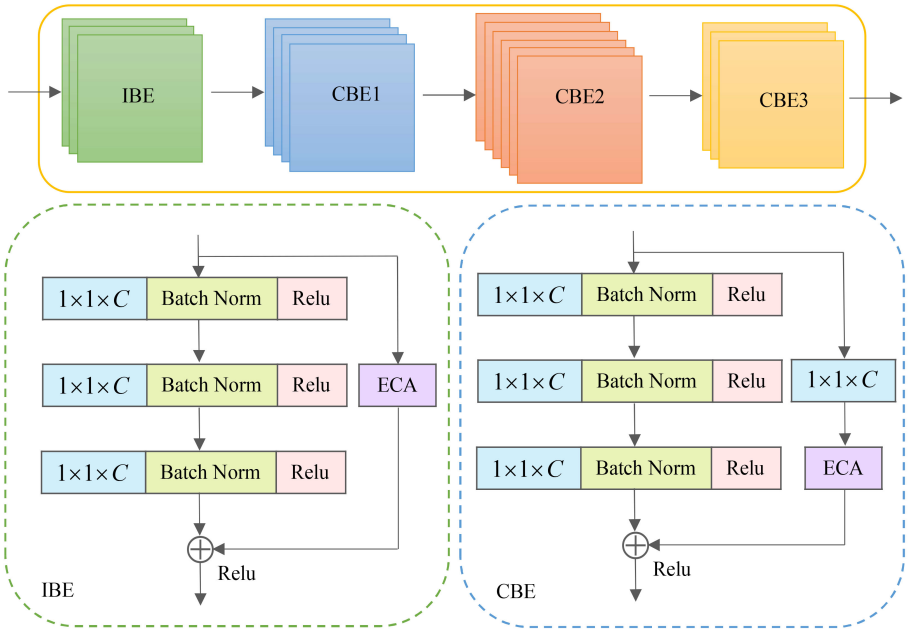


FIGURE 4
Feature refinement process of the efficient residual modules.

To enhance the original model and improve its performance in image processing and computer vision tasks, we propose integrating the ECA module with the identity and convolutional blocks, respectively and refer to them as IBE and CBE structures. When the number of input and output channels of the essence or convolutional residual blocks is the same, we can directly incorporate an element-wise shortcut link by adding the input and output. This configuration forms the IBE structure. However, when the number of input and output channels differs in the identity or convolutional residual blocks, we introduce a convolution layer in the shortcut connection. This additional layer adjusts the dimension of the feature map to accommodate the disparity in input and output channel numbers, resulting in the formation of the CBE structure. We aim to enhance the original model and improve its performance in various image processing and computer vision tasks by employing these IBE and CBE structures.

3.5 Loss function

Cross entropy(CE) loss is a frequently used loss function in deep learning, especially in multiclassification problems. It draws upon concepts from information theory and measures the proximity between the actual output and the desired output. In information theory, the CE is utilized to estimate the average code length. In the context of deep learning, the CE loss function quantifies the dissimilarity between the model's output's probability distribution and the actual label's probability distribution. A smaller CE value indicates a closer match between the two probability distributions. Given two probability distributions, $P_A(x)$ and $P_B(x)$, the CE between them can be expressed as Equation (8):

$$\begin{aligned} H(A,B) &= \sum_{i=1}^{n} P_A(x_i) \log \frac{1}{P_B(x_i)} \\ &= -\sum_{i=1}^{n} P_A(x_i) \log P_B(x_i), \end{aligned} \tag{8}$$

where $P_A(x)$ represents true label distribution in the given expression, while $P_B(x)$ represents the predicted distribution. As a measure, the CE quantifies the disparity between the expected value and the actual label value. More precisely, the CE loss function gauges the uncertainty of the predicted distribution about the actual distribution. To measure the distance and dissimilarity between two probability distributions, kullback-leibler divergence (KL-divergence) is employed. The KL-divergence is represented as Equation (9):

$$\begin{aligned} D_{KL}(A \parallel B) &= \sum_{i=1}^{n} P_A(x_i) \log \frac{P_A(x_i)}{P_B(x_i)} \\ &= \sum_{i=1}^{n} P_A(x_i) \log P_A(x_i) - \sum_{i=1}^{n} P_A(x_i) \log P_B(x_i) \\ &= -H(A) + H(A,B), \end{aligned} \tag{9}$$

where $D_{KL}(A \parallel B)$ achieves its minimum value only when $P_A(x)$=$P_B(x)$, indicating that the closer the predicted result is to the actual result better. The CE loss function is a specific instance of KL-divergence and finds extensive application in deep learning's multi-classification problems. We aim to minimize the CE loss function to make the predicted $P_B(x)$ as similar as possible to the actual label distribution $P_A(x)$. This alignment ensures that the model's predictions are consistent with the results.

The multi-class CE loss function serves as the evaluation criterion for the model. The network aims to minimize the CE by updating the weights of its nodes. To achieve this, the model

employs the stochastic gradient descent algorithm. This algorithm optimizes the loss function to determine the optimal parameters and minimize the loss. The optimization process of the stochastic gradient descent algorithm can be defined as Equation (10):

$$\theta_i = \theta_j - \alpha * \frac{\partial J(\theta)}{\partial \theta}, \tag{10}$$

where θ_i represents the weight of the current network node, θ_j denotes the weight from the previous iteration of the network, and α represents the learning rate of the model. During each iteration, the model optimizes θ through gradient descent, aiming to minimize the CE. The goal is to reach the lowest possible the CE, enabling the entire model to converge toward the global optimal solution.

4 Experiments

This chapter begins by introducing the dataset utilized in the study. It then proceeds to describe the training process of ERNet, followed by conducting comparative and ablation experiments to demonstrate the significance of ERNet in hyperspectral corn seed classification. The results obtained from these experiments provide valuable insights and reference points for evaluating the effectiveness of ERNet in the classification task.

4.1 Dataset used

We method run on a Windows 10 PC with AMD Ryzen 5 3600X Central Processing Unit (CPU) at 3.80 GHz, The dataset (CSHID) utilized in this article is sourced from SSTNet (Zhang et al., 2022), encompassing ten different corn varieties cultivated in Henan Province: Baiyu 607, Baiyu 808, Baiyu 818, Baiyu 833, Baiyu 879, Baiyu 897, Baiyu 918, Baiyu 8317, Baiyu 9284, and Fengda 601. The data was collected using Surface Optics' SOC 710 Portable Visible/Near Infrared Imaging Spectrometer. Each corn variety consists of 120 samples, with each piece containing 128 spectral bands. The original spectra were precisely cropped to ensure accuracy, resulting in 129,230 sample images employed in this study. The Figure 5 showcases a comparison of selected images before and after cropping.

4.2 Experimental settings

The experimental setup for this article consisted of a computer equipped with an AMD Ryzen 7 5800H with Radeon Graphics CPU, operating at 3.20GHz and 16GB of RAM. Additionally, it included an NVIDIA GeForce GTX 1650 graphics card with 4GB of video memory. The software environment for the experiments involved Python 3.7.13 and torch-gpu-1.10.1, running on the 64-bit Windows 11 operating system.

The fully connected layer incorporates dropout technology to prevent overfitting during model training. Additionally, the model's parameters are optimized using the Adam optimizer. Classification results determine a learning rate of 0.01 as the optimal choice. Furthermore, an exponential decay learning rate enhances model stability during later training. This approach gradually reduces the learning rate over time. The training process follows a batch training method with a batch size of 32. Batch training involves dividing the training dataset into several batches, each containing a specific number of samples. The model performs forward propagation and back propagation calculations on each set to update the parameters. After 150 iterations, the loss rate stabilizes, indicating that the model has converged and achieved relatively stable performance.

4.3 Identification evaluation

When it comes to deep learning, more data is often required for practical training than traditional machine learning approaches. This paper randomly divides the dataset into a training set and a test

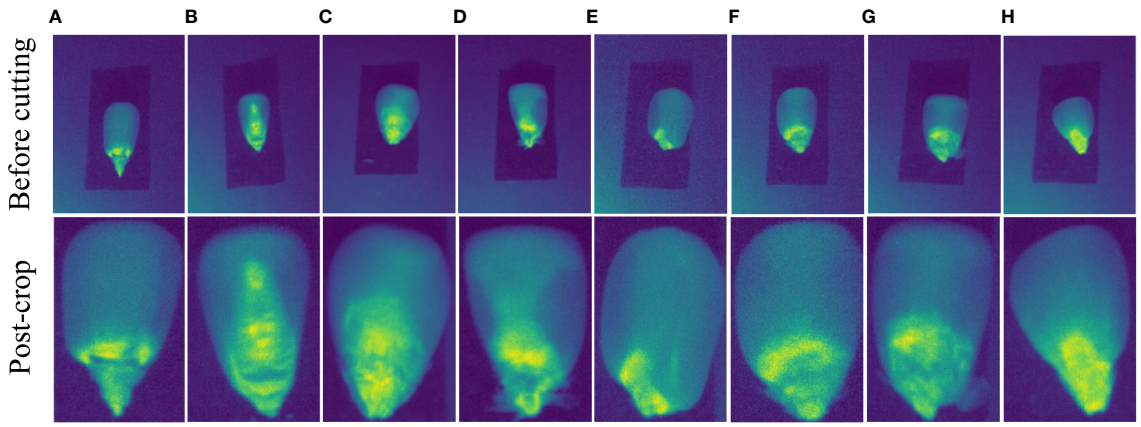


FIGURE 5
The provided images showcase different types of seed images. The pictures labeled "before cutting" are sourced from SSTNet, while those labeled "after cropping" have been manually and accurately cropped. The above eight sets of images correspond to the following corn varieties: **(A)** baiyu 607, **(B)** baiyu 808, **(C)** baiyu 818, **(D)** baiyu 833, **(E)** baiyu 8317, **(F)** baiyu 9284, **(G)** baiyu 897, and **(H)** fengda 601.

set following a "training set: test set = 4:1" principle. Four machine learning and six deep learning models are selected as reference models to conduct comparative experiments. The machine learning models consist of fuzzy k-nearest neighbor (FKNN) (Kumbure et al., 2020), random forest algorithm (RFA) (Chen et al., 2021b), stochastic gradient descent (SGD) (Lei and Tang, 2021), and spatial-spectral feature extraction method (FSVM) (Jin et al., 2022). The deep learning models include hybrid spectral net (HybridSN) (Roy et al., 2019), centernet (Jin et al., 2021), spatial source phase net (SSPNet) (Lin et al., 2022), spatial, spectral, and texture aware attention network (SSTNet) (Zhang et al., 2022), convolutional neural network with a bidirectional gated recurrent unit (CNN-BiGRU) (Lu et al., 2023), and Convolutional Neural networks with long short-term memory (CNN-LSTM) (Wang and Song, 2023).

Model performance is assessed using four metrics: F1 score, recall, precision, and accuracy. Accuracy measures the correct classification rate of both positive and negative samples. Precision is the ratio of true positives to all positive classifications. Recall measures the percentage of correctly classified positive models out of all positive examples. The F1 score is a comprehensive evaluation index that combines precision and recall. Higher values of these metrics indicate better classification performance. By comparing the metric results across different models, their effectiveness in classification tasks can be evaluated. We used the same test sets and training parameters in comparison tests to assess ERNet against several different approaches. The outcomes, as reported in Table 2.

RFA (Chen et al., 2021b) employs the random forest algorithm to compute variable importance and weights for security risk indicators, demonstrating high accuracy on large-scale datasets. FSVM (Jin et al., 2022) utilizes principal component analysis to extract features from spatial-spectral data and trains and optimizes the model using support vector machines, resulting in good classification performance on small sample datasets. FKNN (Kumbure et al., 2020) utilizes local mean vectors and Bonferroni means, showcasing strong performance despite significantly imbalanced data class distributions. SGD (Lei and Tang, 2021) introduces high-probability bounds on computational and statistical errors, enabling the development of a new learning rate for non-convex learning with SGD by adjusting the number of passes to balance these errors. SSPNet (Lin et al., 2022) utilizes spatial source phase (SSP) maps derived from complex-valued fMRI data as input for CNN and achieves noteworthy results in image recognition. HybridSN (Roy et al., 2019) and SSTNet (Zhang et al., 2022) are hybrid CNN models that jointly leverage 3D-CNN to represent spatial-spectral features from spectral bands. SSTNet additionally incorporates a spatial channel attention mechanism. Both methods deliver satisfactory performance in hyperspectral image classification. CenterNet (Jin et al., 2021) combines deep learning and image processing techniques, utilizing genetic algorithms to determine indicators and evaluate results, resulting in commendable classification performance. CNN-BiGRU (Lu et al., 2023) combines a convolutional neural network with a bidirectional gated recurrent unit, introducing residual mechanisms and an improved convolutional attention module, demonstrating promising outcomes in rice disease identification. CNN-LSTM (Wang and Song, 2023) combines a convolutional neural network (CNN) with a long short-term memory (LSTM) network and achieves accurate identification of corn varieties in conjunction with hyperspectral imaging technology. Nevertheless, the classification results obtained by these traditional and deep learning methods still lower than ERNet.

Table 2 makes it clear that when compared to other techniques, the machine learning models RFA (Chen et al., 2021b) and FSVM (Jin et al., 2022) perform worse in classification. RFA (Chen et al., 2021b) and FSVM (Jin et al., 2022) perform somewhat worse in classification than FKNN (Kumbure et al., 2020) and SGD (Lei and Tang, 2021). Although deep learning techniques like CNN-BiGRU (Lu et al., 2023), CNN-LSIM (Wang and Song, 2023), and CenterNet (Jin et al., 2021) have considerable benefits, their classification

TABLE 2 Identification results of different deep learning methods tested on the CSHID dataset.

Method	F1-score	Recall	Precision	Accuracy
FKNN (Kumbure et al., 2020)	0.9610	0.9583	0.9637	0.9625
RFA (Chen et al., 2021b)	0.9426	0.9411	0.9435	0.9479
SGD (Lei and Tang, 2021)	0.9585	0.9663	0.9703	0.9705
FSVM (Jin et al., 2022)	0.9467	0.9417	0.9519	0.9458
HybridSN (Roy et al., 2019)	0.9621	0.9667	0.9673	0.9708
CenterNet (Jin et al., 2021)	0.9253	0.9255	0.9260	0.9258
SSPNet (Lin et al., 2022)	0.9695	0.9695	0.9696	0.9693
SSTNet (Zhang et al., 2022)	0.9795	0.9791	0.9800	0.9792
CNN-BiGRU (Lu et al., 2023)	0.9396	0.9385	0.9384	0.9393
CNN-LSTM (Wang and Song, 2023)	0.9497	0.9501	0.9534	0.9509
ERNet	0.9833	0.9830	0.9846	0.9836

(Optimal: red; Suboptimal: blue).

performance isn't perfect. HybridSN (Roy et al., 2019), SSPNet (Lin et al., 2022), and SSTNet (Zhang et al., 2022) do not outperform our ERNet on a variety of indicators, even though they take into account spectral spatial information and perform well in classification. In conclusion, our ERNet performs exceptionally well in classification due to its superiority in picture feature extraction. Regarding overall performance, the ERNet network demonstrates notable advantages across all evaluation metrics. The accuracy achieved by the ERNet network reaches an impressive 98.36%. The accuracy improvement ranges from 1.31% to 3.78% compared to the machine learning models. Similarly, the other deep learning models show accuracy gains ranging from 0.44% to 3.27%. These results highlight the significant enhancement in corn hyperspectral image classification accomplished by the ERNet network.

The accuracy and loss convergence curves of ERNet during testing are shown in The Figures 6A, B. It is clear that ERNet exhibits faster convergence before 20 epochs, and by the 130th epoch, it has achieved good convergence and high accuracy. After more than 130 epochs, ERNet stabilizes.

Comparing the training time of each model on the CSHID dataset, it can be seen from Table 3 that ERNet outperforms traditional machine learning with the latest network models for training hyperspectral images in terms of training time, which shows that the ERNet model achieves an excellent balance between efficiency of use and improvement in accuracy, specifically through the advantages of combining residual networks with lightweight attention mechanisms to achieve network performance improvement.

TABLE 3 Training and testing times on the CSHID dataset, training times are based on one epoch.

Method	Training time(s)	Testing time(min)
FKNN (Kumbure et al., 2020)	25.7	20.3
RFA (Chen et al., 2021b)	35.4	28.4
SGD (Lei and Tang, 2021)	26.0	28.3
FSVM (Jin et al., 2022)	34.6	27.8
HybridSN (Roy et al., 2019)	37.1	29.7
CenterNet (Jin et al., 2021)	46.6	31.8
SSPNet (Lin et al., 2022)	37.6	32.4
SSTNet (Zhang et al., 2022)	26.4	29.7
CNN-BiGRU (Lu et al., 2023)	26.2	23.4
CNN-LSTM (Wang and Song, 2023)	35.9	26.8
ERNet	**20.3**	**19.6**

The test time is the result of categorizing the entire dataset. (Optimal results are bolded.)

4.4 Ablation study

Ablation experiments were performed to evaluate the effectiveness of each module in ERNet for hyperspectral maize seed detection. The following ablation operations were performed on ERNet individually: 1) our ERNet without efficient channel attention module (-w/o ECA); 2) our ERNet without convolutional block-ECA (-w/o CBE); 3) our ERNet without identity block-ECA (-w/o IBE). The ablation experiments enabled a thorough evaluation of the effect of each module on the performance of ERNet in recognizing hyperspectral corn seeds.

Table 4 presents the f1-score, recall, and accuracy results for each ablation experimental model and the corresponding accuracy score for the full ERNet model. By comparing the practical outcomes, it is evident that the complete ERNet model achieved the highest scores across all metrics compared to the ablation models.

5 Discussion

The research presented in this article holds significant importance for corn seed identification within the agricultural domain. By leveraging an efficient residual network to process high-dimensional hyperspectral image data, the accuracy and efficiency of corn seed identification can be substantially enhanced. This, in turn, enables precise planting management and facilitates

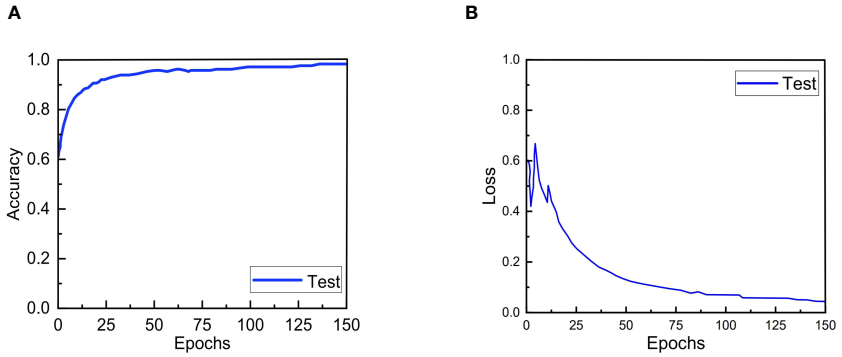


FIGURE 6
Accuracy and loss convergence over the number of epochs on the test set. **(A)** Accuracy convergence over the number of epochs. **(B)** Loss convergence over the number of epochs.

TABLE 4 Results of different modules for the implementation of ablation studies on test samples may exhibit discriminatory tendencies.

Method	F1-score	Recall	Precision	Accuracy
-w/o ECA	0.9570	0.9578	0.9572	0.9571
-w/o CBE	0.9354	0.9341	0.9343	0.9351
-w/o IBE	0.6432	0.6387	0.6359	0.6328
ERNet (full model)	0.9833	0.9830	0.9846	0.9836

(Optimal: red; Suboptimal: blue).

advancements in crop varieties for agricultural production. The intended outcome of this research is to demonstrate experimentally that features within hyperspectral images can be effectively extracted using an efficient residual network, leading to accurate classification and identification of corn seeds. Furthermore, this article's research methods and findings can serve as a valuable reference for studying hyperspectral image recognition and classification in other crop-related research endeavors.

Future challenges include realizing complete seed screening in the recognition process and using hyperspectral technology for maturity discrimination to achieve a true sense of superior breed recognition. These challenges are worthwhile to pursue in order to develop more functional deep learning models for seed recognition in a variety of 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

XYL: Writing – review & editing. MJZ: Writing – original draft, Writing – review & editing. LYZ: Writing – review & editing. LZ: Formal analysis, Writing – review & editing. XWX: Writing – review & editing. WYZ: Data curation, Writing – review & editing. WDZ: Methodology, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported in part by the Natural Science Foundation of Henan Province under Grant 232300420428, in part by the Teacher Education Curriculum Reform Research of Henan Province under Grant 2024-JSJYYB-099, in part by the Key Specialized Research and Development Program of Science and Technology of Henan Province under Grants 232102210018, 232102211044, and in part by the Teacher Education Curriculum Reform Research of Henan Institute of Science and Technology under Grant 2024JSJY04.

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

Ahila Priyadharshini, R., Arivazhagan, S., Arun, M., and Mirnalini, A. (2019). Maize leaf disease classification using deep convolutional neural networks. *Neural Computing Appl.* 31, 8887–8895. doi: 10.1007/s00521-019-04228-3

Ahmad, M., Shabbir, S., Roy, S. K., Hong, D., Wu, X., Yao, J., et al. (2021). Hyperspectral image classification—traditional to deep models: A survey for future prospects. *IEEE J. select. top. Appl. Earth observat. Remote Sens.* 15, 968–999. doi: 10.1109/JSTARS.2021.3133021

Ahmed, M. R., Yasmin, J., Wakholi, C., Mukasa, P., and Cho, B.-K. (2020). Classification of pepper seed quality based on internal structure using x-ray ct imaging. *Comput. Electron. Agric.* 179, 105839. doi: 10.1016/j.compag.2020.105839

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.* 3, 993–1022.

Cao, X., Yao, J., Xu, Z., and Meng, D. (2020). Hyperspectral image classification with convolutional neural network and active learning. *IEEE Trans. Geosci. Remote Sens.* 58, 4604–4616. doi: 10.1109/TGRS.36

Chen, J., Chen, J., Zhang, D., Sun, Y., and Nanehkaran, Y. A. (2020). Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* 173, 105393. doi: 10.1016/j.compag.2020.105393

Chen, M., Wang, J., Zhu, L., Deng, H., Wu, H., and Zhang, Z. (2023). "Research on contour extraction and edge detection of sugarcane image based on matlab," in *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)*. (IEEE), 1784–1787. doi: 10.1109/ICPECA56706.2023.10076152

Chen, J., Zhang, D., Suzauddola, M., and Zeb, A. (2021a). Identifying crop diseases using attention embedded mobilenet-v2 model. *Appl. Soft Computing* 113, 107901. doi: 10.1016/j.asoc.2021.107901

Chen, Y., Zheng, W., Li, W., and Huang, Y. (2021b). Large group activity security risk assessment and risk early warning based on random forest algorithm. *Pattern Recognit. Lett.* 144, 1–5. doi: 10.1016/j.patrec.2021.01.008

Ding, C., Luo, Z., Hou, Y., Chen, S., and Zhang, W. (2023). An effective method of infrared maritime target enhancement and detection with multiple maritime scene. *Remote Sens.* 15, 3623. doi: 10.3390/rs15143623

Ding, C., Pan, X., Gao, X., Ning, L., and Wu, Z. (2020). Three adaptive sub-histograms equalization algorithm for maritime image enhancement. *IEEE Access* 8, 147983–147994. doi: 10.1109/Access.6287639

ElMasry, G., Mandour, N., Wagner, M.-H., Demilly, D., Verdier, J., Belin, E., et al. (2019). Utilization of computer vision and multispectral imaging techniques for classification of cowpea (vigna unguiculata) seeds. *Plant Methods* 15, 1–16. doi: 10.1186/s13007-019-0411-2

Flores, P., Zhang, Z., Igathinathane, C., Jithin, M., Naik, D., Stenger, J., et al. (2021). Distinguishing seedling volunteer corn from soybean through greenhouse color, color-infrared, and fused images using machine and deep learning. *Ind. Crops Products* 161, 113223. doi: 10.1016/j.indcrop.2020.113223

Gan, L., Xia, J., Du, P., and Chanussot, J. (2018). Multiple feature kernel sparse representation classifier for hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* 56, 5343–5356. doi: 10.1109/TGRS.36

Gao, Q., and Lim, S. (2019). A probabilistic fusion of a support vector machine and a joint sparsity model for hyperspectral imagery classification. *GISci. Remote Sens.* 56, 1129–1147. doi: 10.1080/15481603.2019.1623003

Ghaderizadeh, S., Abbasi-Moghadam, D., Sharifi, A., Tariq, A., and Qin, S. (2022). Multiscale dualbranch residual spectral–spatial network with attention for hyperspectral image classification. *IEEE J. Select. Top. Appl. Earth Observat. Remote Sens.* 15, 5455–5467. doi: 10.1109/JSTARS.2022.3188732

Guohou Li, Z. L., Songlin Jin, W. Z., Xipeng Pan, Z. L., and Li Qin, W. Z. (2024). Mix convolutional neural networks for hyperspectral wheat variety discrimination. *Spectrosc. Spectral Anal.* 44, 807. doi: 10.3964/j.issn.1000-0593(2024)03-0807-07

Hu, X., Yang, L., and Zhang, Z. (2020). Non-destructive identification of single hard seed via multispectral imaging analysis in six legume species. *Plant Methods* 16, 1–13. doi: 10.1186/s13007-020-00659-5

Huang, B., Wang, Z., Shang, J., Chen, G., and Radenkovic, M. (2022). A spectral sequence-based nonlocal long short-term memory network for hyperspectral image classification. *IEEE J. Select. Top. Appl. Earth Observat. Remote Sens.* 15, 3041–3051. doi: 10.1109/JSTARS.2022.3159729

Javanmardi, S., Ashtiani, S.-H. M., Verbeek, F. J., and Martynenko, A. (2021). Computer-vision classification of corn seed varieties using deep convolutional neural network. *J. Stored Products Res.* 92, 101800. doi: 10.1016/j.jspr.2021.101800

Jia, S., Zhao, Q., Zhuang, J., Tang, D., Long, Y., Xu, M., et al. (2021). Flexible gabor-based superpixel-level unsupervised lda for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 59, 10394–10409. doi: 10.1109/TGRS.2020.3048994

Jin, X., Che, J., and Chen, Y. (2021). Weed identification using deep learning and image processing in vegetable plantation. *IEEE Access* 9, 10940–10950. doi: 10.1109/Access.6287639

Jin, S., Zhang, W., Yang, P., Zheng, Y., An, J., Zhang, Z., et al. (2022). Spatial-spectral feature extraction of hyperspectral images for wheat seed identification. *Comput. Electrical Eng.* 101, 108077. doi: 10.1016/j.compeleceng.2022.108077

Khatri, A., Agrawal, S., and Chatterjee, J. M. (2022). "Wheat seed classification: utilizing ensemble machine learning approach," in *Scientific programming*, vol. 2022. Hindawi.

Koklu, M., and Ozkan, I. A. (2020). Multiclass classification of dry beans using computer vision and machine learning techniques. *Comput. Electron. Agric.* 174, 105507. doi: 10.1016/j.compag.2020.105507

Kumbure, M. M., Luukka, P., and Collan, M. (2020). A new fuzzy k-nearest neighbor classifier based on the bonferroni mean. *Pattern Recognit. Lett.* 140, 172–178. doi: 10.1016/j.patrec.2020.10.005

Lei, Y., and Tang, K. (2021). Learning rates for stochastic gradient descent with nonconvex objectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 4505–4511. doi: 10.1109/TPAMI.2021.3068154

Li, C., Anwar, S., Hou, J., Cong, R., Guo, C., and Ren, W. (2021). 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, 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.83

Li, B., Sun, Z.-x., Yang, A.-k., and Liu, Y.-d. (2023). Study on detection of the internal quality of pumpkin seeds based on terahertz imaging technology. *J. Food Measurement Characterizat.* 17, 1576–1585. doi: 10.1007/s11694-022-01727-1

Liang, Z., Zhang, W., Ruan, R., Zhuang, P., and Li, C. (2022). Gifm: An image restoration method with generalized image formation model for poor visible conditions. *IEEE Trans. Geosci. Remote Sens.* 60, 1–16. doi: 10.1109/TGRS.2022.3227548

Lin, Q.-H., Niu, Y.-W., Sui, J., Zhao, W.-D., Zhuo, C., and Calhoun, V. D. (2022). *Sspnet: An interpretable 3d-cnn for classification of schizophrenia using phase maps of resting-state complex-valued fmri data* vol. 79. (Elsevier), 102430. doi: 10.1016/j.media.2022.102430

Lu, Y., Wu, X., Liu, P., Li, H., and Liu, W. (2023). Rice disease identification method based on improved cnn-bigru. *Artif. Intell. Agric.* 9, 100–109. doi: 10.1016/j.aiia.2023.08.005

Ma, T., Tsuchikawa, S., and Inagaki, T. (2020). Rapid and non-destructive seed viability prediction using near-infrared hyperspectral imaging coupled with a deep learning approach. *Comput. Electron. Agric.* 177, 105683. doi: 10.1016/j.compag.2020.105683

Okwuashi, O., and Ndehedehe, C. E. (2020). Deep support vector machine for hyperspectral image classification. *Pattern Recognit.* 103, 107298. doi: 10.1016/j.patcog.2020.107298

Paoletti, M. E., Haut, J. M., Fernandez-Beltran, R., Plaza, J., Plaza, A. J., and Pla, F. (2018). Deep pyramidal residual networks for spectral–spatial hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 57, 740–754. doi: 10.1109/TGRS.2018.2860125

Roy, S. K., Krishna, G., Dubey, S. R., and Chaudhuri, B. B. (2019). Hybridsn: Exploring 3-d– 2-d cnn feature hierarchy for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* 17, 277–281. doi: 10.1109/LGRS.2019.2918719

Ruslan, R., Khairunniza-Bejo, S., Jahari, M., and Ibrahim, M. F. (2022). Weedy rice classification using image processing and a machine learning approach. *Agriculture* 12, 645. doi: 10.3390/agriculture12050645

Sellami, A., Farah, M., Farah, I. R., and Solaiman, B. (2019). Hyperspectral imagery classification based on semi-supervised 3-d deep neural network and adaptive band selection. *Expert Syst. Appl.* 129, 246–259. doi: 10.1016/j.eswa.2019.04.006

Shi, Y., Wang, Z., Du, X., Ling, G., Jia, W., and Lu, Y. (2022). Research on the membrane fouling diagnosis of mbr membrane module based on eca-cnn. *J. Environ. Chem. Eng.* 10, 107649. doi: 10.1016/j.jece.2022.107649

Su, H., Yu, Y., Wu, Z., and Du, Q. (2020). Random subspace-based k-nearest class collaborative representation for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 59, 6840–6853. doi: 10.1109/TGRS.2020.3029578

Sun, X., Li, Y., Li, G., Jin, S., Zhao, W., Liang, Z., et al. (2023). Scgnet: Efficient sparsely connected group convolution network for wheat grains classification. *Front. Plant Sci.* 14, 1304962. doi: 10.3389/fpls.2023.1304962

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassanien, A. E., and Pandey, H. M. (2020). An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. *Comput. Electron. Agric.* 175, 105456. doi: 10.1016/j.compag.2020.105456

Wang, Y., and Song, S. (2023). Variety identification of sweet maize seeds based on hyperspectral imaging combined with deep learning. *Infrared Phys. Technol.* 130, 104611. doi: 10.1016/j.infrared.2023.104611

Wang, X., Tan, K., Du, Q., Chen, Y., and Du, P. (2019). Caps-triplegan: Gan-assisted capsnet for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 57, 7232–7245. doi: 10.1109/TGRS.2019.2912468

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q. (2020). "Eca-net: Efficient channel attention for deep convolutional neural networks," *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.*, 11534–11542. doi: 10.1109/CVPR42600.2020

Xi, B., Li, J., Li, Y., Song, R., Hong, D., and Chanussot, J. (2022). Few-shot learning with classcovariance metric for hyperspectral image classification. *IEEE Trans. Image Process.* 31, 5079–5092. doi: 10.1109/TIP.2022.3192712

Yao, J., Cao, X., Hong, D., Wu, X., Meng, D., Chanussot, J., et al. (2022). Semi-active convolutional neural networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 60, 1–15. doi: 10.1109/TGRS.2022.3230411

Yao, J., Zhang, B., Li, C., Hong, D., and Chanussot, J. (2023). Extended vision transformer (exvit) for land use and land cover classification: A multimodal deep learning framework. *IEEE Trans. Geosci. Remote Sens* 61, 1-15. doi: 10.1109/TGRS.2023.3284671

Zhang, J., Dai, L., and Cheng, F. (2021a). Corn seed variety classification based on hyperspectral reflectance imaging and deep convolutional neural network. *J. Food Measurement Characterizat.* 15, 484–494. doi: 10.1007/s11694-020-00646-3

Zhang, W., Li, Z., Li, G., Zhuang, P., Hou, G., Zhang, Q., et al. (2024a). Gacnet: Generate adversarial-driven cross-aware network for hyperspectral wheat variety identification. *IEEE Trans. Geosci. Remote Sens.* 62, 1–14. doi: 10.1109/TGRS.2023.3347745

Zhang, W., Li, Z., Sun, H.-H., Zhang, Q., Zhuang, P., and Li, C. (2022). Sstnet: Spatial, spectral, and texture aware attention network using hyperspectral image for corn variety identification. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2022.3225215

Zhang, L., Liu, Z., Ren, T., Liu, D., Ma, Z., Tong, L., et al. (2020). Identification of seed maize fields with high spatial resolution and multiple spectral remote sensing using random forest classifier. *Remote Sens.* 12, 362. doi: 10.3390/rs12030362

Zhang, W., Sun, X., Zhou, L., Xie, X., Zhao, W., Liang, Z., et al. (2023b). Dual-branch collaborative learning network for crop disease identification. *Front. Plant Sci.* 14, 1117478. doi: 10.3389/fpls.2023.1117478

Zhang, X., Wang, Y., Zhang, N., Xu, D., Luo, H., Chen, B., et al. (2021b). Spectral–spatial fractal residual convolutional neural network with data balance augmentation for hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* 59, 10473–10487. doi: 10.1109/TGRS.2020.3046840

Zhang, F., Zhang, F., Wang, S., Li, L., Lv, Q., Fu, S., et al. (2023a). Hyperspectral imaging combined with cnn for maize variety identification. *Front. Plant Sci.* 14, 1254548. doi: 10.3389/fpls.2023.1254548

Zhang, W., Zhao, W., Li, J., Zhuang, P., Sun, H., Xu, Y., et al. (2024b). Cvanet: Cascaded visual attention network for single image super-resolution. *Neural Networks* 170, 622–634. doi: 10.1016/j.neunet.2023.11.049

Zhao, X., Que, H., Sun, X., Zhu, Q., and Huang, M. (2022b). Hybrid convolutional network based on hyperspectral imaging for wheat seed varieties classification. *Infrared Phys. Technol.* 125, 104270. doi: 10.1016/j.infrared.2022.104270

Zhao, W., Zhang, W., Pan, X., Zhuang, P., Xie, X., Li, L., et al. (2022a). Lessl: Can lego sampling and collaborative optimization contribute to self-supervised learning? *Inf. Sci.* 615, 475–490. doi: 10.1016/j.ins.2022.10.058

Zheng, J., Feng, Y., Bai, C., and Zhang, J. (2020). Hyperspectral image classification using mixed convolutions and covariance pooling. *IEEE Trans. Geosci. Remote Sens.* 59, 522–534. doi: 10.1109/TGRS.36

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 18 April 2024
DOI 10.3389/fpls.2024.1373590

Check for updates

OPEN ACCESS

EDITED BY
Hao Lu,
Huazhong University of Science and Technology, China

REVIEWED BY
Jana Shafi,
Prince Sattam Bin Abdulaziz University, Saudi Arabia
Nirmal Mazumder,
Manipal Academy of Higher Education, India

*CORRESPONDENCE
Sultan Alfarhood
sultanf@ksu.edu.sa

RECEIVED 20 January 2024
ACCEPTED 19 March 2024
PUBLISHED 18 April 2024

CITATION
Uddin MS, Mazumder MKA, Prity AJ, Mridha MF, Alfarhood S, Safran M and Che D (2024) Cauli-Det: enhancing cauliflower disease detection with modified YOLOv8. *Front. Plant Sci.* 15:1373590. doi: 10.3389/fpls.2024.1373590

COPYRIGHT
© 2024 Uddin, Mazumder, Prity, Mridha, Alfarhood, Safran and Che. 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.

Cauli-Det: enhancing cauliflower disease detection with modified YOLOv8

Md. Sazid Uddin[1], Md. Khairul Alam Mazumder[1], Afrina Jannat Prity[1], M. F. Mridha[1], Sultan Alfarhood[2*], Mejdl Safran[2] and Dunren Che[3]

[1]Department of Computer Science, American International University-Bangladesh, Dhaka, Bangladesh, [2]Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia, [3]School of Computing, Southern Illinois University, Carbondale, IL, United States

Cauliflower cultivation plays a pivotal role in the Indian Subcontinent's winter cropping landscape, contributing significantly to both agricultural output, economy and public health. However, the susceptibility of cauliflower crops to various diseases poses a threat to productivity and quality. This paper presents a novel machine vision approach employing a modified YOLOv8 model called Cauli-Det for automatic classification and localization of cauliflower diseases. The proposed system utilizes images captured through smartphones and hand-held devices, employing a finetuned pre-trained YOLOv8 architecture for disease-affected region detection and extracting spatial features for disease localization and classification. Three common cauliflower diseases, namely 'Bacterial Soft Rot', 'Downey Mildew' and 'Black Rot' are identified in a dataset of 656 images. Evaluation of different modification and training methods reveals the proposed custom YOLOv8 model achieves a precision, recall and mean average precision (mAP) of 93.2%, 82.6% and 91.1% on the test dataset respectively, showcasing the potential of this technology to empower cauliflower farmers with a timely and efficient tool for disease management, thereby enhancing overall agricultural productivity and sustainability

KEYWORDS

cauliflower disease detection, machine vision, YOLOv8, agricultural disease management, vegetable disease detection

1 Introduction

Agriculture is not only the primary source of food security, in agriculturally driven countries like Bangladesh, it is also one of the main sources of employment opportunities (Eunice et al., 2022). The agricultural sector holds a crucial position in maintaining rural communities, particularly in developing nations, as it provides the primary source of sustenance, income, and employment. This sector contributes around 6.4% of global

economic productivity which surmounts to over 5 million dollars (Statistics Times, 2018). Bangladesh is considered an agricultural nation, and its economy relies significantly on agricultural production. According to The Global Economy, up until 2022, more than 37% of the workforce is engaged in agriculture (The Global Economy, 2021) and accounts for 11.50% of the GDP of the nation (Bangladesh Bureau of Statistics, 2023). In Bangladesh, where agriculture is crucial, Cauliflower is a notable vegetable appreciated both for its popularity, health benefits and economic significance. Cauliflower belongs to the Brassicaceae family and is full of fiber and vitamins B (Pourdarbani and Sabzi, 2023), which are beneficial to health. Being a cruciferous vegetable, it provides heart-healthy fiber and choline, a substance critical for learning, memory, muscles, and sleep. Cauliflower is incredibly versatile in the kitchen, fitting seamlessly into various dishes whether consumed fresh or cooked.

Cauliflower is cultivated in numerous countries. In terms of production volume, the top countries include China, India, USA, Mexico, Spain, Italy, Turkey and Bangladesh. Bangladesh produced 283 kilotons of cauliflower in 2020 (Helgi Library, 2022). The ideal growing conditions for cauliflower are areas between 11 and 60° N with typical temperatures ranging from 5 to 28°C. While developing, it can tolerate temperatures ranging from -10°C to 40°C for a few days (Singh et al., 2018). Cauliflower farming faces challenges from several diseases like bacterial spot rot, black rot, downy mildew etc. The growth and productivity of cauliflower can be significantly impacted by these diseases. Farmers must identify these diseases early on and use the right method to control them. There exists multiple approaches for the control of cauliflower diseases such as physical control (hot water, nanoparticles), chemical control (pesticides) and biological control (Aqueous extracts, Bacillus etc.) (Liu et al., 2022). Pesticides used to protect cauliflower can be harmful to human health, and diseases caused by bacteria or fungus can cause problems including allergies when consumed (Pathak et al., 2022). This has an impact on the amount and quality of cauliflower as well as contributing to the annual loss of a significant chunk of harvests to plant diseases. Traditional ways of identifying cauliflower diseases in farming face significant challenges. Frequently, they depend on manual inspection, which is laborious, error-prone, and can miss early diseases. Most farmers, especially those in remote areas, cannot afford to hire and retain agricultural experts for disease identification.

The increasing adoption of optical imaging, machine vision and artificial intelligence techniques in vegetable disease detection and management has resulted in increasing demand for these applications in various areas of precision agriculture (Teet and Hashim, 2023). Convolutional Neural Network (CNN) approaches, in particular, are a promising option offered by more recent techniques utilizing modern technologies (Gu et al., 2018). These methods are faster, more accurate, and scalable, allowing for continuous crop monitoring. Many researchers are actively involved in identifying cauliflower diseases using CNN approaches, but these methods, while effective in disease classification, fall short in the problem of disease localization. In our research, we analyzed the latest methods for detecting and classifying crop diseases (especially cauliflower diseases) and introduced an approach using the YOLOv8 object detection model as a base which not only classifies cauliflower diseases but also identifies the region of affected areas in the images. Among all diseases, we worked on Black Rot, Downy Mildew, and Bacterial Sport Rot.

This paper introduces a modified YOLOv8 model for the localization and labelling of cauliflower diseases. This model combines the pre-trained knowledge of the YOLOv8 model with extra convolutional layers to improve the accuracy for identifying cauliflower disease. The primary contributions of this paper are enumerated as follows:

1. Custom YOLOv8s Model Development: The paper introduces a tailored YOLOv8s model designed specifically for identifying three prevalent cauliflower diseases—Downey mildew, bacterial spot rot, and black rot. The development of this custom model addresses the unique challenges posed by cauliflower diseases.
2. Performance Evaluation of Base YOLOv8 Models: The study conducts a comprehensive evaluation of base YOLOv8 models on a cauliflower disease detection dataset. By employing rigorous testing and comparison methodologies, the paper sheds light on the baseline performance of YOLOv8 models in the context of cauliflower disease detection.
3. Systematic Model Modifications for Improved Detection Performance: Building upon the baseline evaluation, the paper systematically applies different modifications to enhance the detection accuracy and average precision of the base YOLOv8 model. This contributes valuable insights into the specific adjustments and fine-tuning strategies that yield improvements in the model's ability to accurately detect and classify cauliflower diseases.
4. Open Access to Annotated Dataset and Proposed Model: The paper not only presents novel insights into custom model development and systematic modifications but also contributes to the research community by providing an annotated version of the VegNet cauliflower disease classification dataset (Sara et al., 2022). This dataset, along with the proposed custom YOLOv8s model, is made openly accessible. This contribution facilitates reproducibility, encourages further research, and establishes a foundation for ongoing advancements in the field of computer vision applied to agriculture.

The remaining portions of the paper are organized as follows: We discuss the existing literature for cauliflower disease detection and related problems in Section 2. We talk about an overview of the dataset and generally used image pre-processing techniques, and the methodology of the paper is explained in Section 3. We present the results of the performance observation of the models in Section 4. We discuss our findings in Section 5. The conclusion and potential future research endeavors are discussed in Section 6.

2 Literature review

Crop diseases pose serious risks to food production, economic stability, and food security, having a substantial impact on global

agriculture. These diseases have the potential to cause significant yield losses, endangering the livelihoods of millions of farmers and putting vital crops needed for commerce and subsistence at risk. In addition to being essential for maintaining a steady and secure food supply, crop diseases highlight the need for novel approaches, like the use of cutting-edge technologies like deep learning, to improve disease detection and mitigation techniques in the agricultural industry. The most recent and notable research endeavors in this domain are discussed in this section.

(Arun and Umamaheswari, 2023) introduced the Complete Concatenated Deep Learning (CCDL) framework, a multiple crop disease classification model capable of labelling crop diseases across many species of crop. The core functional unit of this architecture is the Complete Concatenated Block (CCB), which strategically places a point-wise convolution layer before each convolution layer to limit the increment of parameters in the model. The reorganized Plant Village dataset was used by the researchers to train this architecture. The PCCDL-PSCT approach proposed by the authors performed best, obtaining an impressive accuracy of 98.14% with a smaller model size of about 10 MB. (Chug et al., 2023) presented an innovative framework that combines the strengths of both machine learning and deep learning. The proposed framework comprises 40 diverse Hybrid Deep Learning (HDL) models. The performance of the HDL models was notably impressive on the IARI-TomEBD dataset, achieving high accuracy levels ranging from 87.55% to 100%. To validate the effectiveness of the approach, the researchers conducted experiments using two publicly available plant disease datasets, namely PlantVillage-TomEBD and PlantVillage-BBLS. (Huang et al., 2023) introduced an approach based on FC-SNDPN (Fully Convolutional – Switchable Normalization Dual Path Networks) for the automated detection and identification of crop leaf diseases. To mitigate the impact of complex backgrounds on the recognition of crop diseases and insect pests, the authors utilized a Full Convolutional Network (FCN) algorithm based on the VGG-16 model for target crop image segmentation. The SNDPN approach unites the connection method between DenseNet and ResNet layers, forming a neural network utilizing Switchable Normalization (SN) layers. The method proposed combines SNDPN for detecting diseases and FCN for segmenting the foreground, demonstrated an identification accuracy of 97.59% on the augmented dataset, affirming the efficacy of the proposed methodology. (Haridasan et al., 2023) employed a computer vision-centric approach, incorporating image processing, ML and DL techniques to diminish reliance on traditional methods for safeguarding paddy crops against diseases. The utilization of image segmentation to pinpoint the afflicted regions of the paddy plant was proposed, identifying diseases solely based on their visual characteristics. A combination of a SVM classifier and CNN was employed for the recognition and classification of specific types of paddy diseases. By incorporating ReLU and SoftMax functions, the proposed deep learning strategy achieved a validation accuracy of 91.45%.

(Mallick et al., 2023) presented an innovative deep learning approach for the identification of pests and diseases affecting mung beans. To address the challenge posed by the limited number of available mung bean crop images available for training, the researchers employed transfer learning, which yielded highly promising results for swift and effective disease as well as pest detection. The proposed model successfully distinguished 6 types of mung bean diseases and 4 types of pests from healthy and diseased leaves collected across various seasons. Through experimentation, the proposed lightweight DL model for mung bean disease and pest detection demonstrated an impressive average accuracy of 93.65%. (Zhao et al., 2023) introduced enhancements to the YOLOv5s model for improved crop disease detection. The modifications included refining the CSP structure in the feature fusion stage, incorporating a lightweight composition to reduce model parameters, and extracting feature information through multiple branches. Addressing scaling issues during training, an improved DIoU loss function replaced the Generalized IoU loss function from the original YOLOv5. Through transfer learning, the enhanced model exhibited superior mean average precision (mAP) compared to YOLOv3, YOLOv4, YOLOv4-tiny, YOLOv5s, Faster R-CNN and SSD models, achieving recall, F1 and mAP mAP, F1 score, and recall of 87.89%, 91%, and 95.92%, respectively. These values marked improvements of 4.58%, 5%, and 4.78%, respectively, compared to YOLOv5s. (Lin et al., 2023) introduced an enhanced YOLOX-Tiny network, denoted as YOLO-Tobacco, designed for detecting brown spot disease in open-field tobacco crop images. Their objective was to uncover crucial disease features and improve the fusion of diverse feature levels, facilitating the detection of dense disease spots across various scales. The YOLO-Tobacco network demonstrated an AP (average precision) of 80.56% on the test set, surpassing available lightweight detection models such as YOLOX-Tiny, YOLOv5s, and YOLOv4-Tiny by 3.22%, 8.99%, and 12.03%, respectively. (Hu et al., 2023a) introduced a novel Multi-Scale Dual-branch model for pest identification from rice crop images, employing a GAN (generative adversarial network) and an enhanced ResNet to discern pests in complex background images. To optimize the calculations ratio of residual blocks, the ConvNeXt residual block was incorporated into the ResNet model and a dual-branch structure was devised to extract features of disease affected spots of varying sizes, adjusting the convolution kernel size for each branch. Training the new model on a systematically expanded dataset improved recognition accuracy by 2.66% compared to the original ResNet model. In comparison with base networks like AlexNet, DenseNet, VGG, ResNet, and Transformer under similar conditions, the new model demonstrated superior performance, achieving a disease recognition accuracy of 99.34%.

(Thakur et al., 2023) presented a lightweight CNN model named 'VGG-ICNN' designed for identifying crop diseases through plant-leaf images. The VGG-ICNN model comprises approximately 6 million parameters, significantly fewer than many existing high-performing DL models. The model's effectiveness was assessed across five diverse public datasets encompassing various crop types, including multi-crop datasets like Embrapa and PlantVillage with 93 and 38 categories, respectively, and single crop datasets like Maize, Rice and Apple each with four or five categories. Experimental outcomes indicated that the proposed method surpassed several recent DL approaches in crop disease identification, achieving an accuracy of 99.16% on the PlantVillage

dataset. (Zhu et al., 2023) introduced EADD-YOLO, a model for accurate and efficient apple leaf disease detection model based on YOLOv5. EADD-YOLO utilized the shufflenet inverted residual blocks in the backbone and utilizing depthwise convolution to propose an efficient feature learning module in the neck. To improve detection accuracy for diseases of various sizes in different scenes, a coordinate attention module was embedded in critical locations to highlight crucial information and suppress irrelevant details. Additionally, the SIoU was used as the bounding box regression loss instead of CIoU to improve prediction box localization accuracy. Experimental results demonstrated mAP of 95.5% and 625 FPS inference on video on the apple leaf disease dataset (ALDD). Compared to other recent works on ALDD, the proposed method improved detection accuracy and speed by 12.3% and 596 FPS, respectively, with significantly fewer parameters and FLOPs. (Wang et al., 2023a) introduced Cropformer, a novel deep learning method designed for crop classification on multiple scenarios. Addressing the limitations of existing approaches that focused on extracting a single feature, Cropformer adopted a two-step classification process. In the initial step, the model undergoes a pre-training phase of self-supervised fashion to learn about crop growth features, followed by a second step involving supervised fine-tuned classification using weights derived from the first step. The study conducted comprehensive experiments on multi-scenario crop classification, covering scenarios regarding season and sample size, and transfer scenarios in 5 study areas with diverse crop types. Comparison with existing approaches revealed that the Cropformer not only achieved significantly higher accuracy in crop classification, but also demonstrated superior accuracy utilizing fewer samples. The proposed approach presents a notable advancement in addressing the challenges of multi-scenario crop classification through its unique two-step classification strategy. (Hu et al., 2023b) introduced a novel Lesion Proposal CNN based on Class-Attention called CALP-CNN designed for strawberry disease identification. The CALP-CNN employs a class response map to pinpoint the primary lesion object and suggest distinctive lesion details. Utilizing a cascading architecture, CALP-CNN concurrently addresses challenges related to complex backgrounds and the potential misclassification of similar but different instances. Experimental evaluations conducted on a self-assembled dataset of strawberry diseases attest to the effectiveness of CALP-CNN. The classification results for CALP-CNN demonstrate metrics of 92.56%, 92.55%, 91.80%, and 91.96% for accuracy, precision, recall, and F1-score, respectively. (Masood et al., 2023) introduced MaizeNet, a deep learning (DL) approach designed for the accurate identification and classification of diverse maize crop leaf diseases. Their method, an enhanced Faster-RCNN approach, employed the ResNet-50 model utilizing spatial channel attention as the underlying network for computing deep keypoints, which were subsequently localized and categorized across various classes. MaizeNet demonstrated notable effectiveness with an accuracy of 97.89% and a mAP of 94%, underscoring its efficacy in accurately locating and classifying different types of maize leaf infections.

Research on deep learning techniques for the diagnosis of cauliflower disease is noticeably lacking. Although deep learning is being used more and more in agricultural settings, especially for crop disease detection, the particular field of cauliflower diseases is still not well studied. Few researchers attempted to employ the deep learning techniques to identify cauliflower diseases. (Rajbongshi et al., 2022) introduced an online expert system based on machine vision designed for the identification of cauliflower diseases. The system processed images captured via smartphones and handheld devices then identifying them to recognize diseases and provide assistance to cauliflower farmers. The feature extraction process enabled the classification of four types of cauliflower diseases, including 'bacterial soft,' 'white rust,' 'black rot,' and 'downy mildew.' The experiment utilized 776 images, employing K-means clustering for the segmentation of disease-affected regions, followed by co-occurrence and statistical feature extraction. BayesNet, Kstar, Random Forest, LMT, BPN, and J48 classification algorithms were employed for classification of cauliflower diseases. The evaluation of these algorithms revealed that the Random Forest classifier superseded others with an accuracy approaching 89.00%. (Shakil et al., 2023) developed an agro-medical expert system for the diagnosis of cauliflower diseases. The affected portions of cauliflowers were segmented using the k-means clustering algorithm. Subsequently, 10 statistical and GLCM features were extracted from the segmented images. After selecting the top N features (where N = {5, 10}), the SMOTE technique was applied to address training datasets with varying feature quantities. Five machine learning (ML) algorithms were then utilized, and their performance was assessed for non-augmented and augmented datasets. The identical procedure was applied to both datasets, and the classifier's performance was tested on both. Logistic regression (LR) was found as the most accurate method, achieving a 90.77% accuracy based on top 9 features in the augmented dataset. (Kanna et al., 2023) conducted experiments to assess various pre-trained DL models for the early prediction of diseases in cauliflower plants. The study focused on 3 classes of cauliflower diseases, namely Bacterial spot rot, Black rot, Downy Mildew, along with healthy cauliflower images, sourced from the VegNet dataset. Transfer learning models like EfficientNetB0, Xception, EfficientNetB1-B2-B3-B4, MobileNetV2, DenseNet201, InceptionResNetV2, and ResNet152V2, were trained and evaluated based RMS error, accuracy, precision, recall and F1-score. Notably, EfficientNetB1 demonstrated the best validation accuracy (99.90%), the smallest loss (0.16), and RMS error (0.40) during the experimentation. (Maria et al., 2022) introduced several methodologies for identifying diseases affecting cauliflower plants, comparing the effectiveness of traditional ML and TL. In their study, traditional machine learning involved image preprocessing followed by k-means clustering for image segmentation, and then the extraction of ten pertinent features. Various classification techniques were compared, with the Random Forest algorithm producing accuracy of 81.68%. Moreover, they explored CNN architectures for TL, including InceptionV3, MobileNetV2, ResNet50, and VGG16. Among these, InceptionV3 exhibited the highest accuracy at 90.08%, showcasing superior performance compared to the traditional machine learning approach. (Li et al., 2022) introduced a detection and classification model for surface defects in fresh-cut cauliflower based on a CNN with transfer

learning. A dataset comprising 4,790 images of fresh-cut cauliflower, categorized into healthy, diseased, browning, and mildewed classes, was collected for the study. The authors fine-tuned the pre-trained MobileNet model to enhance both training speed and accuracy. Optimizing the model involved selecting the best configuration of hyper-parameters and freezing layers. Tests which combined VGG19, InceptionV3, and NASNetMobile, results were compared. Experimental outcomes demonstrated that, with an LR of 0.001, dropout set at 0.5, and 80 frozen layers, the MobileNet model achieved a loss value of 0.033, an accuracy of 99.27%, and an F1 score of 99.24% on the test dataset. (Abdul Malek et al., 2022) conducted research on the classification of four distinct cauliflower diseases, namely bacterial soft rot, black rot, buttoning, and white rust, utilizing several CNN models in conjunction with transfer learning. The dataset employed for this study comprised approximately 2500 images. Notably, InceptionV3 emerged as the most successful among the various CNN models investigated, achieving a remarkable test accuracy of 93.93%. This performance surpasses the outcomes observed in comparable experiments conducted in recent times.

In summary, the aforementioned works reveal significant advancements in the application of DL techniques for crop disease classification, including diverse models. These models have demonstrated high accuracy and efficiency in identifying diseases across various crops, contributing to improved agricultural practices. However, despite these notable achievements, the specific domain of cauliflower disease detection has received limited attention. While researchers have explored deep learning methods for cauliflower disease identification, there is a noticeable gap in addressing the precise localization of diseases within cauliflower images. Existing studies primarily focus on classifying diseases without providing information on the spatial distribution of symptoms within the images. This lack of emphasis on detection and localization limits the practical applicability of the models in precision agriculture, where identifying not only the presence but also the location of diseases is crucial for targeted interventions.

Motivated by this gap in the current state of cauliflower disease detection, our research aims to address the challenge of cauliflower disease detection and localization. By leveraging state-of-the-art DL techniques and drawing inspiration from successful models like YOLO, we developed a model that not only accurately classifies cauliflower diseases but also provides insights into the spatial distribution of disease symptoms within the images. This approach aligns with the broader goal of advancing precision agriculture by offering farmers a more comprehensive detection system of disease patterns in their cauliflower crops. The significance of our proposed approach lies in its potential to enhance disease management strategies by enabling farmers to pinpoint the specific locations where disease symptoms are most prevalent. This information can guide targeted interventions, such as precise application of pesticides or other treatments, minimizing resource usage and environmental impact. Ultimately, our research seeks to contribute to the development of a more robust and practical solution for cauliflower disease detection and localization, thereby addressing a critical need in the domain of precision agriculture.

3 Materials and methods

3.1 Dataset description

The VegNet dataset (Sara et al., 2022) has been meticulously curated to facilitate the effective recognition of diseases in cauliflower leaf and flower. This dataset encompasses well-organized and technically valuable images of both diseased and healthy cauliflower heads and leaves. The targeted diseases include Downy Mildew, Black Rot, and Bacterial Spot Rot. The selection of Downy Mildew, Black Rot, and Bacterial Spot Rot for this study was based on three key factors. Firstly, despite extensive searches, limited datasets covering cauliflower diseases were found, with the VegNet dataset emerging as the primary resource due to its comprehensive image coverage. Secondly, these diseases were chosen due to their prevalence and economic significance in global cauliflower crops, aligning our study with the pressing concerns of cauliflower growers and agricultural stakeholders. Lastly, each disease offers distinct visual characteristics and poses unique challenges for detection algorithms, enriching the dataset and enabling comprehensive evaluation of our proposed Cauli-Det system. While Bacterial Spot Rot is a cauliflower head disease, Downy Mildew and Black Rot mainly affect cauliflower leaves. Symptoms and disease conditions were verified by a plant pathology expert from the Bangladesh Agricultural Research Institute (BARI). The images were captured manually from the Manikganj area of Bangladesh during the period from 20 Dec to 15 Jan, 2022, when the cauliflower flowers were in full bloom and diseases were prominently observed. A Sony Cyber-Shot W-530 digital camera with a resolution of 14 MegaPixels was used to capture the images in JPEG format which were then pre-processed using Python into a standard image size of 256x256 pixels. Images in this standard format are used as the input to the YOLOv8 model. The authors of the dataset describe that the image preprocessing steps were achieved by tweaking image brightness, contrast, hue and saturation to bring forth the best visualization of disease features for the ease of detection tasks. The VegNet dataset comprises a total of 656 images, distributed among different categories based on the observed conditions. The images were split in a 70%-15%-15% ratio for training, validation and testing respectively. To ensure that the model generalizes without bias in detecting cauliflower diseases reliably, we evaluate the model's performance on both seen data (validation dataset) and unseen data (test dataset).

Table 1 summarizes the distribution of images in the dataset per disease. Figure 1 shows sample images from the VegNet dataset with hand drawn annotations.

3.1.1 Disease causes and symptoms

Downey Mildew: Due to a fungus called Peronospora parasitica (Muimba-Kankolongo, 2018), which causes white, yellow, or brownish patches on older leaves, accompanied by downy gray mold on the undersides are observed on leaves. Lesions and intrinsic holes facilitate penetration, releasing more spores.

Affected areas deepen in color, leading to the death of the leaf. Environmental Factors for Downey Mildew include moisture and low temperatures which favor the growth of this fungal parasite.

TABLE 1 VegNet dataset image distribution.

Class Name	Percentage	Training	Validation	Test	Total
Downey Mildew	27.0%	125	26	26	177
Black Rot	15.2%	70	15	15	100
Bacterial Spot Rot	26.4%	121	26	26	173
Healthy	31.4%	144	31	31	206
Total	100%	460	98	98	656

The percentage column represents the distribution of images per class.

Black Rot: Xanthomonas campestris bacterial Infection causes dull, irregular yellow spots on leaf edges, progressing into V-shaped patches (Sheng et al., 2020). The wide section of the "V" is at the leaf's border and attachment point to the plant. Symptoms may take up to a month to appear after cauliflower growth begins which renders cauliflowers unfit for sale or consumption.

Bacterial Spot Rot: Alternaria brassicicola (Tao et al., 2022) bacterial infection results in lesions on flower heads soaked in water form a rotting mass. Lesions often split, releasing a slimy goo that turns from first brown, then to black when exposed in the atmosphere. Transmission occurs through tools and irrigation water. Warm, moist conditions favor this disease which requires control through agricultural practices such as crop rotation, well-draining soils, and avoiding negative charges during harvest, since no chemical treatment is available.

3.1.2 Dataset annotation and disease localization

The VegNet dataset comprises of image-level labels, where each whole image is labeled as either having one or no disease. Recognizing the importance of finer-grained analysis for disease management, the dataset has been enhanced through manual annotation. Using the annotation tool Makesense.ai, bounding boxes were manually drawn to localize specific regions within the images that exhibit disease symptoms. This detailed annotation approach provides a granular understanding of the spatial distribution of diseases within cauliflower plants. By precisely localizing disease-affected areas, farmers and agricultural practitioners can administer targeted treatments. This enables the application of pesticides, fungicides, or other control measures specifically to the identified regions, minimizing the use of resources and reducing environmental impact. The annotated dataset availability is included in section 6.

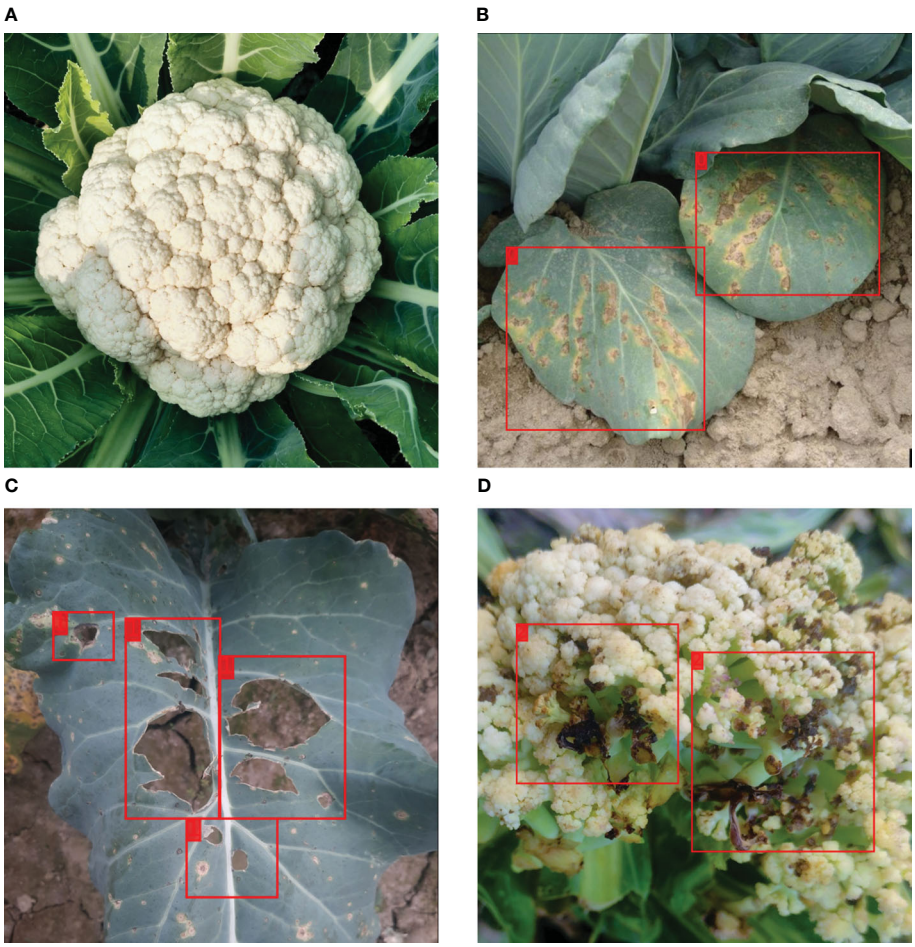


FIGURE 1
Sample images from the VegNet dataset with annotations. **(A)** Healthy cauliflower, **(B)** Downey Mildew infected cauliflower leaf, **(C)** Black Rot infected cauliflower leaf, **(D)** Bacterial Spot Rot infected cauliflower.

3.2 Base YOLOv8 model

The YOLOv8 architecture begins with a series of convolutional layers with stride and kernel size configurations, followed by a batch normalization layer and an activation function. These layers reduce spatial dimensions progressively and at the same time, increases the channel number of the tensor. This is a downsampling process which facilitates the extraction of high-level features (Khan et al., 2020). A critical component of YOLOv8 is the Cross-Stage Partial Fusion module, which incorporates bottleneck structures to enhance feature representation (Wang et al., 2020). These bottlenecks consist of multiple convolutional layers with batch normalization and activation functions. A block of a convolutional layer followed by a batch normalization layer and an activation function is called a *Conv* block by the official implementation of YOLOv8 (Jocher et al., 2023). This repository also uses a faster implementation of the Cross-Stage Partial Fusion module and Spatial Pyramid Pooling module, called the *C2f* and *SPPF* module. Figure 2 shows the implementation of the *Conv*, *C2f* and *SPPF* block using PyTorch modules. The YOLOv8 uses the spatial pyramid pooling module (*SPPF*) to capture features at multiple scales (He et al., 2015). This module utilizes max-pooling operations with different kernel sizes to aggregate contextual information. The architecture employs upsampling layers and concatenation operations to fuse features from different stages. This allows the model to refine and combine information from both high and low-level representations.

The YOLOv8 architecture employs a multi-resolution feature fusion strategy to effectively capture both detailed and high-level information from different levels of the backbone. This process involves upsampling, concatenation, and additional convolutions to ensure that features of different resolutions are appropriately combined in the neck of the network before being fed into the detection head. The Upsampling layers increase the spatial resolution of the lower-resolution features to match that of higher-resolution features. Upsampling is performed using a specified scale factor, effectively enlarging the feature maps. *C2f* modules are inserted in the neck after each concatenation of the high and low-level features. The neck includes a subsequent convolutional layers that transform the features and adjust their channel dimensions. The output of this branch retains the original spatial resolution but gains expressive power through convolutional operations. The process of upsampling, fusion through C2f modules, and concatenation is repeated until the features from all levels of the backbone are combined. The final result is three feature maps, each containing information from different resolutions. These feature maps are then passed to the detection head for further processing and prediction of bounding box coordinates and class probabilities. The complete base YOLOv8 model architecture is visualized in Figure 3. The head of the model constitutes of two components:

1. Detection Head: The detection head receives three feature tensors from the neck as inputs and puts them separately through a series of *Conv* blocks and finally a convolutional layer which converts the channel number to $16 * 4$, where 16 is the number of Distribution Focal Loss channels and 4 is the number that signifies the attributes of the bounding box. These are $[x,y,w,h]$, where x and y coordinates of the center and the width and height of the bounding box.
2. Classification Head: The classification head also receives the same three feature tensors and puts them through a series of *Conv* blocks and a final convolutional layer which converts the channel number to the number of classes, in this case 3 for the types of cauliflower diseases.

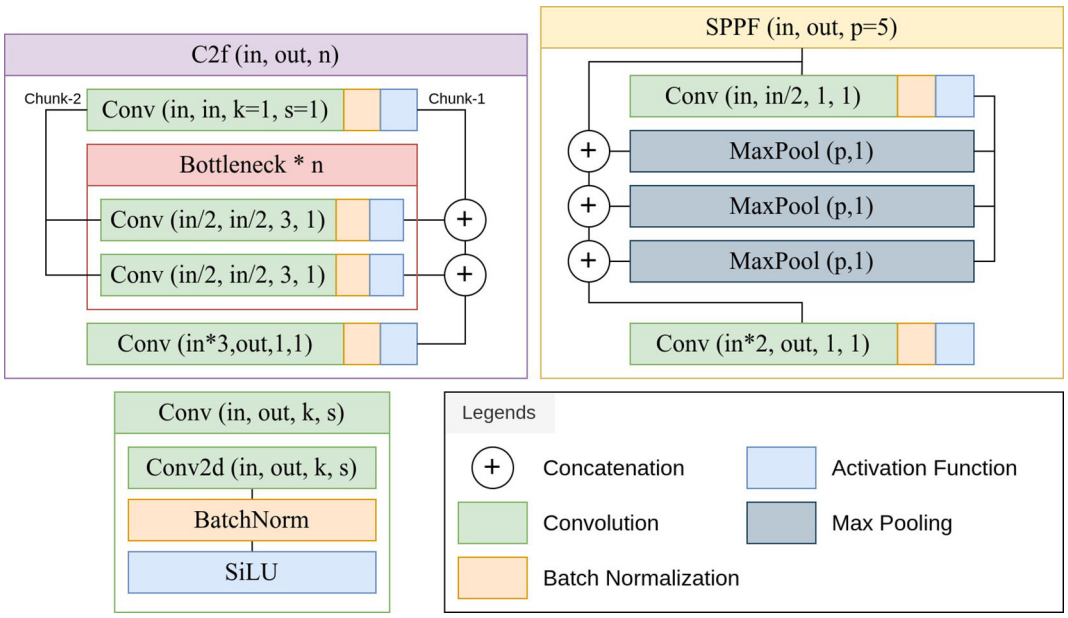


FIGURE 2
(Jocher et al., 2023) Implementation of the *Conv*, *C2f* (Cross-Stage Partial Fusion) and *SPPF* (Spatial Pyramid Pooling) blocks.

3.3 Modifications of YOLOv8 for cauliflower disease detection

3.3.1 Adding extra *Conv* blocks to the head

The original Detection and Classification head of the YOLOv8 model utilizes *Conv* blocks with a kernel size of 3, yielding a tensor with 64 channels for each of the three feature maps. Subsequently, this tensor is transformed into the requisite channels for detection and classification output through the application of a convolutional layer. In an augmentation to the base model, 3 additional *Conv* blocks, each with a kernel size of 1, have been inserted prior to the output convolutional layer. The introduction of more *Conv* blocks facilitates an increased depth within the model architecture while not increasing the number of parameters significantly, which helps with more sophisticated processing of the feature maps. Along with the capabilities of the pre-trained model, these extra *Conv* blocks improve the model's ability to learn the domain specific information of cauliflower diseases. Figure 4 shows the difference between the original YOLOv8 head and the proposed custom head.

3.3.2 Learning rate configuration

After adding extra *Conv* blocks, we conducted experimentation to assess the ramifications of varying learning rates across discrete sections of the network. The pre-trained YOLOv8s model comes trained on the large COCO dataset. The pre-trained weights retain the feature representation power of the COCO objects. We want to preserve that knowledge and apply it to the domain of Cauliflower Disease Detection. By varying the learning rate of different sections of the network, we can control how much of the learned parameters are preserved. To find the optimal LR configuration, modifications are established wherein the parameters of the network are bifurcated into two distinct groups, denoted as the slow group and the fast group. Learning rate for the fast group remains at default levels, while the learning rate for the slow group undergoes a reduction by a factor of 100. The sections of the network subject to this parameter split include, for instance, the Slow Backbone and Fast Neck and Head, as well as the Slow Backbone and Neck and Fast Head, among others. From the results observed, it was determined that the optimal LR configuration is to use the same LR for all sections. The details of the findings are discussed in Section 4.

3.3.3 Activation function of the *Conv* blocks

YOLOv8 by default employs the Swish or Sigmoid-Weighted Linear Unit (SiLU) activation function. Through a series of systematic experiments involving the exploration of diverse activation functions, it was discerned that the utilization of the Hard swish activation function yielded a marginal yet discernible enhancement in mean Average Precision (mAP). The Swish activation function (Ramachandran et al., 2017) is defined in Equation 1.

$$f(x) = x \cdot \sigma(x) \tag{1}$$

where σ represents the logistic sigmoid function.

Hard Swish (Howard et al., 2019), a modification of Swish, is formulated in Equation 2.

$$h(x) = \begin{cases} 0 & \text{if } x \leq -3, \\ x & \text{if } x \geq +3, \\ x \cdot (x+3)/6 & \text{otherwise} \end{cases} \tag{2}$$

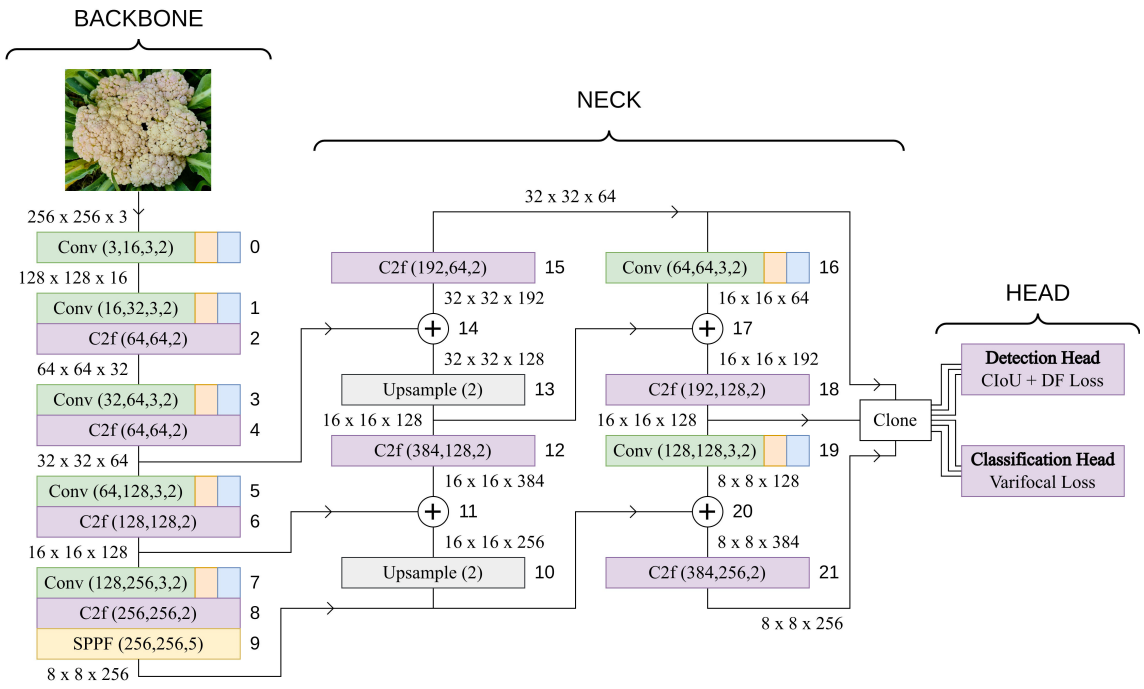


FIGURE 3
Base YOLOv8 architecture. The output dimensions of each layer is based on an input image of size 256x256.

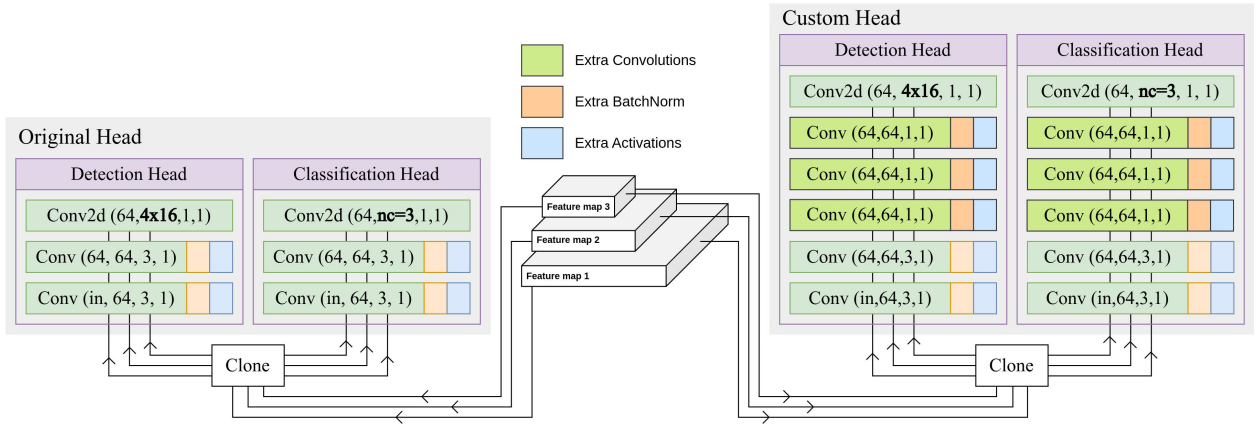


FIGURE 4
Difference between the original YOLOv8 head and the proposed custom head. 3 extra *Conv* blocks have been inserted prior to the output convolutional layer.

While Swish and Hard Swish share a common foundation, the key disparity lies in the non-linear component. Swish incorporates a smooth sigmoid function, whereas Hard Swish introduces a clipped linear function. This discrepancy while not resulting in distinct shapes, provides an advantage. Swish exhibits stronger non-linear characteristics, but the computational cost associated with its smooth sigmoid component slows down training speed. Conversely, Hard Swish provides a compromise by maintaining non-linearity with a simpler clipped linear operation, leading to improved efficiency without sacrificing performance. Figure 5 shows the Swish and Hard Swish activation functions shapes.

3.4 Experimental setup

3.4.1 Hardware and software specifications

The source code for our experiments was built upon the Ultralytics YOLOv8 repository (Jocher et al., 2023) which was modified to compare different modification configurations. The software and hardware specifications are described in Table 2.

3.4.2 Hyperparameters

The default hyperparameters for our experiments are described in Table 3. These parameters hold true for all experiments unless stated otherwise.

3.4.3 Loss functions

YOLOv8 implements 3 loss functions, all of which are summed together to form the total loss which is then passed to the optimizer. The three losses are:

• Varifocal Loss: Varifocal Loss is a loss function to train a dense object detector and predict the IoU Aware Classification Score (IACS), inspired by focal loss (Lin et al., 2017) which is used as the classification loss. Defined in Equation 3.

$$\mathrm{VFL}(p,q) = \begin{cases} -q \times (q\log(p) + (1-q) \times \log(1-p)) & \text{if } q > 0, \\ -\alpha p^{\gamma} \log(1-p) & \text{if } q = 0 \end{cases} \quad (3)$$

where p is the predicted IACS and q is the target IoU Score and α is the weight coefficient.

• CIoU Loss: Complete IoU Loss, which is used as the first part of the regression loss. Defined in Equation 4.

$$\mathrm{CIoU} = 1 - \mathrm{IoU} + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \quad (4)$$

where, ρ is the distance between the predicted bounding box and the correct bounding box
b and b^{gt} represent the center point of the two bounding boxes
c is the diagonal distance of the closure area of the boxes
v measures the consistency of the relative proportion of the boxes.

• DFL: Distribution Focal Loss which is the second part of the regression loss, is calculated using the general distributions of bounding boxes to force the networks to learn the probabilities of values close to the target coordinates. Defined in Equation 5.

$$\mathrm{DFL} = -(\log(S_i) \times (y_{i+1} - y) + \log(S_{i+1}) \times (y - y_i)) \quad (5)$$

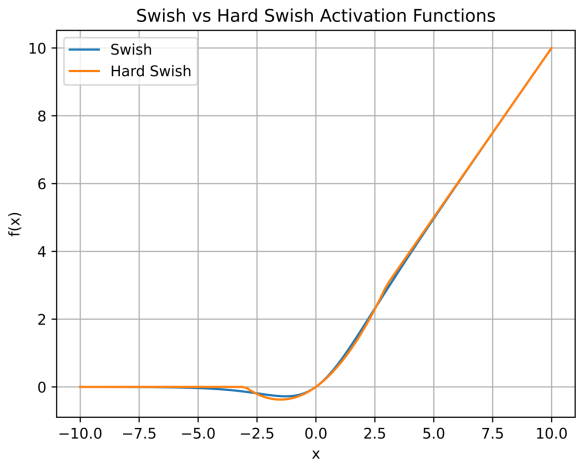


FIGURE 5
Swish and Hard Swish activation function shapes. While being very similar in output, Hard Swish is more computationally efficient.

TABLE 2 Hardware and software specifications.

Hardware Setup		Software Setup	
CPU	Intel(R) Xeon(R)	Programming Language	Python 3.10.2
System Memory	12.7 GB	Deep Learning Framework	PyTorch 2.1.0
GPU	Tesla T4	GPU Support	CUDA 11.1
GPU Memory	16 GB		

where, S_i and S_{i+1} represent the scores or probabilities assigned to adjacent classes or categories. y_i and y_{i+1} represent ground truth labels or true probabilities associated with the classes i and i + 1 respectively

3.4.4 Evaluation metrics

Evaluation metrics are crucial for assessing the performance of object detection models such as YOLO. Evaluating our proposed model's disease detection capabilities, and its performance requires a set of metrics that can quantify its accuracy and efficiency. We used the following evaluation metrics for our experiments:

• Intersection over Union (IoU): IoU measures the spatial overlap between the predicted bounding boxes (BB_{pred}) and correct bounding boxes (BB_{gt}). IoU is calculated as in Equation 6.

$$IoU = \frac{BB_{pred} \cap BB_{gt}}{BB_{pred} \cup BB_{gt}} \quad (6)$$

• Precision: Precision calculates the ratio of true positives within all positive predictions, evaluating the model's capability to avoid false positives. Precision is defined as in Equation 7.

TABLE 3 Default hyperparameters for training.

Hyperparameter	Description	Default Value
epochs	Number of epochs	200
patience	Epochs to wait for early stopping	50
batch	Batch size	32
workers	Concurrent Threads	16
imgsz	Resize image to	256x256 pixels
pretrained	Use Pre-Trained weights	True
optimizer	Optimizer	AdamW
freeze	Disable weight update for these layers	None
lr0	Initial Learning rate	0.001429
momentum	Learning Rate momentum	0.9
box	Box loss gain	7.5
cls	Class loss gain	0.5
dfl	DFL loss gain	1.5

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

• Recall: Recall calculates the proportion of true positives among all ground truth objects, evaluating the model's ability to identify all instances of objects in the dataset. Recall is defined as in Equation 8.

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

• Average Precision (AP): AP is calculating precision-recall curves for different confidence thresholds and then computing the area under the curve (AUC). Average Precision provides a single scalar value that summarizes the model's performance across different precision-recall trade-offs for a class.

• Mean Average Precision (mAP): It is computed by averaging the AP values across all classes. When nc is the number of classes, mAP is defined as in Equation 9.

$$mAP = \frac{1}{nc}\sum_{c=1}^{nc} AP(c) \quad (9)$$

Two different mAP values are calculated for all experiments: mAP50 (mAP is calculated with an IoU threshold of 0.5) and mAP50-95 (mAP calculated at varying IoU thresholds ranging from 0.5 to 0.95).

4 Result analysis

The Mean Average Precision (mAP) serves as a comprehensive metric to encapsulate the performance of object detection models, providing a singular value that reflects both precision and recall. In the context of cauliflower disease detection, the use of mAP is particularly pertinent due to its ability to gauge the model's proficiency in identifying and localizing cauliflower disease instances in an image. The mAP metric synthesizes precision-recall curves across various thresholds, offering a concise representation of the model's overall effectiveness. This is why we used mAP to assess the performance of baseline models as well as different modification configurations of YOLOv8 in order to find the best model for the problem of cauliflower disease detection.

4.1 Performance comparison of baseline models

To find the best baseline model for the problem of cauliflower disease detection, we compared the performance of YOLOv7 (Wang et al., 2023b), along with YOLOv8's nano, small, medium, large and extra large models. The results of these experiments are shown in Table 4. All models were pretrained on the COCO object detection dataset (Lin et al., 2014).

Figure 6 visualizes the performance and parameters counts between mAP and parameters of baseline models. We observe that, while YOLOv7 has the highest validation and test mAP (95.5% and 92.6%) followed closely by YOLOv8s (92.9% and 91.0%), YOLOv7 has more than thrice the parameter count (37.21 million) than YOLOv8s (11.14 million). Other larger sizes

of YOLOv8 are observed to not perform up to the expectation that comes with their higher parameter counts. This phenomenon can be attributed to redundant network depth. Smaller models like the YOLOv8s has enough depth to capture and model the features of cauliflower disease images, resulting in bigger models unable to perform better than YOLOv8s. Increased depth also results in these models being unfit to be employed on lower-end devices for practical applications like video inference. On the other hand, YOLOv8s has the lowest validation and test mAP due its small size. It is thus concluded that YOLOv8s has the best balance of performance and parameter count, judging by the law of diminishing return. Therefore we chose YOLOv8s for further modification and evaluation with a goal of improving the performance while not increasing the parameter count significantly.

4.2 Performance comparison of modified YOLOv8s

4.2.1 Comparison of head configurations

The primary focus was on enhancing the performance of YOLOv8s without significantly increasing the parameter count. The approach employed involved the addition of extra *Conv* blocks to both the detection and classification heads of YOLOv8s. The goal of our experimentation was to discern the impact of augmenting the head on the performance of YOLOv8s. The configurations employed involved adding 1, 2, 3, 4, and 5 extra *Conv* blocks to determine which is the best performing. Table 5 shows the results of adding different numbers of *Conv* blocks to the head of YOLOv8s.

Figure 7 visualizes the performance and cost tradeoffs of adding different numbers of *Conv* blocks. The analysis of the results demonstrates that adding 3 extra *Conv* blocks yields the best mean average precision on the test dataset. Intriguingly, further augmentation to 5 *Conv* blocks do not bring a significant improvement in performance. The diminishing returns observed with the addition of more than 3 *Conv* blocks can be attributed to a phenomenon of diminishing feature discriminability. While the initial addition of extra convolutional blocks contributes to the model's ability to capture and learn complex features, the excessive introduction of these blocks can lead to overfitting or redundant feature extraction. As a result, the model may become overly specialized on the training data, impairing its generalization ability on unseen data.

4.2.2 Comparison of different LR optimization

Following the incorporation of 3 additional *Conv* blocks that demonstrated improved model performance, we further refined the training strategy by exploring the impact of different learning rates across distinct sections of the network. Given the utilization of a pre-trained model as the base, the objective is to fine-tune the learning rates strategically to balance the preservation of learned parameters in the original layers with accelerated learning in the newly added components. The experimentation encompassed six distinct configurations:

- Default: All sections of the network share a uniform learning rate.
- Freeze–Non–Extra–Conv: All layers except the newly added *Conv* blocks are frozen during training.
- Freeze–Back: The backbone is frozen, while the neck and head layers remain trainable with the same learning rate.
- Fast–Extra–Conv: Original layers of the model receive a learning rate reduced by a factor of 100, while the extra *Conv* blocks maintain their original learning rate.
- Fast–Head: Both the backbone and neck undergo a 100x reduction in learning rate, while the head retains the original learning rate.
- Fast–Head–Neck: Only the backbone experiences a 100x reduction in learning rate, with the neck and head maintaining their original learning rates.

Table 6 provides the results of these experiments. Surprisingly, the experimentation revealed that freezing either the entire backbone or all layers except the newly added layers resulted in a significant decline in performance. This outcome was attributed to the restrictive nature of preventing the adjustment of layer weights, compelling the model to rely solely on the pretrained feature extraction. This approach proved suboptimal for detecting cauliflower diseases, as the model struggled to adapt its pre-learned features to the specific nuances of this domain.

Moreover, the results indicated that employing a slower learning rate for any section, even with the intention of preserving pretrained knowledge, led to a small but noticeable drop in performance. Consequently, the decision was made to allow all layers to freely adjust their weights using the default learning rate. This approach yielded the best LR configuration,

TABLE 4 Performance and parameters of base YOLOv7 and YOLOv8 models on the validation dataset and test dataset.

Model	Precision val | test	Recall val | test	mAP50 val | test	mAP50-95 val | test	Parameters
YOLOv7	**0.964** | **0.978**	**0.941** | **0.889**	**0.955** | **0.926**	**0.716** | 0.718	37.21 M
YOLOn8n	0.898 | 0.910	0.809 | 0.828	0.844 | 0.821	0.562 | 0.577	3.01 M
YOLOv8s	0.932 | 0.914	0.817 | 0.832	0.886 | 0.841	0.599 | 0.661	11.14 M
YOLOv8m	0.919 | 0.912	0.865 | 0.868	0.916 | 0.916	0.702 | 0.721	25.86 M
YOLOv8l	0.929 | 0.904	0.870 | 0.875	0.924 | 0.915	0.689 | 0.711	43.63 M
YOLOv8x	0.938 | 0.918	0.872 | 0.847	0.923 | 0.910	0.694 | **0.723**	68.16 M

Bold values signify the value of the best performing base model under each metric.

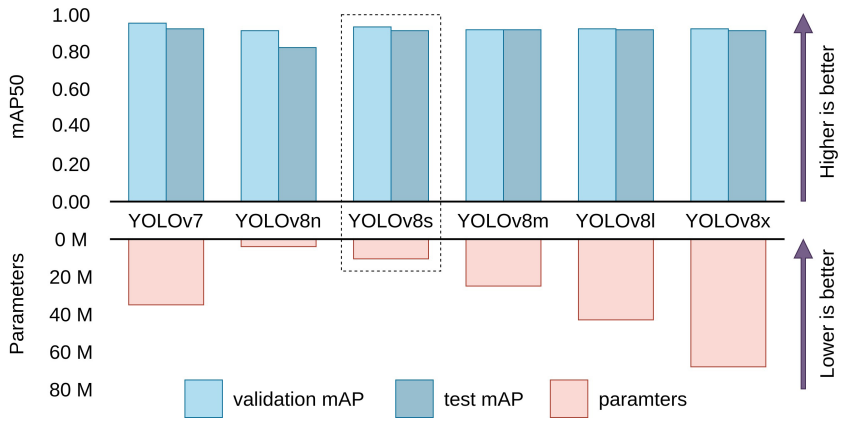


FIGURE 6
Comparison between performance and parameters of baseline models. YOLOv8s is evidently the most efficient model with similar validation and test performance as the other larger models. (Scales are relative).

TABLE 5 Performance and parameter count of Base YOLOv8s and YOLOv8s with extra *Conv* blocks on the validation and test dataset.

Model	Precision val | test	Recall val | test	mAP50 val | test	mAP50-95 val | test	Parameters
YOLOv8s	0.932 | 0.914	0.817 | 0.832	0.886 | 0.841	0.599 | 0.661	11.14 M
YOLOv8s+Conv	0.918 | **0.955**	0.657 | 0.837	0.715 | 0.905	0.545 | 0.688	11.20 M
YOLOv8s+Conv2	**0.943** | 0.901	0.829 | 0.852	0.894 | 0.894	0.670 | 0.694	11.26 M
YOLOv8s+Conv3	0.899 | 0.931	0.846 | 0.829	0.904 | **0.906**	**0.674** | **0.694**	11.32 M
YOLOv8s+Conv4	0.931 | 0.936	**0.852** | **0.859**	**0.905** | 0.903	0.657 | 0.686	11.38 M
YOLOv8s+Conv5	0.937 | 0.946	0.833 | 0.857	0.905 | 0.904	0.662 | 0.688	11.45 M

Bold values signify the value of the best performing model configuration under each metric.

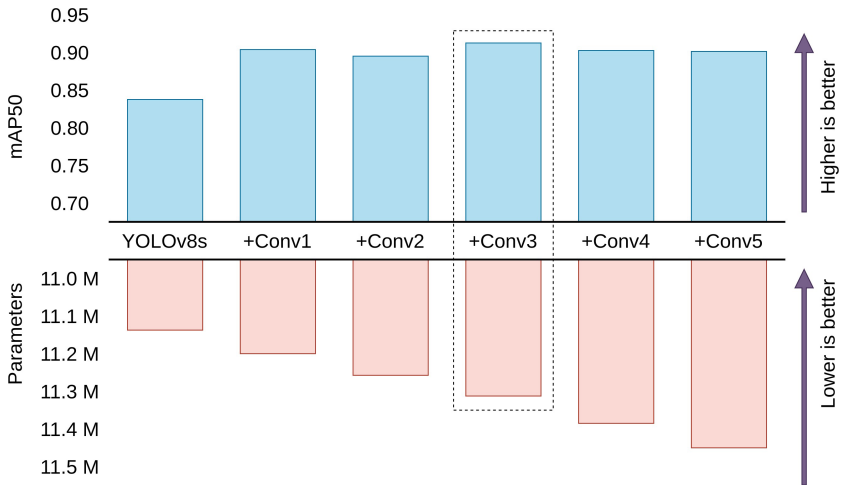


FIGURE 7
Comparison between performance and parameters of different head configurations. YOLOv8s with 3 extra *Conv* blocks is found to be the best model with the highest mean average precision for the test dataset. (Scales are relative).

striking a balance between leveraging the knowledge encoded in the pretrained layers and enabling the model to fine-tune its parameters for improved performance in the targeted cauliflower disease detection domain.

4.2.3 Comparison of different activation functions

Building upon the optimal configuration identified in the previous experiments, where 3 extra *Conv* blocks significantly improved overall performance in YOLOv8s with all layers sharing the same learning rate, we delved into the impact of altering the default activation function. The original YOLOv8 utilizes SiLU, also known as the swish activation function, as the default choice throughout the network. In this subsequent experimentation, we systematically replaced SiLU with alternative activation functions, such as ReLU, Leaky ReLU, Tanh, and Hard Swish, to discern their effects on model performance. The efficacy of Hard Swish can be attributed to its unique characteristics, combining non-linearity with bounded activation values. This enables the model to capture complex patterns while mitigating issues related to vanishing gradients or over-amplification of certain features. Table 7 summarizes the results of this experiment. Hard Swish was found to be the overall best performing activation function, which is also more computationally efficient than the default Swish since Hard Swish doesn't have to calculate a non-linear function like the sigmoid and has a linear mathematical definition which is better for reducing training and inference time.

4.2.4 Evaluation of the proposed model

The proposed model, derived from the extensive experimentation and fine-tuning process, was evaluated on the validation and test dataset to assess its performance comprehensively. Supplementary Figure 1 displays the normalized confusion matrices, Supplementary Figure 2 displays the precision-confidence curves, Supplementary Figure 3 displays the recall-confidence curve, Supplementary Figure 4 displays the F1-confidence curve, Supplementary Figure 5 displays the precision recall curves of our proposed model on the validation and test dataset respectively. Tables 8, 9 summarizes the performance of the proposed model on the validation and test dataset by class respectively. Supplementary Figure 6 displays the training and validation loss along with precision, recall, mAP50 and mAP50-95 progression over the training period.

5 Discussion

The conducted experiments focused on enhancing the YOLOv8 model for cauliflower disease detection, aiming for improved performance without significantly increasing parameters. Initial

TABLE 6 Comparison of different LR configurations on the validation dataset and test dataset.

Configuration	Precision val | test	Recall val | test	mAP50 val | test	mAP50-95 val | test
Default (YOLOv8s+Conv3)	0.899 | 0.931	**0.846** | 0.829	**0.904** | **0.906**	**0.674** | **0.694**
Freeze–Non–Extra–Conv	0.266 | 0.299	0.308 | 0.352	0.226 | 0.272	0.110 | 0.136
Freeze–Back	0.560 | 0.566	0.363 | 0.461	0.417 | 0.511	0.216 | 0.280
Fast–Extra–Conv	0.883 | 0.906	0.687 | 0.765	0.811 | 0.839	0.545 | 0.603
Fast–Head	0.853 | 0.920	0.750 | 0.778	0.830 | 0.846	0.549 | 0.615
Fast–Head–Neck	**0.938** | **0.955**	0.830 | **0.837**	0.895 | 0.905	0.659 | 0.688

Bold values signify the value of the best performing LR configuration under each metric.

TABLE 7 Performance summary of YOLOv8s with additional 3 Conv blocks and alternative activation functions on the validation dataset.

Model	Precision val | test	Recall val | test	mAP50 val | test	mAP50-95 val | test
YOLOv8s +Conv3+SiLU	0.899 | 0.931	0.846 | 0.829	0.904 | 0.906	0.674 | 0.694
YOLOv8s +Conv3+ReLU	0.905 | 0.906	0.815 | 0.829	0.878 | 0.875	0.626 | 0.668
YOLOv8s +Conv3+LeakyReLU	0.915 | 0.942	0.834 | **0.840**	0.892 | 0.900	0.642 | 0.677
YOLOv8s +Conv3+Tanh	0.822 | 0.906	0.623 | 0.702	0.726 | 0.771	0.434 | 0.502
YOLOv8s+Conv3 +Hard Swish	**0.919** | **0.932**	**0.851** | 0.826	**0.920** | **0.911**	**0.677** | **0.701**

Bold values signify the value of the best performing activation function configuration under each metric.

analysis showed that YOLOv7 and YOLOv8l had high mean average precision (mAP) but were impractical for lower-end devices due to large parameter counts. YOLOv8s, with a balanced trade-off between performance and parameters, was selected. Modifications involved adding extra *Conv* blocks to detection and classification heads was explored. Results indicated that incorporating three additional blocks yielded the best performance, with further augmentation leading to diminishing returns likely due to overfitting and redundant feature extraction. The study refined the training strategy, revealing that allowing all layers to freely adjust weights using the default learning rate achieved the optimal configuration. Freezing layers resulted in a decline in performance. The impact of activation functions was explored, with the default Swish emerging as the best-performing choice. In summary, systematic exploration led to an optimized YOLOv8s configuration, with three extra *Conv* blocks, balanced learning rates, and Swish as the activation function, demonstrating superior performance in cauliflower disease detection. However, limited disease variety in the dataset is a major limitation of this research which restricts the generalization scope of Cauli-Det, which included only three types of cauliflower diseases. Given the diverse range of diseases that can affect cauliflower leaves, flowers, and stems, the model's effectiveness may be limited when confronted with other diseases not represented in the dataset. The study focused on optimizing the YOLOv8s model for deployment on lower-end devices. However, the performance evaluation may not fully capture the nuances of diverse hardware configurations. The model's efficiency and accuracy could vary on different devices, and further research may be required to fine-tune the model for optimal performance across a wider range of computing resources. The evaluation of the model's performance was primarily conducted based on offline analysis of collected data. Real-time evaluation, crucial for practical deployment in agriculture, was not explicitly addressed in this research. The model's responsiveness to dynamic changes in the field, such as disease progression or plant growth, remains an unexplored aspect.

6 Conclusion

This research offers a tailored YOLOv8s model specifically designed for detecting prevalent cauliflower diseases, addressing unique challenges in disease identification. Additionally, it provides a comprehensive evaluation of base YOLOv8 models on cauliflower disease datasets, highlighting baseline performance and paving the way for systematic model modifications. By systematically applying adjustments to enhance detection accuracy and average precision, the study offers valuable insights into improving the model's ability to classify cauliflower diseases effectively. Furthermore, the paper contributes to the research community by providing open access to an annotated dataset and the proposed model, fostering reproducibility and facilitating further advancements in computer vision applied to agriculture. We identified the limitations of the proposed model and for future research approaches that may build upon this work to build a better and more capable cauliflower disease detection approach, we plan to address more diseases than the three that were present in the dataset used in this work (Downey Mildew, Black Rot, and Bacterial Soft Rot). We also believe that the success of any disease detection model will be dependent on its real-world applicability, which will rely on demonstrating the models performance on lower end devices which are most likely to be available on the hands of crop farmers. Additionally, another

TABLE 8 Performance summary of the proposed model on the validation dataset by class.

Class	Images	Instances	Prec	Rec	AP50	AP50-95
All	98	332	0.919	0.851	0.920	0.677
Downey Mildew	26	77	0.919	0.844	0.941	0.703
Black Rot	15	215	0.852	0.758	0.845	0.532
Bacterial Spot Rot	26	40	0.987	0.950	0.974	0.797
Healthy	31	0	–	–	–	–

TABLE 9 Performance summary of the proposed model on the test dataset by class.

Class	Images	Instances	Prec	Rec	AP50	AP50-95
All	98	309	0.932	0.826	0.911	0.701
Downey Mildew	26	50	0.901	0.840	0.925	0.686
Black Rot	15	225	0.926	0.667	0.826	0.552
Bacterial Spot Rot	26	34	0.968	0.971	0.983	0.864
Healthy	31	0	–	–	–	–

challenge is the evaluation of these techniques in real-time disease detection, for which video datasets will be required. In a real-time system it is crucial to demonstrate the model's responsiveness to dynamic changes in the field. Drones may be employed to capture live video feed from cauliflower fields which then may be streamed into a model with fast inference to demonstrate the model's real-time disease detection capabilities. Some cauliflower diseases are more prevalent than others depending on region and climate. Conducting research on these diverse set of circumstances can help other researchers on making informed choices when developing tailored disease detection models that are fit for addressing disease management problems according to region, climate and the specific needs of cauliflower plantations. Lastly, We believe that the proposed model will be a valuable addition to the field of disease detection for the domain of precision agriculture.

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. The source code for the modified YOLOv8 network is available at: https://github.com/manchitro/vegnet-yolov8. Best weights are available at: https://github.com/manchitro/vegnet-yolov8/releases.

Author contributions

MU: Conceptualization, Writing – original draft. MKAM: Data curation, Methodology, Writing – original draft. AP: Formal analysis, Validation, Writing – original draft. MFM: Writing – review & editing. MS: Supervision, Validation, Writing – review & editing. SA: Funding acquisition, Visualization, Writing – review & editing. DC: Formal analysis, Validation, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research is funded by the Researchers Supporting Project Number (RSPD2024R890), King Saud University, Riyadh, Saudi Arabia.

Acknowledgments

The authors extend their appreciation to King Saud University for funding this research through Researchers Supporting Project Number (RSPD2024R890), King Saud University, Riyadh, Saudi Arabia.

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/fpls.2024.1373590/full#supplementary-material

References

Abdul Malek, M., Reya, S. S., Zahan, N., Zahid Hasan, M., and Uddin, M. S. (2022). Deep learning-based cauliflower disease classification. in *Computer Vision and Machine Learning in Agriculture*, vol. 2. (Singapore: Springer), 171–186. doi: 10.1007/978-981-16-9991-7_11

Arun, R. A., and Umamaheswari, S. (2023). Effective multi-crop disease detection using pruned complete concatenated deep learning model. *Expert Syst. Appl.* 213, 118905. doi: 10.1016/j.eswa.2022.118905

Bangladesh Bureau of Statistics (2023) *Gdp*. Available online at: https://bbs.gov.bd/site/page/dc2bc6ce-7080-48b3-9a04-73cec782d0df/Gross-Domestic-Product-(GDP) (Accessed November 29, 2023).

Chug, A., Bhatia, A., Singh, A. P., and Singh, D. (2023). A novel framework for image-based plant disease detection using hybrid deep learning approach. *Soft Comput.* 27, 13613–13638. doi: 10.1007/s00500-022-07177-7

Eunice, J., Popescu, D. E., Chowdary, M. K., and Hemanth, J. (2022). Deep learning-based leaf disease detection in crops using images for agricultural applications. *Agronomy* 12, 2395. doi: 10.3390/agronomy12102395

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognition* 77, 354–377. doi: 10.1016/j.patcog.2017.10.013

Haridasan, A., Thomas, J., and Raj, E. D. (2023). Deep learning system for paddy plant disease detection and classification. *Environ. Monit. Assess.* 195, 120. doi: 10.1007/s10661-022-10656-x

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

Helgi Library (2022) *Which country produces the most cauliflower?* Available online at: https://www.helgilibrary.com/charts/which-country-produces-the-most-cauliflower/ (Accessed November 27, 2023).

Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., et al. (2019). Searching for mobilenetv3. *Proceedings of the IEEE/CVF international conference on computer vision*. (pp. 1314–1324).

Hu, K., Liu, Y., Nie, J., Zheng, X., Zhang, W., Liu, Y., et al. (2023a). Rice pest identification based on multi-scale double-branch gan-resnet. *Front. Plant Sci.* 14, 1167121. doi: 10.3389/fpls.2023.1167121

Hu, X., Wang, R., Du, J., Hu, Y., Jiao, L., and Xu, T. (2023b). Class-attention-based lesion proposal convolutional neural network for strawberry diseases identification. *Front. Plant Sci.* 14, 1091600. doi: 10.3389/fpls.2023.1091600

Huang, X., Chen, A., Zhou, G., Zhang, X., Wang, J., Peng, N., et al. (2023). Tomato leaf disease detection system based on fc-sndpn. *Multimedia Tools Appl.* 82, 2121–2144. doi: 10.1007/s11042-021-11790-3

Jocher, G., Chaurasia, A., and Qiu, J. (2023) *Ultralytics YOLOv8*. Available online at: https://github.com/ultralytics/ultralytics.

Kanna, G. P., Kumar, S. J., Kumar, Y., Changela, A., Woźniak, M., Shafi, J., et al. (2023). Advanced deep learning techniques for early disease prediction in cauliflower plants. *Sci. Rep.* 13, 18475. doi: 10.1038/s41598-023-45403-w

Khan, A., Sohail, A., Zahoora, U., and Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artif. Intell. Rev.* 53, 5455–5516. doi: 10.1007/s10462-020-09825-6

Li, Y., Xue, J., Wang, K., Zhang, M., and Li, Z. (2022). Surface defect detection of fresh-cut cauliflowers based on convolutional neural network with transfer learning. *Foods* 11, 2915. doi: 10.3390/foods11182915

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. in *Proceedings of the IEEE international conference on computer vision*. (pp. 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 *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, vol. 13. (Springer International Publishing), (pp, 740–755).

Lin, J., Yu, D., Pan, R., Cai, J., Liu, J., Zhang, L., et al. (2023). Improved yolox-tiny network for detection of tobacco brown spot disease. *Front. Plant Sci.* 14, 1135105. doi: 10.3389/fpls.2023.1135105

Liu, Z., Wang, H., Wang, J., Lv, J., Xie, B., Luo, S., et al. (2022). Physical, chemical, and biological control of black rot of brassicaceae vegetables: A review. *Front. Microbiol.* 13, 1023826. doi: 10.3389/fmicb.2022.1023826

Mallick, M. T., Biswas, S., Das, A. K., Saha, H. N., Chakrabarti, A., and Deb, N. (2023). Deep learning based automated disease detection and pest classification in Indian mung bean. *Multimedia Tools Appl.* 82, 12017–12041. doi: 10.1007/s11042-022-13673-7

Maria, S. K., Taki, S. S., Mia, M. J., Biswas, A. A., Majumder, A., and Hasan, F. (2022). Cauliflower disease recognition using machine learning and transfer learning. in *Smart Systems: Innovations in Computing: Proceedings of SSIC 2021* (Springer) (pp. 359–375).

Masood, M., Nawaz, M., Nazir, T., Javed, A., Alkanhel, R., Elmannai, H., et al. (2023). Maizenet: A deep learning approach for effective recognition of maize plant leaf diseases. *IEEE Access* vol. 11, pp. 52862-52876. doi: 10.1109/ACCESS.2023.3280260

Muimba-Kankolongo, A. (2018). "Chapter 11 - vegetable production", in *Food Crop Production by Smallholder Farmers in Southern Africa*. Ed. A. Muimba-Kankolongo (Cambridge, MA, USA: Academic Press), pp. 205–274. doi: 10.1016/B978-0-12-814383-4.00011-6

Pathak, V. M., Verma, V. K., Rawat, B. S., Kaur, B., Babu, N., Sharma, A., et al. (2022). Current status of pesticide effects on environment, human health and it's eco-friendly management as bioremediation: A comprehensive review. *Front. Microbiol.* 13, p. 962619. doi: 10.3389/fmicb.2022.962619

Pourdarbani, R., and Sabzi, S. (2023). Diagnosis of common cauliflower diseases using image processing and deep learning. *J. Environ. Sci. Stud.* 8, 7087–7092. doi: 10.22034/JESS.2023.391624.1995

Rajbongshi, A., Islam, M., Mia, M., Sakif, T., and Majumder, A. (2022). A comprehensive investigation to cauliflower diseases recognition: an automated machine learning approach. *Int. J. Adv. Sci. Eng. Inform. Technol.(IJASEIT)* 12, 32–41. doi: 10.18517/ijaseit.12.1.15189

Ramachandran, P., Zoph, B., and Le, Q. V. (2017). Searching for activation functions. *arXiv*. doi: 10.48550/arXiv.1710.05941

Sara, U., Rajbongshi, A., Shakil, R., Akter, B., and Uddin, M. S. (2022). Vegnet: An organized dataset of cauliflower disease for a sustainable agro-based automation system. *Data in Brief*, 43, p.108422. doi: 10.1016/j.dib.2022.108422

Shakil, R., Akter, B., Shamrat, F. J. M., and Noori, S. R. H. (2023). A novel automated feature selection based approach to recognize cauliflower disease. *Bull. Electrical Eng. Inf.* 12, 3541–3551. doi: 10.11591/eei.v12i6.5359

Sheng, X. G., Branca, F., Zhao, Z. Q., Wang, J. S., Yu, H. F., Shen, Y. S., et al. (2020). Identification of black rot resistance in a wild brassica species and its potential transferability to cauliflower. *Agronomy* 10 (9), p.1400. doi: 10.3390/agronomy10091400

Singh, B., Singh, B., and Singh, P. (2018). Breeding cauliflower: A review. *Int. J. Vegetable Sci.* 24, 58–84. doi: 10.1080/19315260.2017.1354242

Statistics Times (2018) *List of countries by gdp sector composition*. Available online at: https://statisticstimes.com/economy/countries-by-gdp-sector-composition.php (Accessed December 15, 2023).

Tao, H., Miao, H., Chen, L., Wang, M., Xia, C., Zeng, W., et al. (2022). Wrky33-mediated indolic glucosinolate metabolic pathway confers resistance against alternaria brassicicola in arabidopsis and brassica crops. *J. Integr. Plant Biol.* 64, 1007–1019. doi: 10.1111/jipb.13245

Teet, S. E., and Hashim, N. (2023). Recent advances of application of optical imaging techniques for disease detection in fruits and vegetables: A review. *Food Control* 152, 109849. doi: 10.1016/j.foodcont.2023.109849

Thakur, P. S., Sheorey, T., and Ojha, A. (2023). Vgg-icnn: A lightweight cnn model for crop disease identification. *Multimedia Tools Appl.* 82, 497–520. doi: 10.1007/s11042-022-13144-z

The Global Economy (2021) *Bangladesh: Employment in agriculture*. Available online at: https://www.theglobaleconomy.com/Bangladesh/Employment_in_agriculture/ (Accessed December 20, 2023).

Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2023b). Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. (pp. 7464–7475).

Wang, H., Chang, W., Yao, Y., Yao, Z., Zhao, Y., Li, S., et al. (2023a). Cropformer: A new generalized deep learning classification approach for multi-scenario crop classification. *Front. Plant Sci.* 14, 1130659. doi: 10.3389/fpls.2023.1130659

Wang, C. Y., Liao, H. Y. M., 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. (pp. 390–391).

Zhao, Y., Yang, Y., Xu, X., and Sun, C. (2023). Precision detection of crop diseases based on improved yolov5 model. *Front. Plant Sci.* 13, 1066835. doi: 10.3389/fpls.2022.1066835

Zhu, S., Ma, W., Wang, J., Yang, M., Wang, Y., and Wang, C. (2023). Eadd-yolo: An efficient and accurate disease detector for apple leaf using improved lightweight yolov5. *Front. Plant Sci.* 14, 1120724. doi: 10.3389/fpls.2023.1120724

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 07 May 2024
DOI 10.3389/fpls.2024.1366395

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Jana Shafi,
Prince Sattam Bin Abdulaziz University, Saudi Arabia
Marcin Wozniak,
Silesian University of Technology, Poland

*CORRESPONDENCE
Toktam Zoughi
t.zoughi@shariaty.ac.ir

RECEIVED 06 January 2024
ACCEPTED 17 April 2024
PUBLISHED 07 May 2024

CITATION
Afsharpour P, Zoughi T, Deypir M and Zoqi MJ (2024) Robust deep learning method for fruit decay detection and plant identification: enhancing food security and quality control.
Front. Plant Sci. 15:1366395.
doi: 10.3389/fpls.2024.1366395

COPYRIGHT
© 2024 Afsharpour, Zoughi, Deypir and Zoqi. 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.

Robust deep learning method for fruit decay detection and plant identification: enhancing food security and quality control

Pariya Afsharpour[1], Toktam Zoughi[1*], Mahmood Deypir[2] and Mohamad Javad Zoqi[3]

[1]Department of Electrical and Computer Engineering, Shariaty College, Technical and Vocational University (TVU), Tehran, Iran, [2]Faculty of Computer Engineering, Shahid Sattari Aeronautical University of Science and Technology, Tehran, Iran, [3]Department of Civil Engineering, Faculty of Engineering, University of Birjand, Birjand, Iran

This paper presents a robust deep learning method for fruit decay detection and plant identification. By addressing the limitations of previous studies that primarily focused on model accuracy, our approach aims to provide a more comprehensive solution that considers the challenges of robustness and limited data scenarios. The proposed method achieves exceptional accuracy of 99.93%, surpassing established models. In addition to its exceptional accuracy, the proposed method highlights the significance of robustness and adaptability in limited data scenarios. The proposed model exhibits strong performance even under the challenging conditions, such as intense lighting variations and partial image obstructions. Extensive evaluations demonstrate its robust performance, generalization ability, and minimal misclassifications. The inclusion of Class Activation Maps enhances the model's capability to identify distinguishing features between fresh and rotten fruits. This research has significant implications for fruit quality control, economic loss reduction, and applications in agriculture, transportation, and scientific research. The proposed method serves as a valuable resource for fruit and plant-related industries. It offers precise adaptation to specific data, customization of the network architecture, and effective training even with limited data. Overall, this research contributes to fruit quality control, economic loss reduction, and waste minimization.

KEYWORDS

convolutional neural networks, fruit quality control, deep learning, class activation maps, data augmentation, model generalizability, robustness

1 Introduction

Fruit decay detection and plant identification are crucial aspects in agricultural and horticultural practices, playing a significant role in ensuring crop quality, disease control, and overall productivity (Pessarakli, 1994; Jayasena et al., 2015). Detecting fruit decay accurately and in a timely manner minimizes post-harvest losses, ensures food safety, and optimizes storage and distribution processes (Pessarakli, 1994). Additionally, early detection allows for prompt actions such as sorting and removal, preventing the spread of diseases and preserving the quality of the remaining fruits (Pessarakli, 1994).

Automated fruit decay detection systems based on computer vision and machine learning techniques have demonstrated promising results in terms of accuracy, speed, and cost-effectiveness (Jayasena et al., 2015; Boulent et al., 2019; Lakshmanan, 2019). These systems contribute to reducing post-harvest losses, optimizing storage conditions, and enhancing the overall efficiency of the fruit supply chain.

Plant identification is equally important and serves various purposes in agricultural practices. Accurate identification of plant species and cultivars aids in selecting appropriate varieties for specific environments, optimizing cultivation techniques, and improving agricultural practices (Barbedo, 2018; Wäldchen and Mader, 2018). Furthermore, plant identification plays a vital role in effective pest management by enabling timely and targeted application of control measures (Ferentinos, 2018). It also contributes to biodiversity conservation efforts by facilitating the monitoring and preservation of endangered plant species (Kaur and Kaur, 2019).

Technological advancements, particularly in computer vision, machine learning, and image processing, have greatly facilitated fruit decay detection and plant identification (Barbedo, 2018; Boulent et al., 2019). Computer vision techniques, including feature extraction, pattern recognition, and deep learning algorithms, have proven to be highly effective in automating these tasks. By analyzing images or sensor data captured from fruits or plants, these systems accurately identify signs of decay and classify plant species, even in large-scale agricultural settings (Goodfellow, 2016).

In summary, fruit decay detection and plant identification hold paramount importance in agricultural and horticultural practices. The ability to promptly and accurately detect fruit decay minimizes post-harvest losses and ensures food safety. Similarly, precise plant identification contributes to cultivar selection, pest management, and biodiversity conservation. The integration of computer vision and machine learning techniques has opened up new avenues for developing automated systems that enhance the efficiency, productivity, and sustainability of agricultural processes (Jayasena et al., 2015; Barbedo, 2018; Ferentinos, 2018; Wäldchen and Mader, 2018; Boulent et al., 2019; Kaur and Kaur, 2019).

1.1 The importance of fruit decay detection

In the field of artificial intelligence and deep learning, the detection and analysis of fruit decay and plant identification hold significant importance. Fruit decay is recognized as one of the major challenges in the agricultural product supply chain and the agriculture industry. This decay not only has negative effects on human health and nutrition but also poses a serious problem in the management of agricultural product supply chains. Fruit decay results in significant losses of agricultural products, leading to substantial economic damages for producers and various industries (Brownlee, 2018a; Lewis, 2022; Norman, 2019).

To address these challenges, the use of advanced technologies, such as convolutional neural networks (CNNs) and deep learning algorithms, has gained considerable attention. These techniques offer the potential to develop intelligent models capable of accurately detecting and classifying spoiled fruits with high precision and accuracy. By leveraging the power of artificial intelligence, it becomes possible to enhance fruit quality control, improve supply chain management, and minimize economic losses caused by fruit decay. This article aims to explore the application of CNNs and deep learning techniques in fruit decay detection and plant identification. It provides an overview of the importance of distinguishing between spoiled and non-spoiled fruits, highlighting the negative impacts on human health, nutrition, and the management of agricultural product supply chains (Shahid, 2019; Lewis, 2022). The research presented in this article draws upon previous studies and developments in the field of artificial intelligence and deep learning, with a focus on addressing the challenges associated with fruit decay detection (Sonwani et al., 2022). By examining the existing literature and presenting empirical evidence, this article aims to contribute to the body of knowledge on fruit decay detection and its significance in the agricultural industry. The findings of this research have implications for improving fruit quality, optimizing supply chain processes, and minimizing economic losses for producers and stakeholders in the agriculture sector.

1.2 The need for a better model

Extensive research has been conducted in the field of fruit detection and imaging using convolutional neural networks (CNNs). These studies aim to develop more accurate methods for distinguishing between spoiled and healthy fruits. However, some of these research efforts have not yielded satisfactory results due to limitations and shortcomings. Therefore, there is a need to enhance and improve existing models in this area. The model presented in this article incorporates enhancements and innovations that provide a better response to the requirements of fruit decay detection. The existing models in fruit decay detection have faced challenges related to accuracy and performance. Some models struggled to accurately classify fruits based on their decay level or distinguish between different types of spoilage. These limitations have hindered the effectiveness of fruit quality control and supply chain management processes. Consequently, there is a demand for a more robust and efficient model that can address these shortcomings and deliver improved results.

The model proposed in this article introduces several advancements to overcome the limitations of previous

approaches. It leverages state-of-the-art CNN architectures to improve the accuracy of fruit decay detection. Additionally, novel data augmentation techniques are employed to enhance the model's ability to generalize and adapt to different fruit varieties and decay patterns. To validate the effectiveness of the proposed model, extensive experiments conducted using datasets comprising various fruit types and decay stages. The results demonstrate significant improvements in the accuracy and reliability of fruit decay detection compared to previous methods. The enhanced model is not only achieving higher precision in classifying spoiled and non-spoiled fruits but also exhibits robustness in real-world scenarios, making it a practical solution for fruit quality control and supply chain optimization. With its superior accuracy, this new network can perform effectively and adapt well in various conditions, offering precise and customized performance.

2 Related work

CNNs are a class of deep learning models specifically designed to analyze and extract meaningful features from images (Alex et al., 2012; Simonyan and Zisserman, 2014). They have achieved remarkable success in various computer vision tasks, including image classification, object detection, and semantic segmentation.

The hierarchical nature of CNNs allows them to excel in image classification tasks. By learning increasingly complex features through multiple layers, CNNs can effectively differentiate between different objects or classes in images (Alex et al., 2012). This capability makes CNNs particularly valuable in applications such as image recognition and categorization.

Furthermore, CNNs have shown great potential in semantic segmentation, where the goal is to assign a class label to each pixel in an image. Fully convolutional networks (FCNs), an extension of CNNs, have been specifically designed for this task and have achieved impressive results (Long et al., 2015). FCNs preserve spatial information throughout the network, enabling pixel-wise predictions and facilitating accurate segmentation of objects and regions within images.

In summary, Convolutional Neural Networks have revolutionized image processing and object detection. Their ability to automatically learn hierarchical representations from images, combined with their flexibility in handling various computer vision tasks, has made CNNs indispensable in the field. By leveraging the power of deep learning, CNNs have significantly advanced image understanding, object localization, and semantic segmentation (Girshick et al., 2014).

In the field of image processing, and Fruit Decay Detection several deep learning architectures have made significant contributions. Here, let's compare some of the most important deep learning models specifically relevant to image processing: Convolutional Neural Networks (CNNs): CNNs have become the cornerstone of image processing tasks. They excel at capturing spatial hierarchies and local features through convolutional layers. CNNs have achieved remarkable success in image classification, object detection, image segmentation, and various other computer vision tasks (LeCun et al., 2015). One of the most influential CNN architectures is the VGGNet, which introduced deeper networks with smaller filters, showcasing impressive performance (Simonyan and Zisserman, 2014).

Residual Neural Networks (ResNets) addressed the challenge of training very deep neural networks by introducing skip connections. These connections enable the network to learn residual mappings, allowing for the training of deeper architectures without degradation in performance. ResNets demonstrated superior performance in image classification and won the ImageNet challenge in 2015 (He et al., 2016). U-Net is a popular architecture for image segmentation tasks, particularly in biomedical image analysis. It consists of an encoder-decoder structure with skip connections that enable precise localization of segmentation boundaries. U-Net has been widely adopted in tasks such as medical image segmentation, cell counting, and semantic segmentation (Ronneberger et al., 2015; JananiSBabu, 2020).

Generative Adversarial Networks (GANs) have had a significant impact on image generation and synthesis. They consist of a generator network that produces synthetic images and a discriminator network that distinguishes between real and generated images. GANs have been successful in generating realistic images, image-to-image translation, and style transfer tasks (Goodfellow et al., 2014). An influential GAN architecture is the Progressive Growing of GANs (PGGAN), which progressively grows the resolution of generated images, resulting in high-quality outputs (Karras et al., 2018). EfficientNet is a deep learning architecture that has gained attention for its impressive performance and efficiency. It uses a compound scaling method to balance model size and computational resources, achieving state-of-the-art results with fewer parameters. EfficientNet has shown remarkable performance in image classification tasks, surpassing previous models while being computationally efficient (Tan and Le, 2019). These are just a few examples of influential deep learning architectures in image processing and Fruit Decay Detection. It's worth noting that the field is continually evolving, and new architectures are being introduced regularly, pushing the boundaries of image analysis and understanding.

2.1 Fruit decay detection using CNNs

Several studies have explored the application of convolutional neural networks (CNNs) in fruit decay detection. These studies aim to develop intelligent models capable of accurately distinguishing between spoiled and healthy fruits based on image analysis (Selvaraj et al., 2019).

One notable research effort by Zhang et al. (2020) utilized a CNN model to detect fruit decay in apples. The model achieved a high accuracy of 94.5% in classifying healthy and decayed apples. However, this study focused on a specific fruit type and did not consider other varieties or spoilage types (Fan et al., 2020).

Another study by Li et al. (2019) employed a deep learning model based on a pre-trained CNN architecture to classify different types of fruit decay. The model achieved an accuracy of 92.7% in detecting four kinds of fruit decay. However, this research also focused on a limited range of fruit types and spoilage categories.

The other study proposed DeepFruits, where is a fruit detection system using deep neural networks (Sa et al., 2016). DeepFruits is a notable model developed for fruit detection, which capitalizes on convolutional neural networks (CNNs) and transfer learning through VGG16 network architecture. The system also includes image preprocessing algorithms and neural networks for decision-making, amplifying its performance. Nevertheless, the DeepFruits model encounters challenges with images captured under varying environmental conditions. Those indicates potential performance disruptions when processing photos taken under diverse lighting conditions, backgrounds, or perspectives. Also, the model necessitates abundant training data and considerable computational resources, which might restrict its practical application. DeepFruits represents a significant stride in fruit detection using deep neural networks, but its limitations necessitate further research to augment its robustness in diverse environmental settings (Sa et al., 2016). Another study addresses the challenges in cauliflower disease identification and detection, emphasizing the role of advanced deep transfer learning techniques in automating the process and benefiting agricultural management (Kanna et al., 2023).

FruitDetect is the other model that detects fruit using convolutional neural networks. FruitDetect exploits a convolutional neural network (CNN) and transfer learning with VGG16 to detect and identify fruits. Despite demonstrating accurate fruit detection, the model's limitations include a potential need for larger training datasets to enhance its final accuracy (Faouzi, 2021).

The project "Melanoma Detection using ResNet50" leverages the ResNet50 neural network for melanoma detection. Despite its accurate detection of melanoma, the model might struggle with variable conditions that could affect skin disease detection (Scarlat, 2018; Ramya, 2023).

To overcome these limitations, the proposed method aims to develop a more comprehensive and accurate model for fruit decay detection. The model considers a wider range of fruit types and spoilage categories, enabling it to provide more robust and versatile results.

2.2 Plant identification using CNNs

Plant identification is another area where CNNs have shown promising results. Several studies have demonstrated the effectiveness of CNN models in accurately classifying different plant species based on leaf and flower images. For instance, a study by Wäldchen and Mäder (2018) utilized a CNN model for plant species identification. The model achieved a high accuracy of 98.53% in classifying 1000 different plant species. This research demonstrated the potential of CNNs in plant identification and highlighted the importance of high-quality datasets for training and testing (Ren et al., 2015; Wäldchen and Mader, 2018).

The other study is Deep Learning-Based Banana Plant Diseases and Pest Detection. This study explores the application of deep learning to detect banana plant diseases and pests. The method uses transfer learning with ResNet and InceptionV2 neural networks, enhancing disease detection accuracy and efficiency. Even though the method demonstrates high accuracy and the ability to detect various diseases with fewer errors, it might require more substantial training data and improved adaptation to real-world conditions. The combination of ResNet and InceptionV2 neural networks, with ResNet handling deep networks and InceptionV2 facilitating efficient feature extraction, contribute to the model's improved performance. However, the model's applicability is restricted to bananas, and practical implementation may require access to infrared imaging equipment. The study suggests promising results for disease and pest detection in banana plants but calls for additional research to overcome limitations and expand the model's applicability across diverse fruits and crops (Brital, 2021; Narayanam, 2022). Another study proposes an integrated IoT and deep learning framework, the 'Automatic and Intelligent Data Collector and Classifier', for automating plant disease detection in pearl millet, providing a low-cost and efficient tool to improve crop yield and product quality (Kundu et al., 2021). The other research provides a comprehensive survey of the application of deep Convolutional Neural Networks in plant disease prediction from leaf images, offering valuable insights into pre-processing techniques, models, frameworks, optimization methods, datasets, and performance metrics for researchers in the field of agricultural deep learning (Dhaka et al., 2021).

In line with these findings, the proposed research incorporates plant identification capabilities into the fruit decay detection model. By leveraging the power of CNNs, the model can accurately identify different plant species, providing additional value and applications in the field of agriculture.

3 Methodology

3.1 Convolutional neural networks

Convolutional Neural Networks (CNNs) have demonstrated superiority in various image processing tasks compared to other network architectures. Here are some key reasons why CNNs are often preferred. CNNs are designed to exploit the local spatial correlations present in images. Through their convolutional layers, CNNs learn to capture local patterns and features, allowing them to effectively model image structures. This local receptive field property enables CNNs to extract meaningful information from images efficiently (LeCun et al., 2015).

CNNs use parameter sharing, which significantly reduces the number of parameters compared to fully connected networks. By sharing weights across different spatial locations, CNNs can learn spatial hierarchies and generalize well to new images. This property makes CNNs more efficient and less prone to overfitting (LeCun et al., 1998). CNNs possess translation invariance, meaning they can recognize patterns regardless of their position in an image. This property is crucial for tasks such as object recognition, where the location of an object may vary. CNNs' ability to extract features invariant to translation makes them robust to changes in object position or image transformations (LeCun et al., 2010). CNNs are composed of multiple layers, with each layer learning increasingly

complex and abstract features. The initial layers capture low-level features like edges and textures, while deeper layers learn high-level representations such as object parts or whole objects. This hierarchical feature extraction allows CNNs to capture both fine-grained details and global context, leading to improved performance in complex image analysis tasks (Zeiler and Fergus, 2014). CNNs have benefited from the availability of large-scale pretraining datasets, such as ImageNet. Pretrained CNN models can be fine-tuned or used as feature extractors in various domains with limited labeled data. Transfer learning with CNNs has proven effective in tasks where training data is scarce, accelerating model development and achieving good performance (Yosinski et al., 2014).

While CNNs have demonstrated superiority in image processing and Fruit Decay Detection tasks, it's important to note that the choice of network architecture depends on the specific task, dataset, and computational resources available. Different architectures may have their own advantages in specialized scenarios.

The main objective of this paper is to present an intelligent and accurate method for detecting spoiled and healthy fruits using an advanced 11-layer convolutional neural network (CNN). This novel and advanced approach has been implemented and optimized using the TensorFlow library for deep learning. It starts with collecting and preprocessing a diverse dataset of fruit images, including both healthy and spoiled fruits. The dataset is carefully labeled to ensure accurate classification during the training and testing phases. Next, an advanced 11-layer CNN model is designed and implemented using the TensorFlow library. This model incorporates multiple convolutional and pooling layers, along with fully connected layers for classification. To further improve the model's performance, data augmentation techniques are employed to increase the diversity and size of the training dataset. This helps the model learn robust features and reduces the risk of overfitting. Once the model is developed and optimized, extensive experiments are conducted to evaluate its performance in fruit decay detection and plant identification. The model's accuracy, and confusion matrix are measured to assess its effectiveness. The results of the experiments demonstrate the superior performance of the proposed model compared to existing approaches. The model achieves high precision and accuracy in classifying spoiled and healthy fruits, as well as accurately identifying different plant species.

Convolutional Neural Networks (CNNs) have emerged as a crucial neural network structure for image processing and pattern recognition tasks. They are specifically designed to process grid-like data, such as images, by extracting hierarchical features (Gupta, 2021; Kalra, 2023). Convolutional layers, the primary feature of these networks, detect diverse patterns in images by applying convolutional filters, thereby extracting various features, including edges, corners, and similar patterns. The extracted features are then utilized by the fully connected layers for final decision-making. Figure 1 shows, an overview of the proposed network. This figure illustrates two representative layers of our proposed method, each followed by a set of fully connected layers. The first layer consists of a convolution sublayer and a pooling sublayer, while the second layer shares the same structure but with different dimensions. It is important to note that this figure does not depict all 11 layers of our model. We have nine additional layers with similar configurations but varying dimensions, which are not explicitly shown in this figure. After the 11 convolutional layers, we include three fully connected layers, followed by the output of the model.

The presented approach employs sublayers based on convolution in the initial stage, followed by utilization of the maximum operation on the outcome of the convolutional sublayer, establishing a connection to the pooling sublayer. To ensure optimal performance, the suggested technique integrates batch normalization (BN) and applies the Rectified Linear Unit (ReLU) activation function subsequent to the pooling operations. After the convolutional function, the BN and ReLU are implemented within this framework. Subsequently, fully connected layers are integrated to amalgamate features from diverse frequency bands. The concluding layer in the network

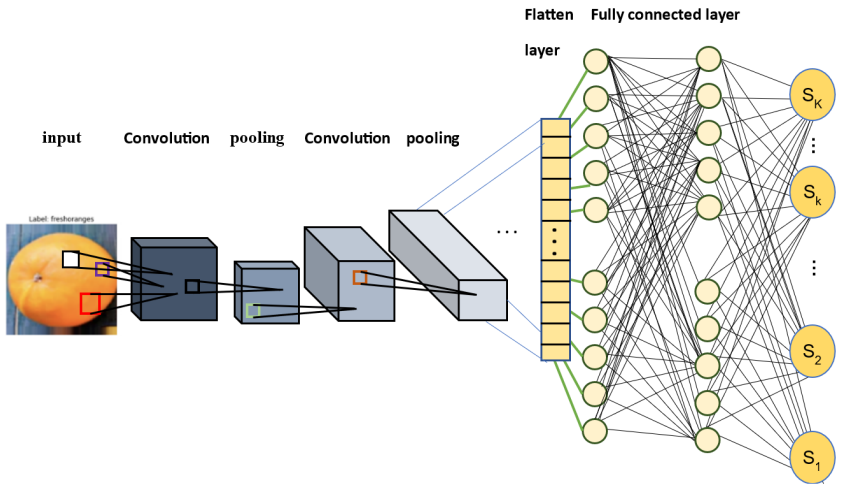


FIGURE 1
Overview of the proposed method.

employs the SoftMax function to compute the fruit class. The proposed approach trains the entire deep neural network employing the back-propagation algorithm.

3.1.1 Convolutional layers

The convolutional layer, the fundamental building block of a CNN, employs filters to scour for patterns aiding in image detection and classification. For instance, a filter designed to detect a face might capture the pattern of identifying eyes within the input mass.

3.1.2 Max pooling layers

Max pooling layers reduce the dimensions of images, discarding superfluous information by selecting the maximum values within each input region (Dertat, 2017; Editorial, 2022).

3.1.3 Fully connected layers

Once all the image features have been extracted, these are forwarded to the fully connected layers. An activation function transforms the feature information into a feature vector (Valliappa Lakshmanan, 2021).

3.1.4 Activation functions

CNNs utilize non-linear activation functions to introduce non-linearity into the network. Depending on the coding environment, linear functions are either defined separately or in conjunction with the convolutional layers. Activation functions modulate the product of the filter and input mass based on their unique characteristics. Among them, the Rectified Linear Unit (ReLU) activation function, which zeroes out any negative input while keeping positive inputs unchanged, is highly favored due to its computational efficiency (Brownlee, 2018b). In many instances, the SoftMax activation function is used as the activation function for the final layer.

3.1.5 Batch normalization and dropout layers

Batch Normalization and Dropout layers are employed to prevent overfitting and stabilize transitions between layers. Batch Normalization layers normalize input data and ensure optimal distributions. Conversely, Dropout layers randomly deactivate neurons during each training iteration, preventing over-reliance on specific neurons and maintaining a balance between neurons and features (Brownlee, 2018b).

3.2 Model construction

The proposed model offers several advantages and holds significant value. The goals of this paper encompass several key objectives. Firstly, the aim is to develop a customized 11-layer CNN model that surpasses the performance of existing models like VGG-16, VGG-19, LeNet-5, and AlexNet in fruit quality classification, as well as outperform transfer learning methods such as VGG16 and ResNet. Secondly, measures are implemented to counteract overfitting through data augmentation and early stopping mechanisms, ensuring the model's ability to generalize and maintain robustness across different datasets. The paper also focuses on enhancing visual explain ability by integrating Class Activation Maps (CAMs), which improve interpretability and credibility of the model's predictions. Additionally, the construction of a robust and versatile model is emphasized, validated through confusion matrix analysis to highlight its efficacy in making precise predictions and accurately identifying both spoiled fruits and diverse plant species. The paper further addresses the challenge of limited data by developing a methodology adept at managing scenarios with restricted training data. It explores diverse agricultural applications, including fruit quality control and identification of various plant species. The methodology proposed in the paper aims to mitigate economic losses and material wastage in the fruit industry by establishing a reliable mechanism for identifying and segregating fruit based on quality. Furthermore, the paper lays the groundwork for future applications and expansions, envisioning the model's potential for identifying plant species, monitoring their growth, and potentially detecting diseases by integrating additional relevant data. The extension of the model's functionality to video detection of plants and fruits is also proposed to broaden its application spectrum. Finally, the transformation of the model into a library is suggested, facilitating its incorporation into web applications and software to amplify knowledge dissemination and applicability across multiple sectors, including agriculture and artificial intelligence. The paper follows a systematic approach to achieve these objectives. This research utilized the TensorFlow and Keras libraries for model implementation. Additionally, the PIL, NumPy, and Matplotlib libraries were employed for testing and evaluation purposes.

Class Activation Maps (CAMs) are generated in the proposed model through a technique known as global average pooling. This process involves taking the average of feature maps obtained from the last convolutional layer of the network. By performing global average pooling, we obtain a class-specific activation map that highlights regions of the input image that are most relevant to the predicted class.

Regarding the gaps in model generalization and customization, the proposed model addresses several shortcomings compared to VGG-16, VGG-19, LeNet-5, and AlexNet. Firstly, the model incorporates additional layers and techniques beyond the standard architectures, allowing for improved performance in terms of accuracy and robustness. We have introduced specific modifications to enhance the model's ability to handle various image disturbances, such as covered, fuzzy, rain, and strong sunlight conditions. This addresses a significant gap in generalization, as the model demonstrates enhanced adaptability to real-world scenarios.

Furthermore, the proposed model tackles the limitation of limited data scenarios by incorporating techniques such as data augmentation and transfer learning. These approaches help mitigate the challenges of limited training data and improve the model's ability to generalize well to unseen instances. This is in contrast to the aforementioned models, which may face difficulties in achieving optimal performance when data is scarce. By explicitly addressing these gaps in model generalization and customization, the proposed model offers improved accuracy, robustness, and adaptability compared to VGG-16, VGG-19, LeNet-5, and AlexNet.

3.3 Dataset

The "Fruits Fresh and Rotten for Classification" dataset, comprising over 13,000 images across six different classes, was utilized. This dataset, subdivided into training, testing, and validation categories, was sourced from reputable websites like Kaggle. Figure 2 illustrates samples from six distinct classes of this database. The properties of this dataset are shown on Table 1 (Kalluri, 2018).

3.4 Preprocessing

The fruit images underwent a standardization process, resizing them to 224 by 224 pixels and normalizing their pixel values between 0 and 1. This step ensured consistent input dimensions and stable model training. To enhance the diversity of the training set, various data augmentation techniques were employed using the 'ImageDataGenerator' tool. These techniques included shear, zoom, rotations, translations, scaling, and horizontal flip. The parameters used for each transformation were carefully selected to introduce meaningful variations to the images (Azevedo, 2023).

Figure 3 showcases examples of training images after the application of 'ImageDataGenerator', demonstrating the effectiveness of these augmentations in creating variability within the dataset. These preprocessing steps are vital for enhancing the model's robustness and improving its ability to generalize to different scenarios, contributing to its overall performance.

3.5 Model structure

The deployed CNN architecture for fruit quality classification is meticulously designed to discern between fresh and rotten fruits. The model is organized in a sequential manner, starting with convolutional layers that capture intricate patterns in the input data. Subsequently, max-pooling layers are employed to reduce spatial dimensions and retain essential features. The model further incorporates fully connected layers to facilitate complex feature extraction and decision-making. Batch Normalization layers are strategically inserted to enhance the stability and convergence of the training process. Additionally, Dropout layers are incorporated to mitigate overfitting issues, promoting better generalization.

Figure 4 illustrates the sequential arrangement of these layers, providing a visual representation of the proposed model structure. This architecture is tailored to effectively capture and classify distinct features associated with fruit quality, contributing to the model's robust performance.

Figure 5 shows the proposed model structure. The model begins with convolutional layers, followed by Batch Normalization, max-pooling layers, Dropout layers, and finally fully connected layers.

As shown in Figure 6, the block diagram of the proposed algorithm for fruit decay detection and plant identification involves several key steps. First, a diverse dataset of fruit images, both healthy and spoiled, is collected and labeled. The images are then preprocessed by resizing them to a standard size and normalizing the pixel values. Data augmentation techniques are applied to enhance the training process. Next, a convolutional neural network (CNN) model with 11 layers is designed and implemented using TensorFlow. The model consists of multiple convolutional and pooling layers to extract features from the images, followed by fully connected layers for classification. Activation functions, batch normalization, and dropout layers are utilized to improve performance and prevent overfitting. The model is trained using the categorical-cross-entropy loss function and the Adam optimizer (Amigo, 2019). Early stopping is employed to prevent overfitting, and the best models are saved during training. In addition to evaluating the model's performance through accuracy and confusion matrix analysis, Class Activation Maps (CAMs) were utilized to gain insights into the model's decision-making process.

3.6 Objective function and optimization

The model employed the categorical-cross-entropy loss function, tailored for effective multi-class classification. For optimization, the Adam optimizer was chosen, providing adaptive learning rates and expedited convergence to local minima, thereby enhancing the training efficiency (C. Ltd, 2023). The categorical cross-entropy formula, represented by Equation 1, encapsulates the essence of the loss function, facilitating a robust mechanism for distinguishing between fresh and rotten fruits.

$$CE = -\sum_{i=1}^{C'=2} t_i \log\left(f(s_i)\right)$$
$$= -t_1 \log\left(f(s_1)\right) - (1 - t_1) \log\left(1 - f(s_1)\right) \quad (1)$$



FIGURE 2
Different class of this dataset.

TABLE 1 The properties of "Fruits Fresh and Rotten for Classification" dataset.

	Image	Train	Test
1	Fresh Apples	1693	395
2	Fresh Banana	1581	381
3	Fresh Oranges	1466	388
4	Rotten Apples	2342	601
5	Rotten Banana	2224	530
6	Rotten Oranges	1595	403

3.7 Early stopping and model selection

Fruit decay detection and plant identification are crucial aspects in agricultural and horticultural practices, playing a significant role in ensuring crop quality, disease control, and overall productivity (Pessarakli, 1994; Jayasena et al., 2015). Detecting fruit decay accurately and in a timely manner minimizes post-harvest losses, ensures food safety, and optimizes storage and distribution processes (Pessarakli, 1994). Additionally, early detection allows for prompt actions such as sorting and removal, preventing the spread of diseases and preserving the quality of the remaining fruits (Pessarakli, 1994).

Automated fruit decay detection systems based on computer vision and machine learning techniques have demonstrated promising results in terms of accuracy, speed, and cost-effectiveness (Jayasena et al., 2015; Boulent et al., 2019; Lakshmanan, 2019). These systems contribute to reducing post-harvest losses, optimizing storage conditions, and enhancing the overall efficiency of the fruit supply chain.

Plant identification is equally important and serves various purposes in agricultural practices. Accurate identification of plant species and cultivars aids in selecting appropriate varieties for specific environments, optimizing cultivation techniques, and improving agricultural practices (Barbedo, 2018; Wäldchen and Mader, 2018). Furthermore, plant identification plays a vital role in effective pest management by enabling timely and targeted application of control measures (Ferentinos, 2018). It also contributes to biodiversity conservation efforts by facilitating the monitoring and preservation of endangered plant species (Kaur and Kaur, 2019).

The early stopping technique was used to prevent overfitting and select the optimal models. This technique halts the training process when the error on the validation data increases, thereby preserving model accuracy. Furthermore, the model checkpoint was used to save the best models during training (C. Ltd, 2023; Gençay, 2023). Figure 7 depicts the utilization of Early Stopping and Model Checkpoint techniques in the training process of the proposed model. These techniques have been incorporated to enhance the model's training efficiency and performance. The Model Checkpoint, which saves the best-performing model during training.



FIGURE 3
An example of training images after using "imageDataGenerator".

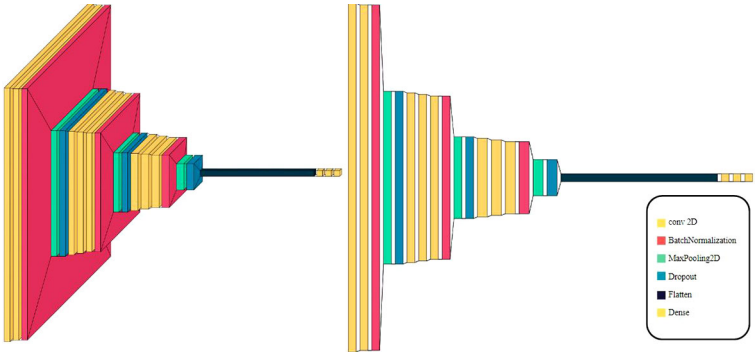


FIGURE 4
Visualization of the 11-layer the proposed model architecture.

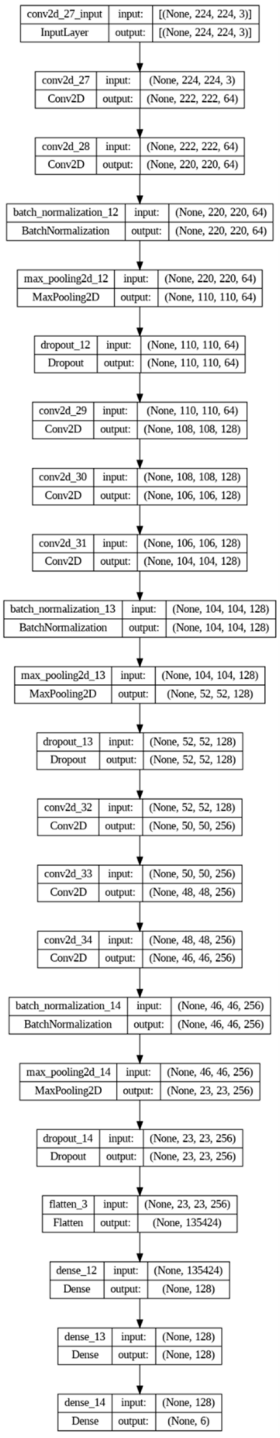


FIGURE 5
The proposed model structures.

4 Results

In this paper, we use an 11-layer convolutional neural network. We successfully built a high-accuracy artificial intelligence model for identifying and detecting three types of fruits (apple, orange, and banana) and their freshness or ripeness. The model achieved excellent performance. With careful evaluation and appropriate training, the provided artificial intelligence model works effectively in detecting the freshness or ripeness of fruits and identifying beneficial and harmful plants.

However, during the execution of this project, we encountered some challenges, including the lack of suitable datasets, model implementation and layer arrangement, limited powerful hardware resources, and the preparation of coding environments. Despite these challenges, by making efforts and optimizing the available resources, we developed a high-accuracy model. The proposed model's performance is compared with other existing CNN models, and its generalization capabilities and robustness are tested on separate data. The results, including accuracy comparisons and confusion matrix visualization, demonstrate the superiority of the proposed model.

The CNN model was trained using the training and validation data for 129 epochs. After this stage, the model ceased to show significant improvement, indicating that further training would not yield better results. In other words, the changes in weights were not meaningful, and further focus on training the model would not yield better results. Figure 8 depicts the variation of the proposed model's accuracy and loss values on the training data. Additionally, this figure showcases a visual representation of the model's performance.

To prevent overfitting, the early stopping callback is used. This callback evaluates the model's performance in each epoch and stops the training process earlier if no improvement is observed. This decision ensures that the model, considering the information learned in previous epochs, is selected and saved. This version is considered the final result of the training and can be utilized with high accuracy for detecting fresh and rotten fruits. Based on the usage of this callback, the training of the model was stopped at epoch 129. Additionally, the model's best performance was saved using model checkpoint, with the model from epoch 99 being considered the best performance. Figure 9 showcases the performance of the proposed method on a selection of samples from different classes.

As Figure 10 shows, this model underwent rigorous testing on datasets featuring challenging conditions, including intense shadows, extreme lighting variations, and partial image obstructions, yielding accurate results in most instances. Notably, under normal conditions, predictions were consistently accurate, showcasing the robust performance of the model. Only in cases with extreme shadows did the model occasionally exhibit errors. The comprehensive test results, encompassing predictions under both normal and challenging conditions, are depicted in Figure 10.

Figure 11 shows, VGG16 model accuracy and loss changes during training. As shown in Figure 12, the proposed model achieved a training accuracy of 99.8% and a validation accuracy of 99.7%. The model achieved a test accuracy of 99.93%, demonstrating the model's strong generalization capabilities. The libraries used for loading and processing images include Pandas, TensorFlow, Keras, PIL, and Matplotlib. A total of 2698 images were used to evaluate the model's performance, and excellent results were obtained. These results indicate that the model accurately detects fresh and rotten fruits with high precision and can be used as a powerful tool in fruit-related industries. Based on the comparison

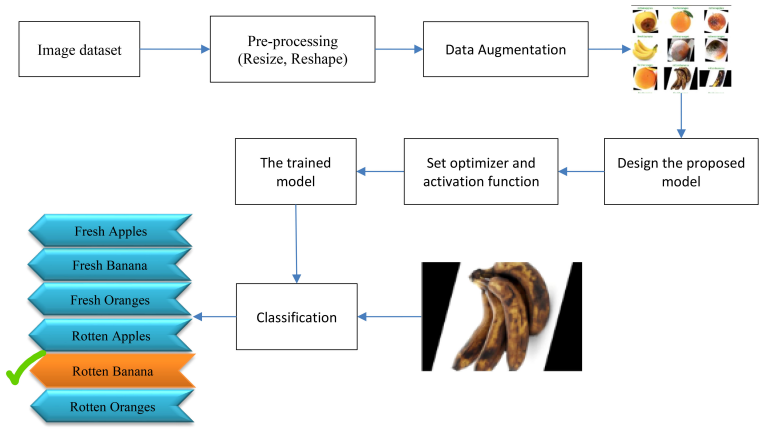


FIGURE 6
Block diagram of the proposed method.

depicted in Figures 11 and 12, it can be concluded that the proposed method outperforms VGG16. Table 2 compares the proposed method with various state-of-the-art methods. This table presents a comparison between recent deep learning methods and other state-of-the-art approaches, allowing us to conclude that the proposed method outperforms these methods. This conclusion is based on the significant design aspects incorporated into the proposed network. The proposed network outperforms the "Transfer Learning with VGG16" and "Transfer Learning with Res-Net" approach as well as other networks such as VGG-16, LeNet-5, CNN, and AlexNet.

4.1 Confusion matrix

The Confusion Matrix is a performance measurement for machine learning classification. It helps visualize the performance of an algorithm. As shown in Figure 13, the confusion matrix of the proposed model shows a high number of correct predictions, with only a few misclassifications, confirming the model's robustness.

4.2 Class activation maps

To visualize influential regions in the decision-making process of our convolutional neural network (CNN), we employed Class Activation Maps (CAMs) using the GradCAM technique. CAMs provide a visual representation of critical areas within images that significantly contribute to accurate classification. These maps offer insights into the model's attention by juxtaposing original images with CAMs, revealing impactful regions during classification.

By incorporating CAMs generated through GradCAM, we validate the alignment of the model's focus with relevant image features, enhancing transparency in decision-making post-training. This visualization reinforces the efficacy of the CNN in recognizing and highlighting critical aspects of the input data. The performance and effectiveness of the model can be observed in Figure 14, where a selection of CAMs for specific images is presented.

```
history = model.fit(
    train_generator,
    batch_size=32,
    epochs=499,
    validation_data=val_generator,
    callbacks=[early_stopping,model_checkpoint]
)
```

FIGURE 7
Early stopping and model checkpoint techniques in the proposed model training.

5 Discussion

The successful implementation of an 11-layer convolutional neural network (CNN) for fruit quality control signifies a significant advancement in utilizing deep learning for agricultural purposes. The remarkable accuracy achieved during training, validation, and testing phases (99.8%, 99.9%, and 99.93% respectively) surpasses established methods, establishing this novel approach as a leading method for precise fruit classification.

Beyond its impressive accuracy, this methodology demonstrates more than just classification proficiency. It showcases robustness and adaptability, crucial for real-world applications, especially in domains where misclassification can have significant consequences. While existing models like VGG-16, VGG-19, LeNet-5, and AlexNet have proven effective in various domains, their application in fruit quality control reveals potential gaps in model generalization and customization, which the proposed model effectively addresses.

The meticulously crafted 11-layer CNN, optimized for the unique challenges of fruit classification, is not just a classifier but a result of strategic decisions. These decisions include thoughtful architectural design for nuanced feature extraction, as well as the implementation of data augmentation and early stopping techniques to mitigate overfitting and enhance generalization.

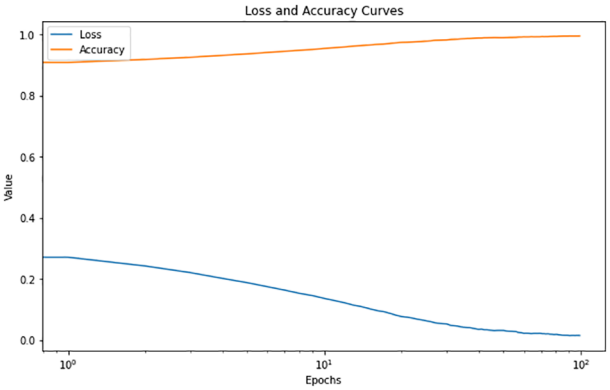


FIGURE 8
Model accuracy and loss changes during training.

This tailored approach enables the model to handle the diverse characteristics of different fruits, capturing intricate patterns and features that generic models may overlook.

The inclusion of Class Activation Maps (CAMs) not only enhances transparency but also facilitates continuous model refinement. By providing visual insights into decision-making processes, CAMs enable a deeper understanding and optimization of feature extraction and classification, leading to incremental improvements in model performance. The robustness of the proposed method is evident from the analysis of the confusion matrix, which reveals a high number of correct predictions with minimal misclassifications. This robustness reinforces the method's reliability and its potential to be deployed in real-world fruit quality control scenarios.

In this section of the article, it is essential to note that due to constraints in the dataset in this domain, enhancing the model's accuracy for various conditions, especially intense shadows, can be achieved with an increased dataset. Expanding the dataset for different scenarios, including intense shadow conditions, holds the potential to further improve the model's performance.

Moreover, the model demonstrates versatility beyond fruit classification, extending its capabilities to identify various plant species. This broadens its applicability in diverse agricultural scenarios. Additionally, the model proves effective even with



FIGURE 9
Model performance results after evaluation on test data.

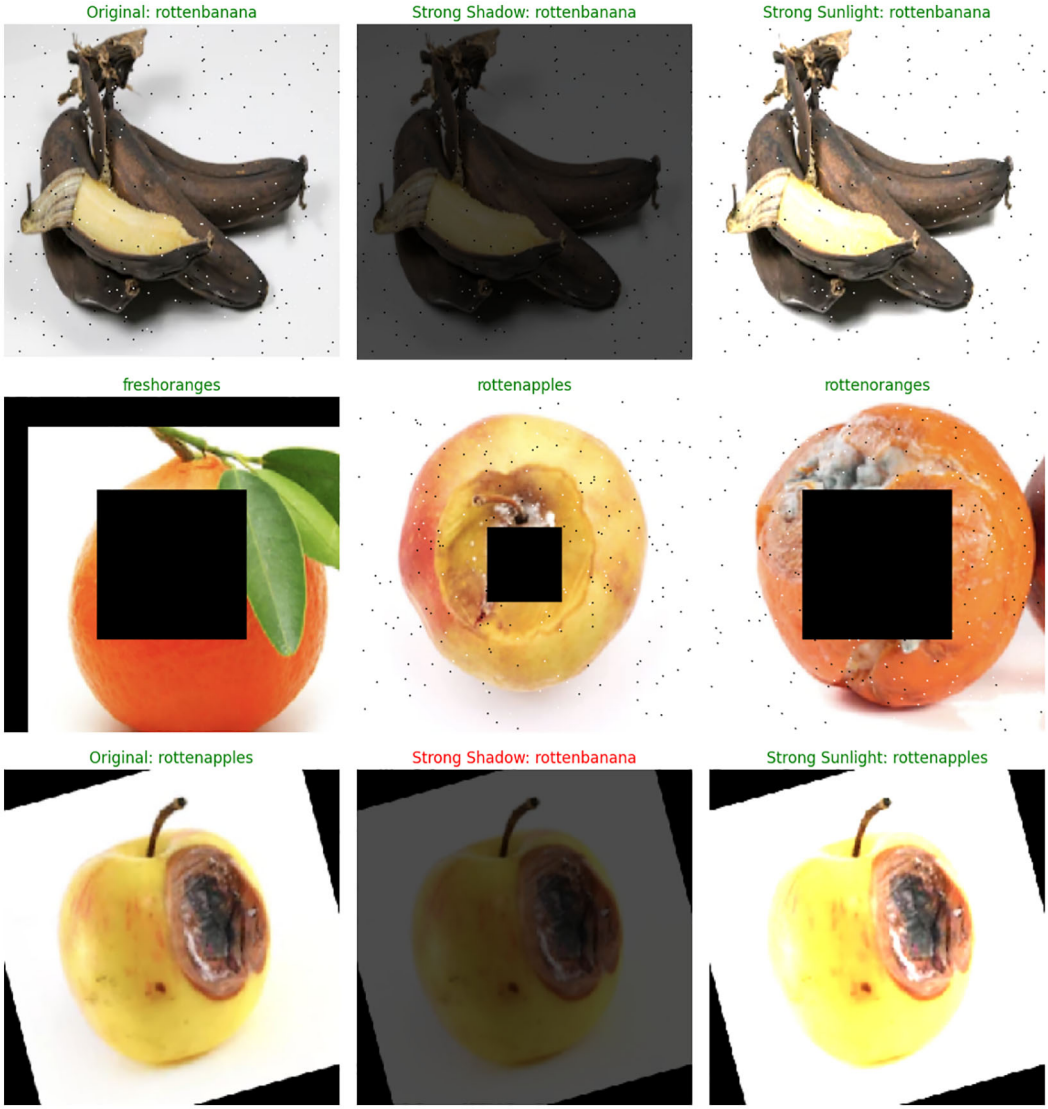


FIGURE 10
Model performance result for inputs with different conditions.

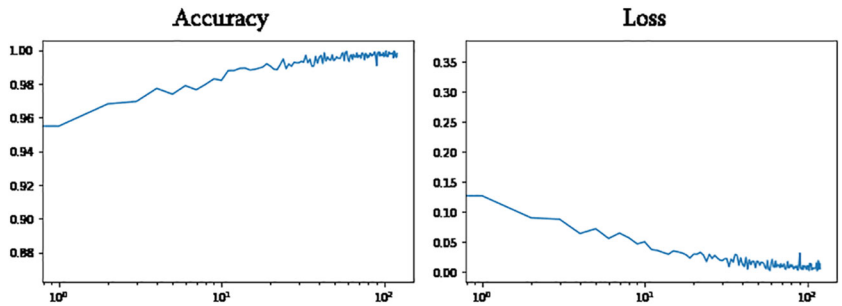


FIGURE 11
VGG16 model accuracy and loss changes during training.

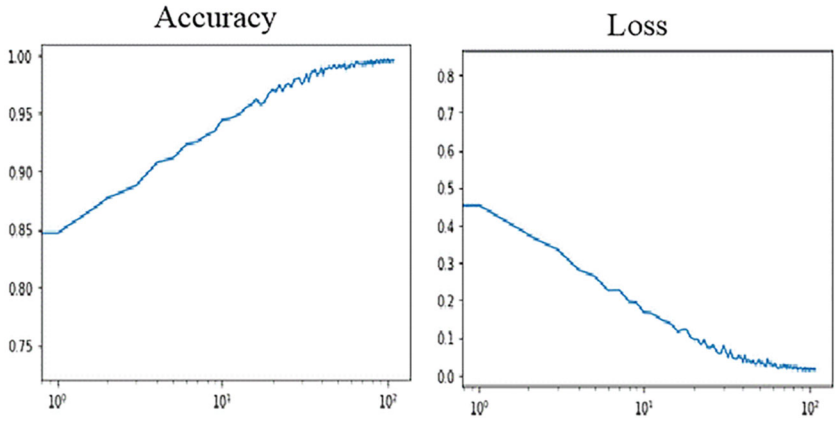


FIGURE 12
The proposed model accuracy and loss changes during training.

limited training data, making it a practical tool for deployment in different agricultural contexts.

The proposed model for automated agriculture systems in fruit protection introduces several significant advancements compared to previous approaches. These include robustness and generalization through handling variations in decay patterns and environmental conditions, effective handling of limited data scenarios using active learning, and data augmentation techniques, and a strong emphasis on interpretability and explainability through feature visualization. These advances enhance the practicality and performance of the proposed model, making it a novel and comprehensive solution for automated agriculture systems in fruit protection.

6 Conclusion

The proposed method has made significant strides in fruit quality control and other agricultural applications. Its custom model architecture, robustness-enhancing strategies, and versatility set it apart. While its exceptional accuracy sets a new benchmark, the holistic approach to design and application is what truly distinguishes it. It goes beyond being just a classifier, showcasing the integration of deep learning into specialized domains. The use of Class Activation Maps (CAMs) and a focus on transparency and model refinement are notable features. They improve decision-making and enable continuous model improvement through visual data and practical applications. This research has practical implications, particularly in enhancing fruit quality control and reducing economic waste. The model's effectiveness in distinguishing between fresh and rotten fruits, as well as its robust performance validated through confusion matrices and CAMs, demonstrates potential beyond its current application. It can be extended to create accurate models for detecting plant-related videos, identifying species, monitoring growth, and detecting diseases. This broadens its applicability from industry to research. Sharing knowledge through web applications and software libraries can be a valuable resource across various fields, including agriculture and artificial intelligence.

7 Future work

In terms of future work, there are several areas that can be explored to enhance the proposed method. Firstly, data expansion

TABLE 2 Comparison of the accuracies.

Model	Accuracy
VGG-16	99.1
VGG-19 (Pathak and Makwana, 2021)	76.48
LeNet-5 (Pathak and Makwana, 2021)	82.93
AlexNet (Pathak and Makwana, 2021)	83.56
CNN (Pathak and Makwana, 2021)	98.23
Proposed method	99.93
Transfer Learning with VGG16	99.44
Transfer Learning with Res-Net	99.59

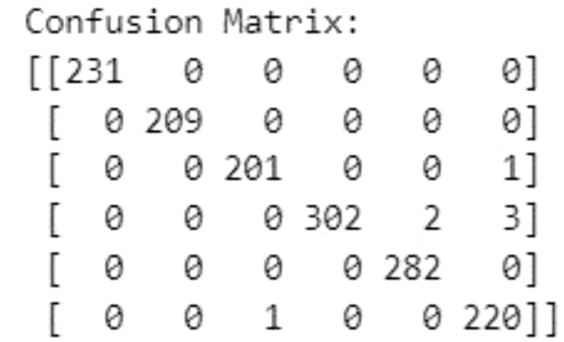


FIGURE 13
The confusion matrix.

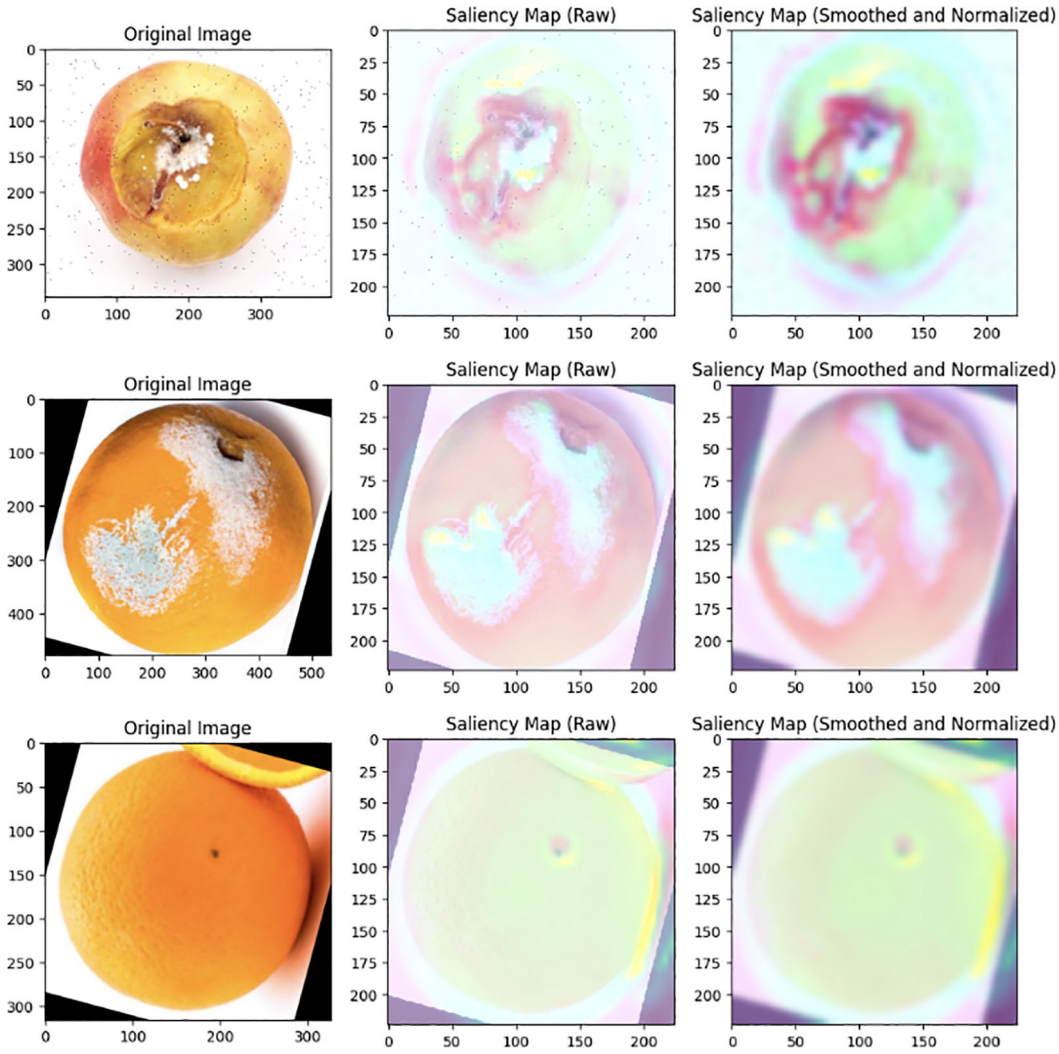


FIGURE 14
Insights from GradCAM illuminate key image features guiding CNN decisions.

through augmenting the dataset to encompass diverse conditions, including challenging scenarios like intense shadows, can significantly improve the model's real-world accuracy. This would involve collecting and incorporating more varied and representative data to ensure the model's robustness. Secondly, further refinement of the model's architecture is essential. Through iterative exploration and fine-tuning, the adaptability and performance of the model can be enhanced across different conditions. This may involve experimenting with different network architectures, optimizing hyperparameters, and incorporating advanced techniques such as attention mechanisms or transfer learning.

Additionally, continuous evaluation and benchmarking against contemporary models will be crucial to ensure that the proposed approach remains at the forefront of accuracy and efficiency. Regularly assessing its performance and comparing it with state-of-the-art methods will help identify areas for improvement and guide future research directions. Furthermore, we can provide accessible resources for practical implementation in fields like agriculture and artificial intelligence. This would involve creating intuitive interfaces that allow users to apply the model easily and obtain valuable insights from the fruit decay detection and plant identification system. Overall, these future directions, including data expansion, model architecture refinement, technique exploration, and continuous evaluation, can contribute to advancing the proposed method and its potential impact in various industries and research fields.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification. Three original packages have been used and/or developed in the framework of this study. All of these are publicly available on official repository and/or main DevOps platforms: https://github.com/pariyaaf/FruitDisease Detection-pariya.

Author contributions

PA: Data curation, Formal analysis, Methodology, Software, Writing – original draft. TZ: Software, Supervision, Visualization, Writing – review & editing. MD: Project administration, Resources, Validation, Visualization, Writing – review & editing. MZ: Methodology, Software, Validation, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

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/fpls.2024.1366395/full#supplementary-material

References

Alex, K., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 25, 84–90. doi: 10.1145/3065386

Amigo, H. (2019) *Cross entropy*. Available online at: https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=howdy_amigo&logNo=221442864397.

Azevedo, N. (2023) *Data Preprocessing Techniques: 6 Steps to Clean Data in Machine Learning*. Available online at: https://www.scalablepath.com/data-science/data-preprocessing-phase.

Barbedo, J. G. (2018). Factors influencing the accuracy of plant disease recognition models in real-life scenarios. *Plant Dis.* 102, 2394–2401. doi: 10.1016/j.biosystemseng.2018.05.013

Boulent, J., Fuentes, A., and Valente, J. (2019). Computer vision for fruit detection and localization: A review. *Food Bioprocess Technol.* 12, 153–167. doi: 10.1007/s11947-023-03005-4

Brital, A. (2021) *Inception V2 CNN Architecture Explained*. Available online at: https://medium.com/AnasBrital98/inception-v2-cnn-architecture-explained-128464f742ce.

Brownlee, J. (2018a). *Machine Learning Algorithms From Scratch* (Victoria: Machine Learning Mastery).

Brownlee, J. (2018b). *Better Deep Learning* (Victoria: Machine Learning Mastery).

C. Ltd (2023). *Mastering AI model training* (Cybellium Ltd).

Dertat, A. (2017) *Applied Deep Learning - Part 4: Convolutional Neural Networks*. Available online at: https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2.

Dhaka, V. S., Meena, S. V., Rani, G., Sinwar, D., Ijaz, M. F., and Woźniak, M. (2021). A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. *Sensors* 21, 4749. doi: 10.3390/s21144749

Editorial, K. (2022) *Pooling layers in a convolutional neural network*. Available online at: https://keepcoding.io/blog/capas-pooling-red-neuronal-convolucional/ (Accessed 14 10 2023).

Fan, S., Li, J., Zhang, Y., Tian, X., Wang, Q., He, X., et al. (2020). On line detection of defective apples using computer vision system combined with deep learning methods. *J. Food Eng.* 286, 110102. doi: 10.1016/j.jfoodeng.2020.110102

Faouzi, B. (2021). "FruitDelect," in *GitHub*. Available at: https://github.com/fbraza/FruitDetect.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* 145, 311–3185. doi: 10.1016/j.compag.2018.01.009

Gençay, R. (2023) *Early stopping based on cross-validation*. Available online at: https://www.researchgate.net/figure/Early-stopping-based-on-cross-validation_fig1_3302948.

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*, Vol. pp. 580–587.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning* (MIT press).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* 27, 2672–2680. doi: 10.1007/978-3-658-40442-0_9

Gupta, C. (2021). Modern Machine and Deep Learning Systems as a way to achieve Man-Computer Symbiosis. *arXiv e-prints*. arXiv-2101. doi: 10.3390/s21165386

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

JananiSBabu (2020) *ResNet50_From_Scratch_Tensorflow*. Available online at: https://github.com/JananiSBabu/ResNet50_From_Scratch_Tensorflow.

Jayasena, D. D., Boyles, S., and Dykes, G. A. (2015). Rapid detection of fruit spoilage using a novel colorimetric gas sensor array. *Sensors Actuators B: Chem.* 216, 515–521.

Kalluri, S. R. (2018) *Fruits fresh and rotten for classification*. Available online at: https://www.kaggle.com/datasets/sriramr/fruits-fresh-and-rotten-for-classification/code.

Kalra, K. (2023) *Convolutional Neural Networks for Image Classification*. Available online at: https://medium.com/khwabkalra1/convolutional-neural-networks-for-image-classification-f0754f7b94aa.

Kanna, G. P., Kumar, S. J. K. J., Kumar, Y., Changela, A., Woźniak, M., Shafi, J., et al. (2023). Advanced deep learning techniques for early disease prediction in cauliflower plants. *Sci. Rep.* 13, 18475. doi: 10.1038/s41598-023-45403-w

Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). "Progressive growing of GANs for improved quality, stability, and variation," in *International Conference on Learning Representations*.

Kaur, S., and Kaur, P. (2019). Plant species identification based on plant leaf using computer vision and machine learning techniques. *J. Multimedia Inf. System* 6, 49–60. doi: 10.33851/JMIS.2019.6.2.49

Kundu, N., Rani, G., Dhaka, V. S., Gupta, K., Nayak, S. C., Verma, S., et al. (2021). IoT and interpretable machine learning based framework for disease prediction in pearl millet. *Sensors* 21, 5386. doi: 10.3390/s21165386

Lakshmanan, L. (2019) *ML Design Pattern #2: Checkpoints*. Available online at: https://towardsdatascience.com/ml-design-pattern-2-checkpoints-e6c254c5fe.

Narayanam, K. L., Krishnan, R. S., Robinson, Y. H., Julie, E. G., Vimal, S., Saravanan, V., et al (2022). Banana plant disease classification using hybrid convolutional neural network. *Computational Intelligence and Neuroscience*, 1–13. doi: 10.1155/2022/9153699

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, 2278–2324. doi: 10.1109/5.726791

LeCun, Y., Kavukcuoglu, K., and Farabet, C. (2010). "Convolutional networks and applications in vision," in *Proceedings of 2010 IEEE international symposium on circuits and systems*. (IEEE), 253–256.

Lewis, J. (2022) *How Does Food Waste Affect the Environment*? Available online at: https://earth.org/how-does-food-waste-affect-the-environment/.

Li, S., Luo, H., Hu, M., Zhang, M., Feng, J., Liu, Y., et al. (2019). Optical non-destructive techniques for small berry fruits: A review. Artificial Intelligence in Agriculture, 2, 85–98

Long, J., Shelhamer, E., and Darrell, T. (2015). "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Vol. pp. 3431–3440.

Norman, C. (2019). AI in Pursuit of Happiness, Finding Only Sadness: Multi-Modal Facial Emotion Recognition Challenge. *arXiv preprint*. arXiv:1911.05187.

Pathak, R., and Makwana, H. (2021). Classification of fruits using convolutional neural network and transfer learning models. *J. Manage. Inf. Decision Sci.* 24, 1–12.

Pessarakli, M. (1994). Respon of green beans (Phaseolus vulgaris L.) to salt stress in handbook of plant and crop physiology. doi: 10.1201/b10329-48

Ramya, M. (2023). Identification of skin disease using machine. *Int. J. Creative Res. Thoughts (IJCRT)*.

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: 10.1109/tpami.2016.2577031

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).

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., and McCool, C. (2016). Deepfruits: A fruit detection system using deep neural networks. *Sensors*. 16 (8), 1222. doi: 10.3390/s16081222

Scarlat, A. (2018). "Melanoma - resNet50 fine tune," in *Kaggle*. Available at: https://www.kaggle.com/code/drscarlat/melanoma-resnet50-fine-tune/notebook.

Selvaraj, M. G., Vergara, A., Ruiz, H., Safari, N., Elayabalan, S., and Ocimati, W. (2019). AI-powered banana diseases and pest detection. *Plant Methods* 15, 1–11. doi: 10.1186/s13007-019-0475-z

Shahid, M. (2019). "Learn Convolutional Neural Network from basic and its implementation in Keras," in *Towards Data Science*. Available at: https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529.

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition.

Sonwani, E., Bansal, U., Alroobaea, R., Baqasah, A. M., and Hedabou, M. (2022). An artificial intelligence approach toward food spoilage detection and analysis. *Front. Public Health* 9, p.816226. doi: 10.3389/fpubh.2021.816226

Tan, M., and Le, Q. (2019). "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. (PMLR), 6105–6114.

Valliappa Lakshmanan, M. G. R. G. (2021). *Practical Machine Learning for Computer Vision* (California: O'Reilly Media).

Wäldchen, J., and Mader, P. (2018). Plant species identification using computer vision techniques: A systematic literature review. *Arch. Comput. Methods Eng.* 25, 507–543. doi: 10.1007/s11831-016-9206-z

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? *Adv. Neural Inf. Process. Syst.* 27.

Zeiler, M. D., and Fergus, R. (2014). "Visualizing and understanding convolutional networks," in *Computer Vision–ECCV 2014: 13th European Conference*, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13. (Springer International Publishing), 818–833.

Zhang, C., Liu, X., Chen, B., Yin, P., Li, J., Li, Y, et al (2020), June. Insulator profile detection of transmission line based on traditional edge detection algorithm. In IEEE International Conference on Artificial Intelligence and Computer Applications, 267–269

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 15 May 2024
DOI 10.3389/fpls.2024.1377269

Check for updates

OPEN ACCESS

EDITED BY
Roger Deal,
Emory University, United States

REVIEWED BY
Pengju Si,
Henan University of Science and Technology, China
Zari Farhadi,
University of Tabriz, Iran

*CORRESPONDENCE
Huiping Guo
imguohp@nwafu.edu.cn;
imghp@163.com

RECEIVED 05 February 2024
ACCEPTED 25 April 2024
PUBLISHED 15 May 2024

CITATION
Guo H, Li Y, Wang H, Wang T, Rong L, Wang H, Wang Z, Wang C, Zhang J, Huo Y and Guo S (2024) Research on autonomous navigation system of greenhouse electric crawler tractor based on LiDAR. *Front. Plant Sci.* 15:1377269. doi: 10.3389/fpls.2024.1377269

COPYRIGHT
© 2024 Guo, Li, Wang, Wang, Rong, Wang, Wang, Wang, Zhang, Huo and Guo. 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.

Research on autonomous navigation system of greenhouse electric crawler tractor based on LiDAR

Huiping Guo[1,2]*, Yi Li[1,2], Hao Wang[1,2], Tingwei Wang[1,2], Linrui Rong[1,2], Haoyu Wang[1,2], Zihao Wang[1,2], Chensi Wang[1,2], Jiao Zhang[1,3], Yaobin Huo[4] and Shaomeng Guo[4]

[1]College of Mechanical and Electronic Engineering, Northwest Agriculture and Forestry University, Yangling, Shaanxi, China, [2]Northern Agricultural Equipment Scientific Observation and Experimental Station, Ministry of Agriculture and Rural Affairs, Yangling, Shaanxi, China, [3]Project Control Department, BYD Automobile Company Limited, Xi'an, China, [4]Xi 'an Dongyang Machinery Company Limited, Xi'an, China

The application of autonomous navigation technology of electric crawler tractors is an important link in the development of intelligent greenhouses. Aiming at the characteristics of enclosed and narrow space and uneven ground potholes in greenhouse planting, to improve the intelligence level of greenhouse electric crawler tractors, this paper develops a navigation system of electric crawler tractors for the greenhouse planting environment based on LiDAR technology. The navigation hardware system consists of five modules: the information perception module, the control module, the communication module, the motion module, and the power module. The software system is composed of three layers: the application layer, the data processing layer, and the execution layer. The developed navigation system uses LiDAR, Inertial Measurement Unit (IMU) and wheel speed sensor to sense the greenhouse environment and the crawler tractor's information, employs the Gmapping algorithm to build the greenhouse environment map, and utilizes the adaptive Monte Carlo positioning algorithm for positioning. The simulation test of different global path planning algorithms in Matlab shows that the A* algorithm obtains the optimal overall global path. In the scene of map 5, the path planned by the A* algorithm is the most significant, and the number of inflection points is reduced by 40.00% and 87.50%, respectively; meanwhile, the path length is the same as that of the Dijkstra algorithm, but the runtime is reduced by 68.87% and 81.49%, respectively; compared with the RRT algorithm, the path length is reduced by 7.27%. Therefore, the A* algorithm and the Dynamic Window Approach (DWA) method are used for tractor navigation and obstacle avoidance, which ensures global path optimality while also achieving effective local path planning for obstacle avoidance. The test results suggest that the maximum lateral deviation of the built map is 6 cm, and the maximum longitudinal deviation is 16 cm, which meets the requirement of map accuracy. Additionally, the results of the navigation accuracy test indicate that the maximum lateral deviation of navigation is less than 13 cm, the average lateral deviation is less than 7 cm,

and the standard lateral deviation is less than 8 cm. The maximum heading deviation is less than 14°, the average heading deviation is less than 7°, and the standard deviation is less than 8°. These results show that the developed navigation system meets the navigation accuracy requirements of electric crawler tractors in the greenhouse environment.

KEYWORDS

electric crawler tractor, greenhouse, lidar, autonomous navigation, path planning

1 Introduction

By the end of 2021, China boasted over 28 million facility greenhouses, covering an area surpassing 40 million mu, making it the largest in terms of land area in the world (Hu et al., 2024). The mechanization of greenhouse cultivation has also become a hot research topic. Due to the enclosed and narrow space in greenhouses, traditional tractors, which cause severe pollution, are being replaced, thus research on electric crawler tractor has become one of the important directions in the study of greenhouse machinery. They serve as traction machinery to complete tasks such as plowing, rotary tillage, seeding, and harvesting within the greenhouse.

Given the enclosed environment of greenhouses, which are characterized by high temperatures and humidity, and the harsh working conditions, the automation of electric crawler tractor is particularly important due to the increased intensity of manual labor (Liu et al., 2022). The key to achieving automation in electric crawler tractor is autonomous navigation technology (Mahmud et al., 2019). Autonomous navigation mainly includes mapping, positioning, and path planning, and the accurate perception of environmental information is critical for precise navigation. Currently, the sensors used for environmental perception mainly include Global Navigation Satellite System (GNSS), vision, LiDAR, attitude sensor, and other sensors (İrsel and Altinbalik, 2018; Jiang et al., 2023; Jia et al., 2015; Arad et al., 2020; Zhang et al., 2020).

Autonomous navigation technology in industrial environments is relatively mature, often employing multi-sensor fusion for navigation, which provides high accuracy but involves higher costs, making it not entirely suitable for the agricultural production environment (Fu et al., 2023a, Fu et al., 2023b). Currently, in agricultural environments, GNSS technology is mainly used for autonomous navigation in fields or orchards (Wang et al., 2014). Zhang et al. (2013) developed a GNSS-based autonomous navigation system for combine harvesters, used for harvesting wheat and rice. Compared to field operations, orchards present challenges like canopy cover, which can affect GNSS signals, making it difficult to ensure high-precision work. Many scholars have introduced IMU, wheel speed sensor, vision, and other sensors on top of GNSS positioning to improve the positioning accuracy of equipment. Liu et al. (2018) designed a navigation operation system for a high clearance sprayer based on GNSS and inertial sensors, meeting the spraying operation requirements in three environments: cement pavements, dry fields, and paddy fields, with good stability and control precision. Kurashiki et al. (2010) used GPS and laser range finder to achieve autonomous positioning of autonomous vehicles in orchards, and the experimental results indicated that the average error in the lateral direction was about 20 cm, meeting the accuracy requirements for orchard navigation. Though the fusion of GNSS with other sensors can solve the problem of navigation accuracy under branch coverage to a good extent, in completely enclosed greenhouse environments, GNSS signals are significantly obstructed, making it difficult to meet the requirements for precise positioning.

Vision sensors can capture a wealth of information in enclosed environments and have been widely used in agricultural navigation applications in recent years (Tan et al., 2020). Wang et al. (2013) utilized monocular vision for tractor navigation, significantly reducing the operator's labor intensity and achieving autonomous navigation of agricultural tractors. Radcliffe et al. (2018) designed a navigation platform based on vision and ultrasonic sensors, using a multi-spectral camera to capture images of orchards and integrating them with background information of the canopy and sky. Machine vision was employed to extract the road in the center of tree rows, using ultrasound to measure distances. Although visual sensors can solve navigation problems in enclosed environments to a certain extent, they are susceptible to factors such as lighting and shadows in the greenhouse environment, making it difficult to achieve stable operation around the clock.

Laser sensors are widely used in enclosed environments such as indoors, offering many advantages including high ranging accuracy, good resolution, strong anti-interference capability, small size, and light weight. Many scholars have used LiDAR to solve the environmental perception and positioning issues of material handling and inspection robots, providing support for autonomous navigation (Jia et al., 2015; Jiang et al., 2022). Single LiDAR can enable navigation on smooth road surfaces, but motion-induced point cloud distortion in environmental perception can lead to inaccuracies in positioning and mapping, causing information biases. LiDAR integration with pose sensors can

enhance the precision of positioning and mapping of LiDAR-based navigation to some extent by rectifying these distortions. However, on navigation of tractors driving on uneven terrains, LiDAR solely integrating with IMU struggles to address the accumulation of IMU measurement biases. Moreover, LiDAR solely integrating with wheel speed sensor fails to adequately address the significant errors in pose estimation caused by signal drift. These indicate that fusing LiDAR with a single type of pose sensor is insufficient to resolve the substantial positioning and mapping errors induced by uneven surfaces. Therefore, the integration of LiDAR with multiple pose sensors is crucial for achieving precise positioning and mapping on uneven terrains.

Although a single LiDAR can solve indoor navigation problems to a certain extent, in greenhouse environments, the ground surfaces on which tractors work are rough and uneven dirt roads. In the process of environmental perception, a single LiDAR encounters point cloud distortion and is affected by the unevenness of the ground, leading to increased cumulative errors in positioning and mapping, deviations in environmental perception, and affecting the accuracy and completeness of mapping. The fusion of LiDAR and attitude sensors can obtain feature information of the greenhouse environment, information about obstacles around the tractor, and the state information of the tractor itself, enabling more accurate and reliable autonomous navigation.

In response to these challenges, this study, based on a remotely operated electric crawler tractor for greenhouses developed by our team, has designed an autonomous navigation system suitable for operating on uneven terrains of greenhouses. This system acquires the environmental data from LiDAR, and the pose information from IMU and wheel speed sensor of the tractor. The system employs an Extended Kalman Filter algorithm to fuse the data from IMU and wheel speed sensor, correcting motion-induced distortions in LiDAR point clouds, thereby improving the accuracy of positioning, mapping, and navigation on uneven surfaces of greenhouses. The overall structure of the research is shown in Figure 1. The main contributions are summarized as follows:

(1) Tailored to the characteristics of greenhouse operation environments, an autonomous navigation system for the tractor was built on the developed remote-controlled greenhouse electric crawler tractor, utilizing LiDAR, IMU, and wheel speed sensor, encompassing the design of both hardware and software systems;

(2) It proposes a mapping method for the greenhouse environment using the Gmapping algorithm, employs the Adaptive Monte Carlo Localization algorithm for positioning, and uses the A* algorithm for global path planning and DWA for obstacle avoidance, validated through simulations and comparison with operational needs;

(3) At the software layer of the electric tractor, the navigation system development is based on the Robot Operating System (ROS), with the mapping performance and navigation accuracy of the navigation system validated within a greenhouse.

The remainder of this paper is organized as follows: The second section discusses the design of the electric crawler tractor navigation

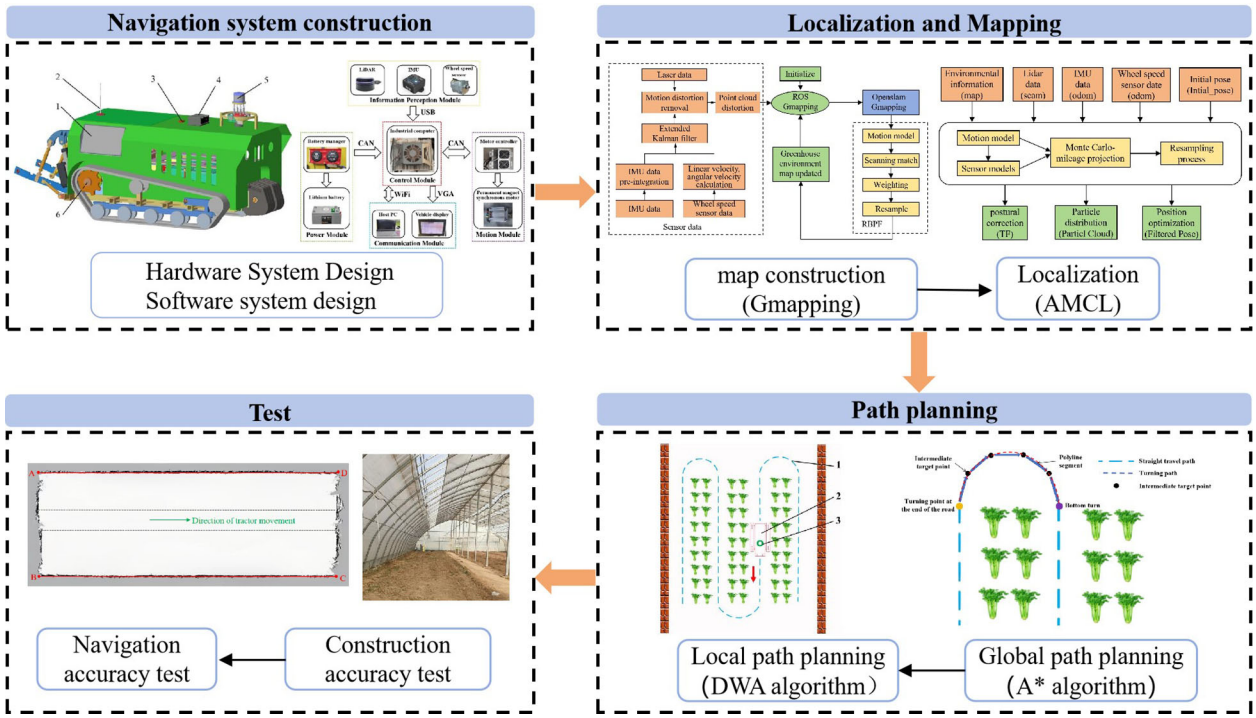


FIGURE 1
Overall structure diagram.

system, including mapping, positioning, and path planning of the navigation system; the third section addresses the testing experiments for mapping, positioning, and navigation accuracy of the designed navigation system, along with an analysis of the test results; finally, the conclusion.

2 Related works

LiDAR is not affected by lighting conditions, possesses strong anti-interference capability, and can accurately measure distances to surrounding objects. It has been widely applied in autonomous navigation for devices in various enclosed spaces (Hiremath et al., 2014). Li et al. (2023a) used LiDAR to construct maps of unknown environments to achieve autonomous path planning and navigation for mobile robots indoors. Li et al. (2023b) utilized LiDAR to acquire information on complex, intelligent factory environments to enhance the precise positioning and navigation performance of AGV (Automated Guided Vehicles) in an intelligent factory setting. While a single LiDAR can achieve indoor navigation, due to point cloud distortion during the autonomous navigation process, cumulative errors can occur over time, affecting navigation precision and failing to obtain accurate positioning.

Many scholars have attempted to solve navigation precision issues by integrating different sensors with LiDAR. Hou et al. (2020) developed a navigation system for transport robots based on dual LiDAR, matching the data from encoders with LiDAR point cloud data to obtain the environmental map and the robot's pose information, thereby increasing the robot's scanning range and map-building efficiency. Hu et al. (2023) adopted LiDAR and laser receivers to design a robot positioning system based on laser sensing, by acquiring the point cloud on the robot's laser receiver through the laser emitted from the LiDAR during scanning. Simultaneously, the laser receiver senses the scanning laser, integrating the time difference of sensing scanning laser and the laser receiver's point cloud features to localize the agricultural robot. Dang et al. (2021) leveraged the complementary characteristics of millimeter-wave radar and LiDAR, proposing an efficient and precise method of detecting, recognizing, and eliminating moving objects through sensor fusion and data association to enhance the performance of positioning and mapping. The above research utilizes external sensors to perceive environmental information for positioning, thus achieving navigation solutions, but their costs are relatively high.

Pose sensors such as IMU and wheel speed sensor are relatively cost-effective, not easily disturbed by external environments, and can provide information on the robot's own pose. Scholars have proposed the fusion of LiDAR with IMU and wheel speed sensor, which can effectively compensate for the limitations of using LiDAR alone, enhancing the robustness and accuracy of positioning and mapping. The IMU measures the acceleration of the robot's motion via an accelerometer and measures the angular velocity of motion via a gyroscope, calculating the current attitude through an attitude fusion algorithm (Pu et al., 2023). Si et al. (2023) aimed at the issues of significant positioning and mapping errors caused by LiDAR point cloud distortion, designed a point cloud distortion correction method based on continuous-time trajectories of IMUs, established a data fusion model between LiDAR and IMUs, and proposed a positioning and mapping method for anti-collision drilling robots based on tightly coupled IMU-LiDAR, effectively improving the precision and performance of positioning and map building. Shen et al. (2023) addressed the problems of easy loss of GNSS signals and poor robustness of traditional SLAM algorithms in orchard environments by proposing a tightly coupled LiDAR/IMU framework. This framework optimizes IMU and LiDAR separately through a factor graph, enabling the IMU to output high-frequency pose information and integrating the LiDAR to construct accurate point cloud maps. Ye et al. (2019) introduced a tightly coupled LiDAR-IMU fusion method, which proposed a refined algorithm with rotation constraints to further align the LiDAR pose with the global map. All the above research utilizes the attitude information provided by IMU to offer a prior estimation for point cloud matching and distortion elimination.

Wheel speed sensor utilize dead reckoning to estimate the changes in a robot's pose over time, assisting in robot positioning and capable of providing high-precision positioning accuracy in the short term (Wu et al., 2017). Ji et al. (2018) integrated LiDAR and wheel speed sensor information, using the Gmapping algorithm to establish an environmental map, thereby achieving positioning and navigation for inspection robots. Lu et al. (2024) received data from LiDAR and wheel speed sensor during the positioning and mapping process to determine the pose of an indoor disinfection robot and to create a two-dimensional grid map of the environment it occupies. Wang and An (2024) designed an autonomous navigation system for indoor medical goods transport and epidemic prevention assistant robots, utilizing the fusion of LiDAR and wheel speed sensor to complete positioning and mapping. High-precision wheel speed sensor can improve the point cloud distortion of LiDAR and enhance the precision of positioning and mapping in navigation.

Although the fusion of LiDAR with a single attitude transducer has improved the accuracy of localization and mapping to a certain extent, for the closed and narrow greenhouse working environments, and the uneven soil roads where electric crawler tractor operate, the precision and robustness of autonomous navigation are relatively weak. By adopting a method that combines LiDAR sensor with position-posture sensor, it is possible to obtain information about the characteristics of the greenhouse environment, information about obstacles surrounding the tractor, as well as the tractor's own state information, improving the navigation accuracy and reliability of the electric tractor's operation. Therefore, this study designs an autonomous navigation system for greenhouse electric track tractors based on LiDAR, integrating IMU and wheel speed sensor through an Extended Kalman Filter fusion, using the fused data to eliminate distortions in the LiDAR point cloud, thus enhancing the precision of localization and navigation.

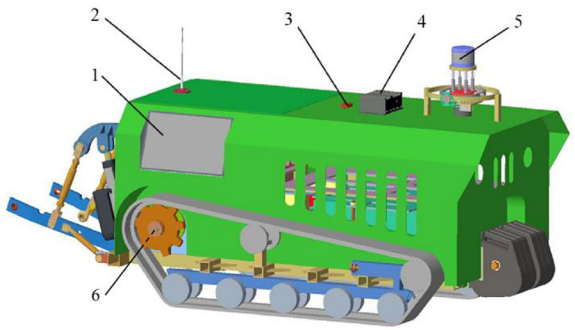


FIGURE 2
3D schematic diagram. 1. Vehicle display; 2. Antenna; 3. IMU; 4. Industrial computer; 5. LiDAR; 6. Wheel speed sensor.

3 Materials and methods

3.1 Design of the electric crawler tractor navigation system

3.1.1 Agronomic parameter

The vegetables grown in the greenhouse can be categorized into leaf vegetables and fruit vegetables. The measurement of vegetable planting agronomic parameters is the key to the operation of electric crawler tractors. Generally, the height of vegetables in the harvest period falls within 50 cm, and the width of the greenhouse ranges between 6 to 16 m, which provides a reference for selecting the navigation hardware and determining the height of the sensor installation. According to the agronomic requirements, the autonomous navigation system of the electric crawler tractor is studied.

3.1.2 Overall system composition

In this paper, the navigation system is designed based on the remote-controlled electric crawler tractor, and the hardware of the navigation system is built, including the vehicle display, antenna, industrial computer, LiDAR,IMU, and wheel speed sensor. The navigation system enables the monitoring, environmental mapping, and releasing of navigation target point instructions through a remote monitoring platform on the PC and the vehicle display. The tractor provides real-time location information and working status feedback to both the PC remote monitoring platform and the vehicle display. The laser radar is installed on the fixed device at the front of the tractor; it obtains the environmental information of the greenhouse and feeds it back to the industrial computer to complete environmental information mapping. IMU is installed in the central position above the tractor; it measures the posture and acceleration of the tractor movement, assists the LiDAR in positioning, and sends the positioning information to the industrial computer, which sends the speed control information to the motor control system through CAN communication. Meanwhile, the motor controller feedbacks the real-time speed information of the chassis to the industrial computer through CAN communication. The tractor autonomous navigation task is

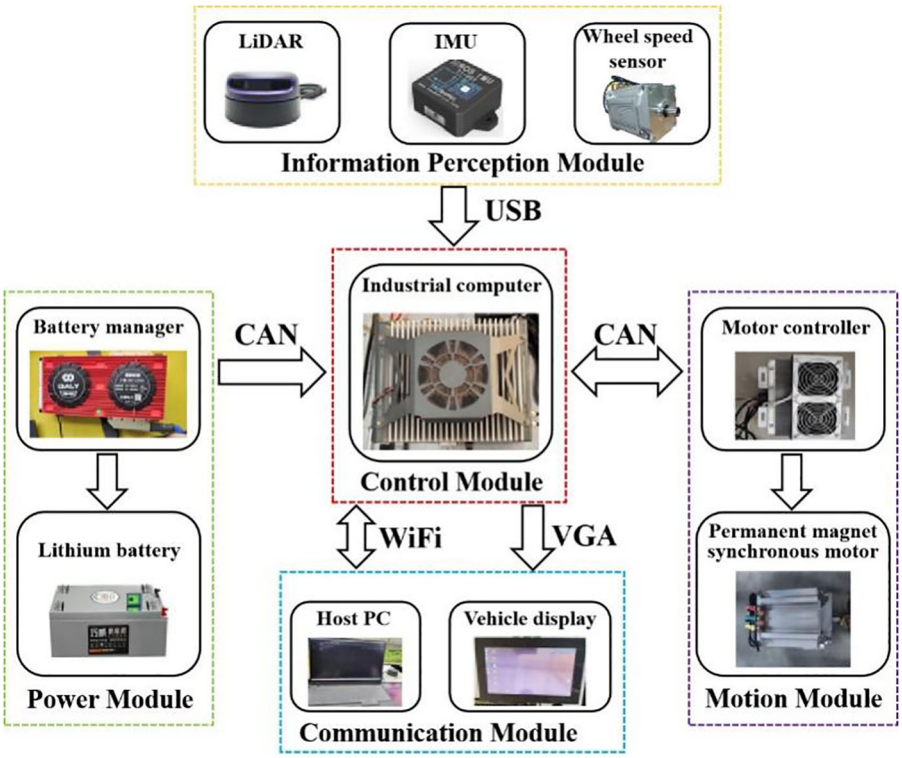


FIGURE 3
Overall hardware structure diagram.

completed by these components together, and the overall structure of the navigation system is illustrated in Figure 2.

3.1.3 Hardware system design

The navigation hardware system of the electric crawler tractor mainly consists of five modules: the information perception module, the control module, the communication module, the motion module, and the power module. The structure of the hardware system is shown in Figure 3.

3.1.3.1 Information perception module

The information perception module mainly includes LiDAR, IMU and wheel speed sensor. By using the RPLIDAR A2 2D LiDAR of Silan Technology and laser triangulation ranging technology, the laser triangulation measurement technology can perform up to 8,000 ranging actions per second, and the measuring distance can reach 25 m, which can meet the needs of mapping and navigation in the greenhouse environment. The IMU adopts a HIFI 6-axis inertial sensor produced by TAObotics company, whose return frequency can reach 300 Hz. It can obtain real-time and accurate attitude information of electric track tractors and coordinate with LiDAR for positioning in the greenhouse environment.

3.1.3.2 Control module

The tractor takes an industrial computer as the top-level controller. The computer is equipped with an i3-9100T processor, 8 GB memory, and 128 GB hard disk, and it runs Ubuntu 16.04 and ROS Kinetic operating system. LiDAR and IMU are connected through the USB interface to achieve real-time communication, and real-time communication with two motor controllers and the battery management system is via CAN bus.

3.1.3.3 Communication module

The industrial computer is connected to the remote monitoring platform of the host PC through a WiFi network, and it is connected to the vehicle display through VGA. The host PC is Lenovo Rescuer R9000P equipped with an R7-5800H processor and 16 GB memory. The vehicle display uses a 15-inch capacitive display to provide various interfaces. The remote monitoring platform on the PC and the vehicle display on the tractor can realize the monitoring of the tractor's working status, environmental mapping, and the issuance of navigation target point instructions; the tractor, as the executor of the command, feedback real-time position information and working status to the host PC remote monitoring platform through the WiFi network and to the vehicle display through VGA. The industrial computer sends the speed control information to the motor controller through CAN communication, and the motor controller feeds back real-time chassis speed information to the industrial computer through CAN communication.

3.1.3.4 Motion module

The tractor adopts the crawler-type differential drive motion mode, and the left and right driving wheels are controlled by an independent permanent magnet synchronous motor. When the tractor is working, the motor controller makes the tractor move by controlling the rotational speed of the permanent magnet synchronous motor, to realize the efficient and stable running of the tractor. The motor model is TZQ180-4-96B-X.

3.1.3.5 Power module

In the power module, two pieces of 96 V lithium iron phosphate batteries are used to provide power for the tractor. Specifically, the charging power supply voltage is 220 V, the total capacity is 300 Ah, the rated operating power is 8 kW, and the power duration 9 h for transportation operations, which can meet the working time requirements in the greenhouse environment.

3.1.4 Software system design

The navigation software system of the electric crawler tractor consists of three layers: the application layer, the data processing

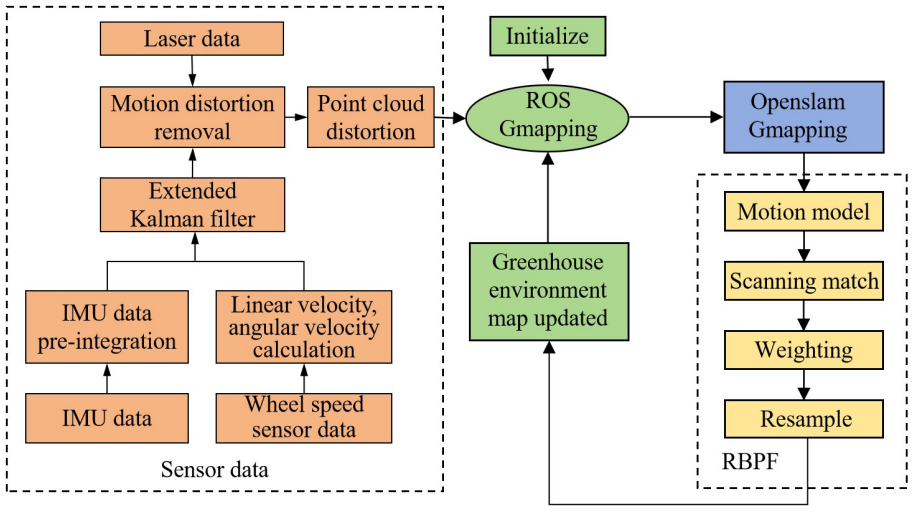


FIGURE 4
Gmapping algorithm-based flowchart for building electric crawler tractor.

layer, and the execution layer. To be specific, the application layer is a navigation task scheduler based on Ubuntu, and the navigation system reaches any target point in the greenhouse environment according to user instructions. The data processing layer is a ROS-based mapping, positioning, navigation, and obstacle avoidance program. It is the core of the navigation software system and the key to realizing the autonomous navigation of electric crawler tractors. The execution layer is a tractor control program based on the Ubuntu open-source real-time operating system. It converts the speed information from the data processing layer to the underlying permanent magnet synchronous motor into the motor speed information, to realize the autonomous navigation of electric crawler tractors in the greenhouse environment (Hu, 2019).

3.2 Principle of the autonomous navigation function

The autonomous navigation process of electric crawler tractors mainly involves greenhouse environment map building, pose estimation of electric crawler tractors, global path planning, and local path planning.

3.2.1 Greenhouse environment map construction

In the design of autonomous navigation system for electric crawler tractor, greenhouse environment map is constructed based on Gmapping algorithm. Firstly, the environmental information is sensed by LiDAR, secondly, the extended Kalman filter algorithm is used to fuse the wheel speed sensor and IMU information to get the approximate position data of the tractor, and this data is used to remove the LiDAR point cloud distortion to get the position and control data of the tractor for a certain period of time to project the approximate trajectory of the tractor, and then the environmental map is projected according to this trajectory to build the corresponding environmental map finally. The flowchart of map building for motorized crawler tractor based on Gmapping algorithm is shown in Figure 4.

Gmapping algorithm is based on the particle filter algorithm (Rao-Blackwell zed Particle Filters, RBPF) and the 2D laser SLAM algorithm. The RBPF algorithm measures the complexity of the algorithm according to the number of particles required to build an environmental map. First, the approximate trajectory of the tractor is calculated using the pose and control data of the tractor in a certain period. Then, the environmental map is calculated according to this trajectory, and the corresponding environmental map is established (Mahmud, 2019). A posteriori estimate of the joint probability of a tractor is represented as Equations (1, 2):

$$p(x_{1:t},m|z_{1:t},u_{1:t-1}) \tag{1}$$

$$p(x_{1:t},m|z_{1:t},u_{1:t-1})=p(m|x_{1:t},z_{1:t})p(x_{1:t}|z_{1:t},u_{1:t-1}) \tag{2}$$

where $z_{1:t}$ denotes the data information observed by LiDAR in period 1 to t, $u_{1:t-1}$ denotes the data controlled by the wheel speed sensor in period 1 to t-1, $x_{1:t}$ denotes the position and posture of the tractor at time t, m represents the raster map, $p(m|x_{1:t},z_{1:t})$ represents the post-probability distribution estimates in the map, and $p(x_{1:t}|z_{1:t},u_{1:t-1})$ represents the posterior probability estimation of tractors.

Based on the RBPF algorithm, particle dissipation and computation are complex in pose estimation. The Gmapping algorithm proposes two improved methods: improved proposal distribution and selective resampling.

(1) Improving proposal distribution. By combining the motion information of the wheel speed sensor with the observation information of the LiDAR to obtain the next generation of particles more accurately, the proposed distribution is improved and gets closer to the target distribution. With an improved proposal distribution, the number of particles and the amount of algorithmic computation can be significantly reduced. The formula for particle weight update is given in Equations (3–7)

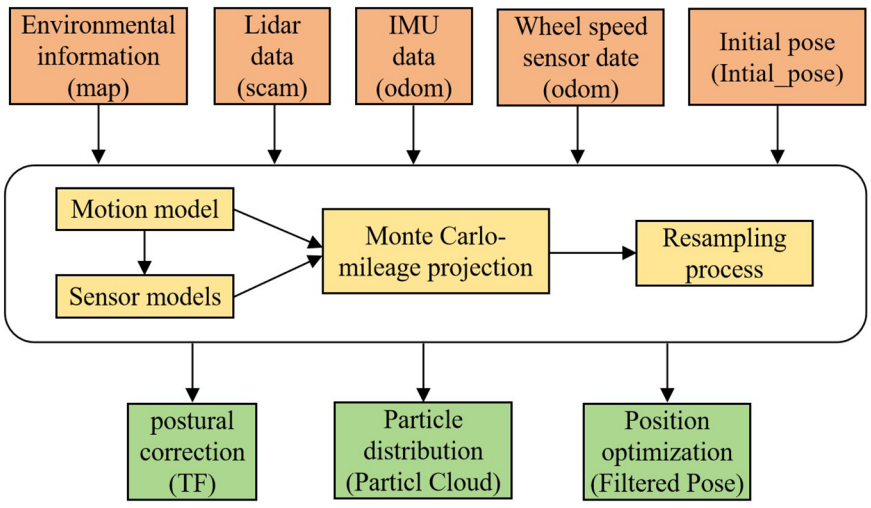


FIGURE 5
AMCL algorithm based electric crawler tractor positional positioning structure diagram.

$$w_t^{(i)}=w_{t-1}^{(i)}\frac{np(z_t|m_{t-1}^{(i)},x^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)},u_{t-1})}{p(x_t^{(i)}|x_{t-1}^{(i)},u_{t-1})}\propto w_{t-1}^{(i)}\cdot p(z_t|m_{t-1}^{(i)},x_t^{(i)}) \quad (3)$$

$$p(x_t|m_{t-1}^{(i)},x_{t-1}^{(i)},z_t,u_{t-1})=\frac{p(z_t|m_{t-1}^{(i)},x_t)p(x_t|x_{t-1}^{(i)},u_{t-1})}{p(z_t|m_{t-1}^{(i)},x_{t-1}^{(i)},u_{t-1})} \quad (4)$$

$$w_t^{(i)}=w_{t-1}^{(i)}\int p(z_t|x')p(x'|x_{t-1}^{(i)},u_{t-1})dx \quad (5)$$

$$u_i^{(t)}=\frac{1}{\eta^{(i)}}\sum_{i=1}^{k}p(z_t|m_{t-1}^{(i)},x_j)p(x_j|x_{t-1}^{(i)},u_{t-1})\cdot(x_j-u_t^{(i)})(x_j-u_t^{(i)})^T \quad (6)$$

$$w_t^{(i)}=w_{t-1}^{(i)}p(z_t|m_{t-1}^{(i)},x_{t-1}^{(i)},u_{t-1})=w_{t-1}^{(i)}\cdot\eta^{(i)} \quad (7)$$

where $w^{(i)}{}_t$ denotes the particle weight, z_t denotes the most recent observation, K represents the number of simulated values, and η is the normalized factor.

(2) Selective resampling. In the Gmapping algorithm, selective resampling is used to determine whether to resample particles by setting an appropriate threshold and according to the dispersion of all particles' weights (i.e., weight variance). Resampling is performed only when the weight of the particle is larger than the set threshold, thus reducing the number of resampling operations, slowing down the rate of particle degradation, and enhancing the accuracy of the algorithm.

3.2.2 Pose estimation of the electric crawler tractor

The position of the electric crawler tractor in the current known greenhouse environment map is determined using the Adaptive Monte Carlo Localization (AMCL) algorithm. As a probabilistic statistical positioning method based on a particle filter, the AMCL algorithm uses the custom KLD method to update the number of particles, fuses the data of IMU and wheel speed sensor with the Kalman filter to maintain the convergence degree of particles, and estimates the probability distribution of tractor pose according to the sampled particles. The pose positioning of the electric track tractor based on the AMCL algorithm is demonstrated in Figure 5. Sufficient random particles are put into the whole pose space, and according to the LiDAR information, each particle is assigned a weight, and the weight of the sampled particles is updated. The larger the weight, the closer the pose is to the real pose.

When the electric crawler tractor is affected by external forces and deviates from the original path, the so-called "kidnap" phenomenon will lead to positioning failure. As the observed information changes, the weight of the particles will decrease, and the AMCL algorithm determines the change in the number of particles and the time by calculating the average weight of the short-term and long-term observation information, as in Equations (8, 9).

$$w_t=w_l+a_l(w_a-w_l) \quad (8)$$

$$w_s=w_s+a_s(w_a-w_s) \quad (9)$$

where w_t and w_s are the long-term and short-term average weights of the observation information, respectively; a_l and a_s are the long-term and short-term average weight coefficients of the observation information, respectively; w_a is the average weight of the particles.

The probability of increasing the random particle population is as in Equation (10):

$$max\left\{0.0,1.0-\frac{w_s}{w_l}\right\} \quad (10)$$

If $w_s/w_l > 1$, the algorithm determines that no random particles are added, and if $w_s/w_l < 1$, the algorithm will increase the number of particles in proportion to the two. Therefore, when the tractor "kidnap" phenomenon occurs, the short-term average weight of the observation information will suddenly decrease, and the AMCL algorithm will increase the number of particles to realize the adaptive repair of global positioning. The number of added particles has a great impact on the algorithm, and the number of added particles can be approximately determined by observing the measurement probability distribution of the sensor and combining the weight of the particles, as in Equation (11):

$$\frac{1}{M}\sum_{m=1}^{M}w_t^m\approx p(z_t|z_{1:t-1},u_{1:t},m) \quad (11)$$

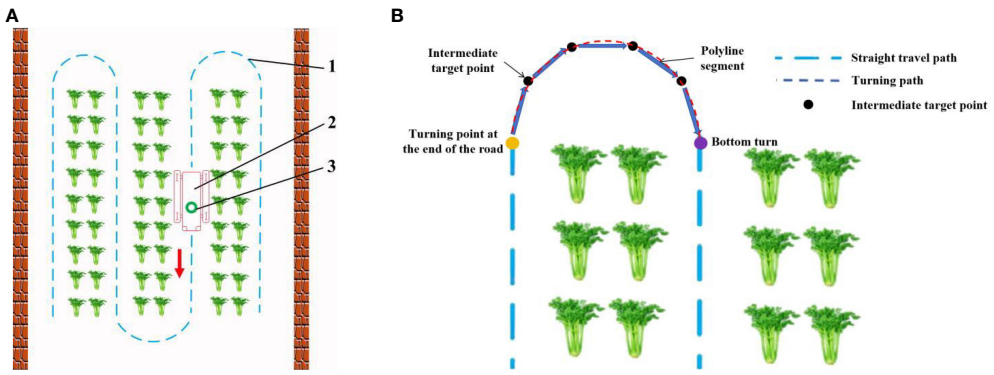


FIGURE 6
Schematic diagram of global path gauge for greenhouse operation of motorized crawler tractor. **(A)** Navigation path diagram. 1 represents a driving track; 2 represents motorized crawler tractor; 3 represents a LiDAR sensor. **(B)** The ground turning path planning diagram.

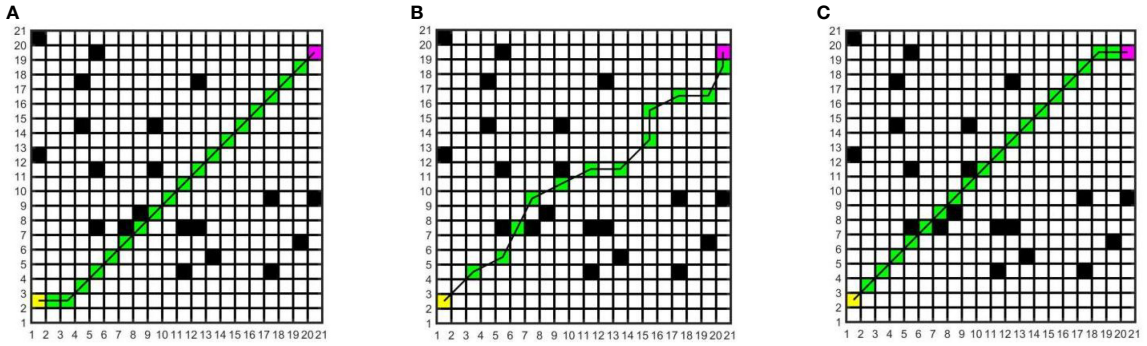


FIGURE 7
Simulation results of path planning for different algorithms in map 1. **(A)** Dijkstra algorithm. **(B)** RRT algorithm. **(C)** A* algorithm.

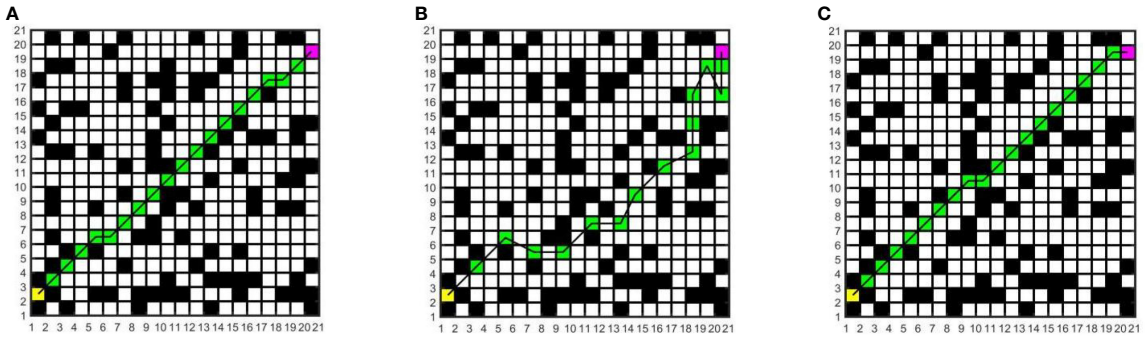


FIGURE 8
Simulation results of path planning for different algorithms in map 2. **(A)** Dijkstra algorithm. **(B)** RRT algorithm. **(C)** A* algorithm.

3.2.3 Global path planning

In the greenhouse environment, the driving path of the electric tractor includes straight-line driving and headland turning. To prevent the tractor from crushing crops in the headland turning process, this study adopts the multi-target point calibration method to perform global path planning, starting from the starting point of the headland turning, passing through several given intermediate target points, and finally reaching the endpoint of the headland turning. As illustrated in Figure 6, the local turning path is divided into several straight lines. Therefore, it is necessary to choose an algorithm with a short running time, fewer inflection points, and a short path length as the global path planning algorithm.

Common global path-planning algorithms include the Dijkstra algorithm, RRT algorithm (Rapid-exploration Random Trees), and A* algorithm (Guruji et al., 2016; Matoui et al., 2017).

In this study, to determine the most suitable global path planning algorithm for the greenhouse planting environment, three different algorithms are compared and simulated in Matlab, and five different raster map scenes are constructed. Specifically, maps 1 to 3 have the same specifications but varying environmental

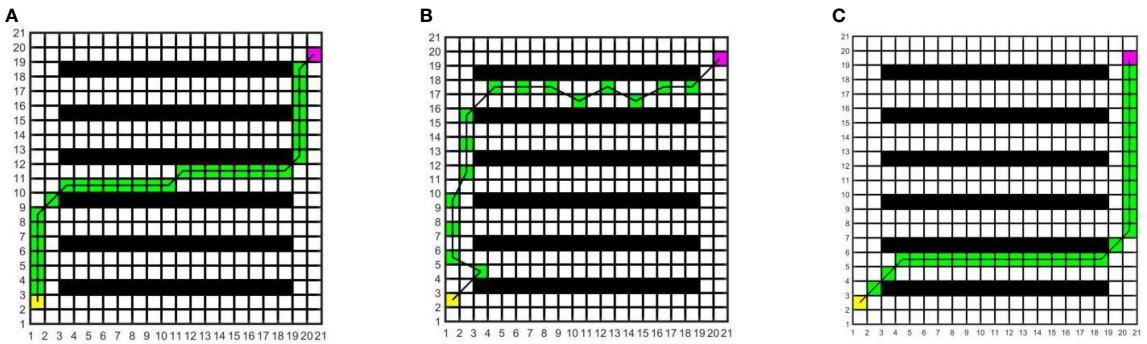


FIGURE 9
Simulation results of path planning for different algorithms in map 3. **(A)** Dijkstra algorithm. **(B)** RRT algorithm. **(C)** A* algorithm.

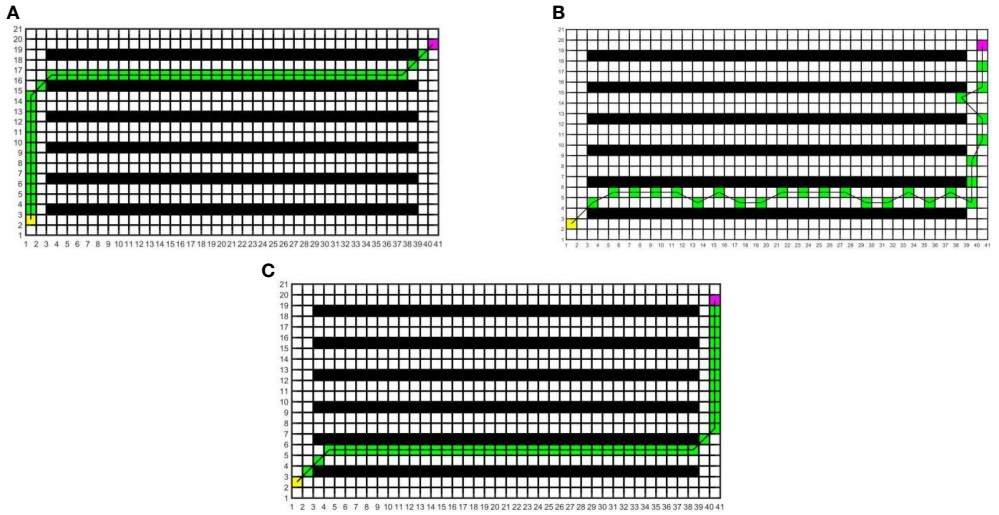


FIGURE 10
Simulation results of path planning for different algorithms in map 4. **(A)** Dijkstra algorithm. **(B)** RRT algorithm. **(C)** A* algorithm.

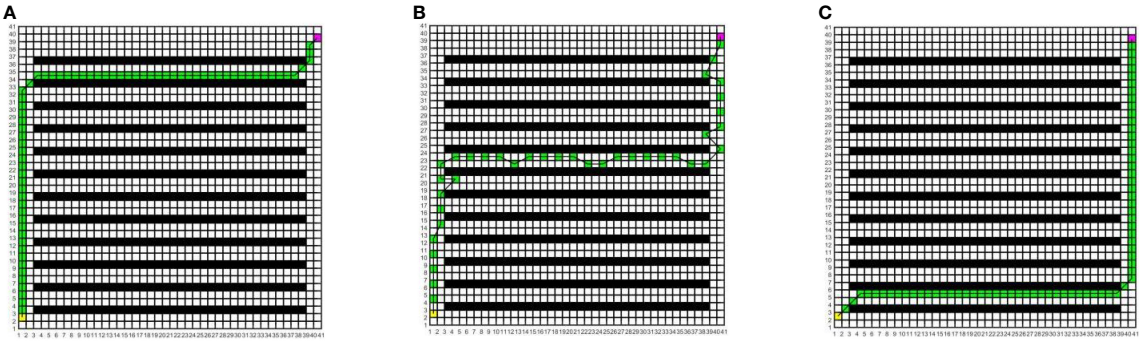


FIGURE 11
Simulation results of path planning for different algorithms in map 5. **(A)** Dijkstra algorithm. **(B)** RRT algorithm. **(C)** A* algorithm.

complexity, and the size is 20×20; Maps 3 to 5 have the same environmental complexity but different specifications, and the sizes of maps 4 and 5 are 20×40 and 40×40, respectively. The Dijkstra algorithm, RRT algorithm, and A* algorithm are compared in five different raster maps, the same starting coordinates are set as (1.5, 2.5), and the end coordinates are set as (cols+0.5, rows-0.5), where cols is the set as the total number of columns in the raster map, and rows is the total number of rows. The yellow block represents the starting point, the purple block represents the endpoint, the white block represents the movable space, the black block represents the obstacle, the green block represents the path node, and the black solid line indicates the generated global path. The running environment of the algorithm is the 64-bit Windows 11 operating system, the PC is Lenovo Rescuer R9000P equipped with the R7-5800H processor and 16 GB memory, and the Matlab version is 2016a. The simulation results are shown in Figures 7–11.

In the five maps, map 1 is a simple environment with relatively few random arrangements of obstacles, map 2 is a chaotic environment with random arrangements of multiple obstacles, and maps 3 to 5 is a simulated greenhouse environment with relatively regular obstacles. For each raster map scene, three path planning algorithms are run 10 times. The initial parameter settings are consistent, and the evaluation indexes include the running time, number of inflection points, and path length. For each algorithm and each map, the average value of each evaluation index for 10 groups is taken, as shown in Table 1.

As can be seen from Table 1, in the five map scenes, the path length planned by the A* algorithm and Dijkstra algorithm is the same, which is reduced by 8.01%, 22.25%, 12.56%, 8.08% and 7.27% respectively compared with the RRT algorithm. In the scenarios of map 1 and map 4, the A* algorithm has the same number of inflection points as the Dijkstra algorithm, which is reduced by 90.00% and 80.00%respectively compared with the RRT algorithm. In the scenes of map 2, map 3, and map 5, the number of inflection points of the A* algorithm is reduced by 25.00% and 63.64%, 57.14% and 75.00%, 40.00% and 87.50% respectively compared with other algorithms. In the scenario of map 1, the running time of the A* algorithm is less than that of the Dijkstra algorithm and more than that of the RRT algorithm. In the scenes of maps 2 to 5,

TABLE 1 Comparison of the running time, inflection points, and path length of different path planning algorithms.

Map scene	Path planning algorithm	Running time/s	Number of inflection points/size	Path length/cm
Map1	Dijkstra algorithm	0.062338	1	26.0416
	RRT algorithm	0.037272	10	28.3093
	A* algorithm	0.041013	1	26.0416
Map2	Dijkstra algorithm	0.060928	4	26.0416
	RRT algorithm	0.085479	11	33.4940
	A* algorithm	0.040065	3	26.0416
Map3	Dijkstra algorithm	0.061533	7	33.0711
	RRT algorithm	0.068319	12	37.8231
	A* algorithm	0.040961	3	33.0711
Map4	Dijkstra algorithm	0.096184	3	53.0711
	RRT algorithm	0.151257	15	57.7376
	A* algorithm	0.042955	3	53.7011
Map5	Dijkstra algorithm	0.168815	5	73.7011
	RRT algorithm	0.283979	24	79.4787
	A* algorithm	0.052560	3	73.0711

the running time of the A* algorithm is reduced by 34.24% and 53.13%, 33.43% and 40.04%, 55.34% and 71.60%, 68.87% and 81.49% respectively, compared with other algorithms. With the increase in obstacle complexity and map size, the RRT algorithm generates more path turning points, and its running time increases significantly. Compared with the RRT algorithm, the Dijkstra algorithm obtains a shorter path and fewer turning points, but it is more time-consuming than the A* algorithm, and the running time increases greatly with the map scale. The A* algorithm obtains the best global path, the path length and running time are shorter, and the improvement effect of search efficiency is more obvious with the increase in the complexity of environment maps. Among them, in map 5, the path planned by the A* algorithm is significantly superior to those of other algorithms. Therefore, this paper adopts the A* algorithm for global path planning of electric crawler tractors in the greenhouse environment.

The A* algorithm provides an efficient direct search method for obtaining the shortest path in the static road network. By combining the advantages of the Dijkstra algorithm and the BFS (Breath First Search) algorithm, it can find an optimal path based on the cost function while improving the efficiency of the algorithm through heuristic search. The cost function of the A* algorithm is given in Equation (12):

$$f(n)=g(n)+h(n) \tag{12}$$

where $f(n)$ is the cost estimate from the initial state through state n to the target state, $g(n)$ is the actual cost of going from the initial state to state n in the state space, $h(n)$ is the estimated cost of the best path from state n to the destination state.

The A* algorithm searches in the direction of approaching the target, and it inspects every node in the search direction during the search process. When a node is reached, the surrounding nodes of the node will be added to OpenList. The node with the smallest estimated value in OpenList will be selected as the next expansion node and added to ClosedList. The process will be repeated until the target node is added to OpenList. When a global planning trajectory from the target point back to the starting point is formed, the pathfinding process is considered successful.

3.2.4 Local path planning

In actual situations, there may be unknown obstacles on the original navigation path of electric crawler tractors. Considering this, based on global path planning, this paper adopts the DWA

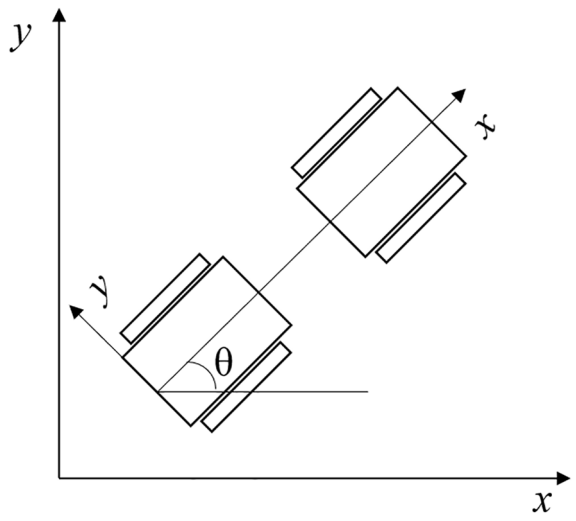


FIGURE 12
Differential kinematics model for electric tracked tractor.

algorithm to detect local environmental information through LiDAR and achieve real-time obstacle avoidance.

3.2.4.1 Kinematics model

The DWA algorithm converts the position control of the tractor into speed control. To use speed sampling to predict the motion trajectory of the electric crawler tractor, the motion model of the electric crawler tractor needs to be analyzed first. The differential kinematic model of the electric crawler tractor is shown in Figure 12.

The DWA algorithm uses the tractor motion model for trajectory simulation and finds the best path in several simulated trajectories. Let $v(t)$ and $\omega(t)$ represent the linear and angular velocity of the tractor at time t in the world coordinate system, respectively (Mahmud et al., 2019; Molinos et al., 2019). In the sampling period t, the displacement is small, and the tractor moves in a straight line at an approximately uniform speed. Then, the pose increment of the two adjacent moments is as in Equation (13):

$$\begin{cases} \Delta x = v_t \Delta t \cos(\theta_t) \\ \Delta y = v_t \Delta t \sin(\theta_t) \\ \Delta \theta = \omega \Delta t \end{cases} \tag{13}$$

The pose at time t+1 can be expressed as in Equation (14):

$$\begin{cases} x(t+1) = x(t) + v_t \Delta t \cos(\theta_t) \\ y(t+1) = y(t) + v_t \Delta t \sin(\theta_t) \\ \theta(t+1) = \theta(t) + \omega \Delta t \end{cases} \tag{14}$$

where $x(t)$, $y(t)$, $\theta(t)$ – the position and posture of the tractor in the world coordinate system at time t.

3.2.4.2 Velocity sampling

In the actual greenhouse environment, the DWA algorithm transforms the obstacle avoidance problem into three-speed constraints of the electric crawler tractor in the speed space, including the maximum and minimum speed constraints of the tractor, the motor acceleration and deceleration constraints, and the braking distance constraints. The above constraints can restrict the movement speed of the tractor within a certain range, and the specific constraints are represented as follows:

3.2.4.2.1 Maximum and minimum speed constraints

Since the hardware performance of the tractor sets limitations on the minimum and maximum speeds of the tractor, the sampling speed of the tractor should be controlled within the interval of the optimal angular speed and linear speed of the tractor, and the constraint is given in Equation (15)

$$V_m = \{(v,\omega) | v \in [v_{min}, v_{max}], \omega \in [\omega_{min}, \omega_{max}]\} \tag{15}$$

where v_{max} and v_{min} are tractor maximum line speed and minimum line speed, and ω_{min} and ω_{max} are maximum angular speed and minimum angular speed of the tractor.

3.2.4.2.2 Motor acceleration and deceleration constraints

The acceleration of the tractor is subject to the output torque of the motor, and the space sampling of the velocity vector should fall within the tolerable range of the motor torque. The constraint is represented as in Equation (16)

$$V_d = \{(v,\omega) | v \in [v_c - \dot{v}_b \Delta t, v_c + \dot{v}_a \Delta t], \omega \in [\omega_c - \dot{\omega}_b \Delta t, \omega_c + \dot{\omega}_a \Delta t]\} \tag{16}$$

where v_c is current linear velocity, ω_c is the current angular velocity, $\dot{v}_a$ is maximum linear acceleration, $\dot{\omega}_a$ is maximum angular acceleration, $\dot{v}_b$ is maximum line deceleration, $\dot{\omega}_b$ is maximum angular deceleration.

3.2.4.2.3 Braking distance constraints

To guarantee that the tractor stops before hitting a random obstacle, the linear speed and angular speed of the tractor before reaching the obstacle should be reduced to 0 under the condition of maximum deceleration. The constraint is expressed as in Equation (17)

$$V_a = \left\{(v,\omega), v \le \sqrt{2dist(v,\omega)\dot{v}_b}, \omega \le \sqrt{2dist(v,w)\dot{\omega}_b}\right\} \tag{17}$$

where $dist(v, \omega)$ is the nearest distance between the tractor and the obstacle.

Finally, the speed of the electric crawler tractor needs to take the intersection of the above three constraint spaces, i.e., the dynamic window speed V_r should satisfy the following condition, as shown in Equation (18):

$$V_r = V_m \cap V_d \cap V_a \tag{18}$$

3.2.4.2.4 Evaluation function

The DWA algorithm evaluates the simulated trajectories in the set of speed groups, selects the motion trajectories with the best performance and sends them to the chassis of the tractor, controls the tractor to finish the evading task, and evaluates the trajectories using the evaluation function. The evaluation function is as in Equation (19)

$$G(v,\omega) = \sigma(\alpha \cdot heading(v,\omega) + \beta \cdot dist(v,\omega) + \gamma \cdot velocity(v,\omega)) \tag{19}$$

The evaluation functions of the DWA algorithm involve azimuth angle, linear velocity, and the nearest distance between simulated trajectory and obstacles, and each evaluation function needs to be normalized (Zhang, 2018; Mahmud et al., 2019), as shown in Equations (20–22):

$$normal_heading(i) = \frac{heading(i)}{\sum_{i=1}^{n} heading(i)} \tag{20}$$

$$normal_dist(i) = \frac{dist(i)}{\sum_{i=1}^{n} dist(i)} \tag{21}$$

$$normal_velocity(i) = \frac{velocity(i)}{\sum_{i=1}^{n} velocity(i)} \tag{22}$$

where $heading$ (v, ω) is the azimuth evaluation subfunction evaluates the azimuth deviation between the direction of the end of the simulated trajectory and the target at the current speed; $dist$ (v, ω) is the obstacle evaluation subfunction represents the closest distance between the simulated trajectory of the corresponding

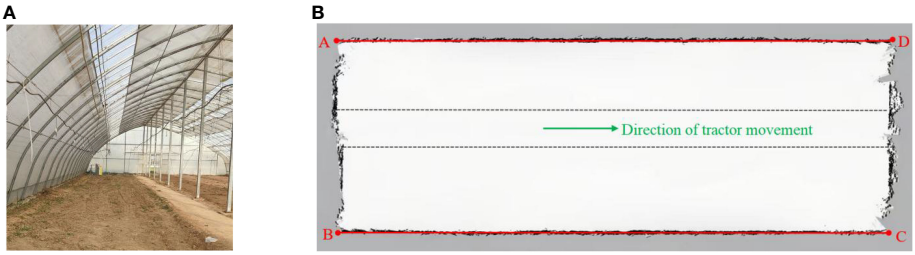


FIGURE 13
Greenhouse environments and maps created by the Gmapping algorithm. **(A)** Greenhouse environment. **(B)** Map created by Gmapping algorithm.

velocity group and the obstacle. If there is no obstacle on the current trajectory, *dist* (v, ω) is set as a constant; *velocity* (v, ω) is the velocity evaluation subfunction represents the velocity magnitude of the simulated trajectory; σ is smoothing function; α is direction influence coefficient, the closer the tractor is to the target point, the greater the α value; β is safety distance factor, the closer the tractor is to the obstacle, the greater the β value; γ is velocity influence coefficient, the higher the speed of the tractor, the greater the gamma value.

4 Results and discussion

The demonstration greenhouse of the Rougu fruit and vegetable professional cooperative in Yangling District of Shaanxi Province is selected to test the mapping performance and navigation accuracy respectively. The greenhouse environment is relatively closed, the size of the greenhouse is 43 m×15.5 m, and the middle road is a relatively flat cement road with regular distribution, and vegetable planting areas are on both sides of the road, as demonstrated in Figure 13A.

TABLE 2 Relative error of facility environment map.

Position	Real value/m	Atlas value/m	Misalignment/ m
AB	15.50	15.46	0.04
BC	43	43.16	0.16
CD	15.50	15.44	0.06
AD	43	42.87	0.13

4.1 Construction accuracy test

Before the drawing test, the systematic error of the wheel speed sensor of the electric crawler tractor is corrected to reduce the positioning error. The drawing test of the electric crawler tractor was conducted on the transport channel of the greenhouse. The transport channel was relatively smooth, and the slippage between the track and the ground was small. The remote-controlled tractor moves slowly in the greenhouse and uses LiDAR to obtain information about the surrounding environment. However, the LiDAR sensor will shake when gullies are encountered. Therefore, the remote-controlled tractor needs to scan and build the map back and forth to ensure that the environmental map is complete.

The built environment map is shown in Figure 13B. In this figure, the solid red line is the reference line at both ends of the greenhouse, the green arrow indicates the direction of tractor movement, and the dotted black line denotes the cement road taken by the map construction. Four map endpoints A, B, C, and D are selected respectively, and the distance between the adjacent endpoints is compared with the actual size of the map construction area, as listed in Table 2.

It can be seen from Table 2 that the tractor will continuously produce cumulative errors during mapping, leading to a slight distortion at the edge of the map. The maximum lateral deviation of the built environment map is 6 cm and the maximum

TABLE 3 Navigation accuracy test results.

Test serial number	Lateral deviation/cm			Heading deviation/°		
	Maximum	Average	Standard deviation	Maximum	Average	Standard deviation
1	-12.40	+6.78	7.90	-13.30	-6.52	7.16
2	-11.80	-6.34	7.42	-12.30	4.48	5.72
3	+8.80	-4.69	5.65	+11.20	5.85	6.56
4	11.60	+6.44	7.01	+10.40	-5.73	6.59
5	-9.50	-5.34	6.06	-12.20	-6.07	7.22

longitudinal deviation is 16 cm, which can meet the map accuracy requirements for navigating electric crawler tractors in the greenhouse.

4.2 Navigation accuracy test

The speed of the electric crawler tractor is set to 1 m/s, the greenhouse environment map is loaded, and the position and posture of the starting point of the tractor operation and the target point and posture of the navigation are set. The AMCL algorithm is employed to obtain real-time position information of the electric crawler tractor, and the A* algorithm is utilized to plan the globally optimal path from the current position of the tractor to the navigation target point. The environment map after expansion is obtained by real-time detection of environmental obstacle information by LiDAR. When an obstacle is detected, the scanned laser point cloud information is fed into the DWA algorithm to replan the optimal path of the tractor.

The test was conducted in five groups with a test distance of 20 m. In each group of tests, data sampling was measured at an interval of 2 m. After the end of navigation, the lateral deviation and heading deviation of the tractor at each sampling point were measured. The left side of the target point is positive deviation, the right side of the target point is negative deviation, the actual course on the left side of the target course is positive deviation, and the actual course on the right side of the target course is negative deviation. The mean deviation and standard deviation of the tracking path were calculated based on the test data. After each test, the initial position of the tractor was reset to prevent the accumulation of errors from affecting the test results. The test results are presented in Table 3.

It can be seen from Table 3 that after five tests in the greenhouse, the maximum transverse deviation is no more than 13cm, the average transverse deviation is less than 7cm, and the mean standard deviation is less than 8cm. The maximum heading deviation is less than, the average heading deviation is less than 14°, and the standard deviation is less than 7°. These results indicate that the system developed in this study can meet the accuracy requirements for autonomous navigation of the electric crawler tractor in the greenhouse.

5 Conclusion

To improve the intelligence level of electric crawler tractors in the greenhouse, based on LiDAR technology, this paper designs an autonomous navigation system of electric crawler tractors for the greenhouse planting environment. The hardware part mainly consists of LiDAR, IMU, wheel speed sensor, industrial computers, etc. The software core control layer is developed based on ROS, and information exchange is realized through distributed node communication. The Gmapping algorithm is employed to build the greenhouse environment map. The mapping test shows that the maximum lateral deviation of the built map is 6cm, and the maximum longitudinal deviation is 16cm, which can meet the map accuracy requirements in the greenhouse. To select the path planning algorithm suitable for greenhouse operations, common path planning algorithms are simulated, and the results suggest that the A* algorithm obtains the best global path, its path length and running time are shorter, and the improvement effect of search efficiency is more obvious with the increase in the complexity of the environment map. Therefore, the A* algorithm is used in this study for global path planning, and the DWA algorithm is used for local path planning. The accuracy test results of the navigation system indicate that the maximum lateral deviation is less than 13cm, the average lateral deviation is less than 7cm, and the standard deviation is less than 8cm. Meanwhile, the maximum heading deviation is no more than, the average heading deviation is less than 7°, and the standard deviation is less than 8°. The accuracy meets the navigation and positioning requirements of electric crawler tractors in greenhouse transportation. At present, this study can only solve the simple positioning and navigation problems in the greenhouse. In future work, different navigation strategies will be formulated according to the specific operation conditions of the greenhouse, and the positioning and navigation accuracy can be further enhanced.

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

HG: Funding acquisition, Project administration, Supervision, Writing – review & editing. YL: Investigation, Methodology, Writing – original draft. HW: Investigation, Methodology, Writing – original draft. TW: Data curation, Validation, Writing – original draft. LR: Data curation, Validation, Writing – original draft. HW: Validation, Writing – original draft. ZW: Validation, Writing – original draft. CW: Validation, Writing – original draft. JZ: Validation, Writing – original draft. YH: Visualization, Writing – review & editing. SG: Visualization, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the National Key R&D Program of China (No.2023YFD2000300); the Shaanxi Provincial Technology Innovation Guidance Special Project (No. 2021QFY08-05) and the Shaanxi Province Key Research and Development Plan Project (No.2022ZDLNY03-04).

Conflict of interest

Author JZ was employed by the company BYD Automobile Company Limited. Authors YH and SG were employed by the company Xi 'an Dongyang Machinery Company Limited.

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

Arad, B., Balendonck, J., Barth, R., Ben-Shahar, O., Edan, O., Hellström, T., et al. (2020). Development of a sweet pepper harvesting robot. *J. Field Robot.* 37, 1027–1039. doi: 10.1002/rob.21937

Dang, X., Rong, Z., and Liang, X. (2021). Sensor fusion-based approach to eliminating moving objects for SLAM in dynamic environments. *Sensors* 21, 230. doi: 10.3390/s21010230

Fu, Z., Liu, Y., Si, P., Tao, F., and Wang, N. (2023a). A multi-sensor high precision location method in urban environment. *IEEE Syst. J.* 17, 6611–6622. doi: 10.1109/JSYST.2023.3316140

Fu, Z., Liu, Y., Si, P., Tao, F., and Wang, N. (2023b). Adaptive fusion multi-IMU confidence level location algorithm in the absence of stars. *IEEE Sensors J.* 23, 18644–18655. doi: 10.1109/JSEN.2023.3294508

Guruji, A., Agarwal, H., and Parsediya, D. (2016). Time-efficient A* algorithm for robot path planning. *Proc. Technol.* 23, 144–149. doi: 10.1016/j.protcy.2016.03.010

Hiremath, S., van der Heijden, G., van Evert, F., Stein, A., and ter Braak, C. (2014). Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter. *Comput. Electron. Agricult.* 100, 41–50. doi: 10.1016/j.compag.2013.10.005

Hou, J., Pu, W., Li, T., and Ding, X. (2020). Development of dual-lidar navigation system for greenhouse transportation robot. *Trans. Chin. Soc. Agric. Engineering.* 36, 80–88. doi: 10.11975/j.issn.1002-6819.2020.14.010

Hu, C. (2019). *ROS robot development practice* (Bei Jing: China Machine Press).

Hu, J., Yang, Y., Li, Y., Hou, J., Sun, Z., Wang, H., et al. (2024). Analysis and prospect of the environmental control systems for greenhouse. *Trans. Chin. Soc. Agric. Engineering.* 40, 112–128. doi: 10.11975/j.issn.1002-6819.202310214

Hu, L., Wang, Z., Wang, P., He, J., Jiao, J., Wang, C., et al. (2023). Agricultural robot positioning system based on laser sensing. *Trans. Chin. Soc. Agric. Engineering.* 39, 1–7. doi: 10.11975/j.issn.1002-6819.202211144

İrsel, G., and Altinbalik, T. M. (2018). Adaptation of tilt adjustment and tracking force automation system on a laser-controlled land leveling machine. *Comput. Electron. Agricult.* 150, 374–386. doi: 10.1016/j.compag.2018.04.021

Ji, Y., Li, H., Zhang, M., Wang, Q., Jia, J., and Wang, K. (2018). Navigation system for inspection robot based on LiDAR. *Trans. Chin. Soc. Agric. Machinery.* 49, 14–21. doi: 10.6041/j.issn.1000-1298.2018.02.002

Jia, S., Li, J., Qiu, Q., and Tang, H. (2015). New corridor edge detection and navigation for greenhouse mobile robots based on laser scanner. *Trans. Chin. Soc. Agric. Engineering.* 31, 39–45. doi: 10.11975/j.issn.1002-6819.2015.13.006

Jiang, Y., Li, Q., Ye, W., and Liu, G. (2023). Development of a GNSS/INS-based automatic navigation land levelling system. *Comput. Electron. Agric.* 213, 108187. doi: 10.1016/j.compag.2023.108187

Jiang, S., Wang, S., Yi, Z., Zhang, M., and Lv, X. (2022). Autonomous navigation system of greenhouse mobile robot based on 3D Lidar and 2D Lidar SLAM. *Front. Plant Science.* 13. doi: 10.3389/fpls.2022.815218

Kurashiki, K., Fukao, T., Nagata, J., Ishiyama, K., Kamiya, T., and Murakami, N. (2010). Laser-based vehi-cle control in orchard. *IFAC Proc. Volumes.* 43, 127–132. doi: 10.3182/20101206-3-JP-3009.00022

Li, G., Han, Y., and Wang, X. (2023a). Implementation of indoor path planning and navigation for mobile robots based on ROS. *Internet Things Technologies.* 13, 87–89. doi: 10.16667/j.issn.2095-1302.2023.08.022

Li, Y., Wang, D., Li, Q., Cheng, G., Li, Z., and Li, P. (2023b). Advanced 3D navigation system for AGV in complex smart factory environments. *Electronics* 13, 130. doi: 10.3390/electronics13010130

Liu, X., Wang, G., Liu, N., Zhang, C., Qi, X., Wang, Y., et al. (2022). Development status and trend analysis of facility agricultural machinery. *Agric. Technol. Equip.*, 61–62.

Liu, Z., Zhang, Z., Luo, X., Wang, H., Huang, P., and Zhang, J. (2018). Design of automatic navigation operation system for Lovol ZP9500 high clearance boom sprayer based on GNSS. *Trans. Chin. Soc. Agric. Engineering.* 34, 15–21. doi: 10.11975/j.issn.1002-6819.2018.01.003

Lu, K., Yang, S., Lian, S., and Feng, A. (2024). Implementation of autonomous navigation simulation control for classroom disinfection robot. *Robot Technique and Application.* 32, 39–42. doi: 10.13695/j.cnki.12-1222/o3.2024.01.002

Mahmud, M., Zainal Abidin, M., Mohamed, Z., Abd Rahman, M., and Iida, M. (2019). Multi-objective path planner for an agricultural mobile robot in a virtual greenhouse environment. *Comput. Electron. Agricult.* 157, 488–499. doi: 10.1016/j.compag.2019.01.016

Matoui, F., Boussaid, B., Metoui, B., Frej, G., and Abdelkrim, M. (2017). Path planning of a group of robots with potential field approach: decentralized architecture. *IFAC Papers Line.* 50, 11473–11478. doi: 10.1016/j.ifacol.2017.08.1822

Molinos, E., Llamazares, Á., and Ocaña, M. (2019). Dynamic window based approaches for avoiding obstacles in movin. *Robotics Autonomous Systems.* 118, 112–130. doi: 10.1016/j.robot.2019.05.003

Pu, W., Liu, X., Chen, H., Xu, H., and Liu, Y. (2023). LIDAR point cloud correction and location based on multisensor fusion. *Laser Optoelectronics Progress.* 60, 275–282.

Radcliffe, J., Cox, J., and Bulanon, D. (2018). Machine vision for orchard navigation. *Comput. Industry.* 98, 165–171. doi: 10.1016/j.compind.2018.03.008

Shen, Y., Xiao, X., Liu, H., and Zhang, X. (2023). Real-time localization and mapping method for agricultural robot in orchards based on LiDAR/IMU tight-couping. *Trans. Chin. Soc. Agric. Machinery.* 54, 20–28+48. doi: 10.6041/j.issn.1000-1298.2023.11.002

Si, L., Wang, Z., Wei, D., Gu, J., Yan, H., Tan, C., et al. (2023). Positioning and navigation method of underground drilling robot for rock-burst prevention based on IMU-LiDAR tight coupling. *J. China Coal Soc.*, 1–16. doi: 10.13225/j.cnki.jccs.2023.0832

Tan, C., Li, Y., Wang, D., Mao, W., and Yang, F. (2020). Review of research on automatic guidance of agricultural vehicles. *J. Agric. Mechanization Res.* 42, 7–14+32. doi: 10.13427/j.cnki.njyi.2020.05.002

Wang, H., and An, T. (2024). Transportation robot system based on gmapping and A* algorithm. *Ind. Control Computer.* 37, 19–21.

Wang, Y., Li, H., Niu, D., Li, X., and Liu, G. (2014). Real-time path planning and navigation in GNSS-controlled land leveling. *Trans. Chin. Soc. Agric. Machinery.* 45, 271–275. doi: 10.6041/j.issn.1000-1298.2014.S0.044

Wang, Y., Yang, F., Wang, T., and Liu, Q. (2013). Research on visual navigation and remote monitoring technology of agricultural robot. *Int. J. Smart Sens. Intelligent Systems.* 6, 466–481. doi: 10.21307/ijssis-2017-550

Wu, K. J., Guo, C. X., Gaorgiou, G., and Roumeliotis, S. I. (2017). *Vins on Wheels// 2017 IEEE International Conference on Robotics and Automation(ICRA)* (Singapore: IEEE), 5155–5162. doi: 10.1109/ICRA.2017.7989603

Ye, H., Chen, Y., and Liu, M. (2019). *Tightly coupled 3D LiDAR inertial odometry and mapping//2019 Imternational Conference on Robotics and Automation (ICRA)* (Montreal, QC, Canada: IEEE), 3144–3150. doi: 10.1109/ICRA.2019.8793511

Zhang, C. (2018). Path planning for robot based on chaotic artificial potential field method. *IOP Conf. Series: Materials Sci. Engineering.* 317, 12056. doi: 10.1088/1757-899X/317/1/012056

Zhang, M., Ji, Y., Li, S., Cao, R., Xu, H., and Zhang, Z. (2020). Research progress of agricultural machinery navigation technology. *Trans. Chin. Soc. Agric. Machinery.* 54, 1–18. doi: 10.6041/j.issn.1000-1298.2020.04.001

Zhang, Z., Noguchi, N., Ishii, K., Yang, L., and Zhang, C. (2013). Development of a robot combine harvester for wheat and paddy harvesting. *IFAC Proc. Volumes.* 46, 45–48. doi: 10.3182/20130327-3-JP-3017.00013

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 05 June 2024
DOI 10.3389/fpls.2024.1365266

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Ahmad Jahanbakhshi,
University of Mohaghegh Ardabili, Iran
Eri Hayashi,
Japan Plant Factory Association, Japan

*CORRESPONDENCE
Soo Chung
soochung@snu.ac.kr

RECEIVED 04 January 2024
ACCEPTED 10 May 2024
PUBLISHED 05 June 2024

CITATION
Kim J-SG, Moon S, Park J, Kim T and Chung S (2024) Development of a machine vision-based weight prediction system of butterhead lettuce (*Lactuca sativa* L.) using deep learning models for industrial plant factory. *Front. Plant Sci.* 15:1365266. doi: 10.3389/fpls.2024.1365266

COPYRIGHT
© 2024 Kim, Moon, Park, Kim and Chung. 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.

Development of a machine vision-based weight prediction system of butterhead lettuce (*Lactuca sativa* L.) using deep learning models for industrial plant factory

Jung-Sun Gloria Kim[1,2], Seongje Moon[1], Junyoung Park[1,2], Taehyeong Kim[1,2,3] and Soo Chung[1,2,3]*

[1]Department of Biosystems Engineering, Seoul National University, Seoul, Republic of Korea, [2]Integrated Major in Global Smart Farm, Seoul National University, Seoul, Republic of Korea, [3]Research Institute of Agriculture and Life Sciences, Seoul National University, Seoul, Republic of Korea

Introduction: Indoor agriculture, especially plant factories, becomes essential because of the advantages of cultivating crops yearly to address global food shortages. Plant factories have been growing in scale as commercialized. Developing an on-site system that estimates the fresh weight of crops non-destructively for decision-making on harvest time is necessary to maximize yield and profits. However, a multi-layer growing environment with on-site workers is too confined and crowded to develop a high-performance system.

This research developed a machine vision-based fresh weight estimation system to monitor crops from the transplant stage to harvest with less physical labor in an on-site industrial plant factory.

Methods: A linear motion guide with a camera rail moving in both the x-axis and y-axis directions was produced and mounted on a cultivating rack with a height under 35 cm to get consistent images of crops from the top view. Raspberry Pi4 controlled its operation to capture images automatically every hour. The fresh weight was manually measured eleven times for four months to use as the ground-truth weight of the models. The attained images were preprocessed and used to develop weight prediction models based on manual and automatic feature extraction.

Results and discussion: The performance of models was compared, and the best performance among them was the automatic feature extraction-based model using convolutional neural networks (CNN; ResNet18). The CNN-based model on automatic feature extraction from images performed much better than any other manual feature extraction-based models with 0.95 of the coefficients of determination (R^2) and 8.06 g of root mean square error (RMSE). However, another multiplayer perceptron model (MLP_2) was more appropriate to be adopted on-site since it showed around nine times faster inference time than

CNN with a little less R^2 (0.93). Through this study, field workers in a confined indoor farming environment can measure the fresh weight of crops non-destructively and easily. In addition, it would help to decide when to harvest on the spot.

KEYWORDS

controlled-environment agriculture, convolutional neural networks, commercialized plant factory, computer vision, data acquisition system, indoor farming, linear motion guide, regression model

1 Introduction

Food security has been seriously threatened by the global pandemic and geopolitical tensions such as COVID-19 and the Russia–Ukraine war (Farcas et al., 2020; Ben Hassen and El Bilali, 2022), along with overpopulation, lower arable lands, and climate change (Hati and Singh, 2021). Overpopulation is anticipated to reach 9.7 billion in 2050, and the demand for food will increase by 70% from the current levels (World Population Prospects - Population Division - United Nations, 2022). In addition, arable land has been decreasing while the urban landscape has expanded (Brain et al., 2023). Climate change, has worsened and has become a reality. According to the European Commission's Copernicus Climate Change Service, July 2023 was the hottest month in Europe because records experienced severe drought, which has significantly impacted agricultural yields. Therefore, improving food production has become one of the most critical issues in the world, and investments in related systems have risen, requiring commercialization (Adenäuer et al., 2023). In this context, several studies are expected to improve crop production systems.

Controlled-environment agriculture (CEA), such as plant factories, has become one of the most representative ways to improve the efficiency of crop production because it allows growers to use less cultivated land and cultivate crops year-round while minimizing damage from diseases and insects (Kozai and Sasaki, 2013; Mitchell, 2022). This can lead to increased food access nationwide, strengthening food security against the global problems mentioned above. Currently, plant factories with state-of-the-art technology, such as precision farming, are drawing public attention, investment, and development (Hati and Singh, 2021; Chamara et al., 2022). It allows cultivators to collect information on the growth state of crops, such as morphological features and weight in real-time, and automatically controls the environment, resulting in maximum yields and quality (Reyes-Yanes et al., 2020).

Among growth information, the fresh weight of crops has been considered an important indicator for monitoring plant growth to increase productivity and profitability (Mokhtar et al., 2022; Gang et al., 2022a; Tong et al., 2023). This indicates that plant growth rate and uniformity play a significant role in monitoring harvest time and the occurrence of disease (Zhang et al., 2022). Notably, plant factories must track the fresh weight of crops because they are cultured in bulk in the system. This is because the owner should meet the ordered quantity and quality, including certain sizes and weights, and have a specific date of shipping when an indoor farm is commercialized (Petropoulou et al., 2023). A delay in shipment is one of the most severe problems in commercial indoor farming. From economic and industrial viewpoints, a delay in shipment increases operation and maintenance costs and decreases product quality, undoubtedly leading to a decline in profits. Therefore, an automatic weight estimation system is required to determine the exact harvest time and to reduce unnecessary expenses.

Normally, most weighing processes are performed only after harvesting. For continuous monitoring, fresh weight could not be measured during cultivation. In addition, workers often carry the crops to an existing weighing system, such as a hanging load cell, electronic, or floor scale, to measure their fresh weight. This requires a lot of labor, working hours, and increased costs. Direct contact with crops is inevitable, increasing the stress of crops and leading to growth inhibition. Therefore, an automatic and nondestructive weighing system is necessary.

Recently, machine vision has been implemented for weight prediction because of its noninvasive and nonintrusive characteristics to demonstrate a contactless weighing system (Hati and Singh, 2021; Lou et al., 2022; Ojo and Zahid, 2023). Most previous studies collected images by phone, which caused considerable confusion when developing prediction models (Moon et al., 2020). In addition, there was a case where only a small amount of data could be obtained despite using multiple cameras because the locations of the cameras were fixed and had limited coverage of view (Lee, 2008). They have also been implemented in experimental greenhouses and laboratories, controlling most of the environment (Jiang et al., 2018; Reyes-Yanes et al., 2020). Many researchers have used public data, such as datasets collected from the lettuce planting laboratory at Wageningen University and research in the Netherlands, to develop machine-vision-based models (Lin et al., 2022; Zhang et al., 2022; Gang et al., 2022a). However, a large number of consistent quality images of crops are necessary to develop a fresh weight prediction model using images collected by the machine vision system. The best approach is to set up an on-site data acquisition system to optimize the actual users of the system.

Actual industrial plant factories operate under different conditions, and it is difficult to obtain images in any case. In greenhouses and laboratories, the air space above crops is much larger than that in plant factories, because most systems do not use vertically stacked growing beds. Therefore, cameras can be fixed on the top above crops with a height of over 1 m (Lee, 2008; Ojo and Zahid, 2023), whereas it is difficult to apply in plant factories that require a limited height of less than 50 cm. The limited height increases the difficulty of installing an image-acquisition system. Furthermore, there are differences between experimental and industrial plant factories. Experimental locations can be modified in systems as intended so that they have only a few hindrances in the experimental process (Chen et al., 2016). However, industrial plant factories had to consider the high humidity and sloshed water generated when workers cleaned the floors and cultivating beds during the experiment. In addition, the machine vision system had to occupy minimal space and not interrupt the workflow of workers. Therefore, this study demonstrates an appropriate machine vision system that acquires images under harsh conditions such as narrow, crowded, and high humidity.

Another important factor to consider in industrial factories is that economic and commercial crops should be prudently chosen to enhance the profitability of commercialized plant factories. Lettuce is a relatively fast-growing and commonly produced crop worldwide in plant factories with artificial lighting (PFALs) (Kozai et al., 2019). Lettuce has several varieties, including iceberg, red leaf lettuce, and Latin lettuce. According to the report 'Hydroponic lettuce market is thriving worldwide during the forecast period 2023–2030,' butterhead lettuce had the highest market share as of 2022 among the hydroponic lettuce. In this study, butterhead lettuce (*Lactuca sativa* var. capitata L. nidus tenerrima) has been selected since it is one of the most famous crops cultivated in commercialized plant factories such as Aerofarms from New Jersey in the United States of America, Vertical Roots from Edmonton in Canada and PlanTfarm from Pyeongtaek in South Korea consumed a lot as a salad.

This study developed a non-destructive fresh weight prediction system for butterhead lettuce in an industrial plant factory using images collected by an automatic image acquisition system. This machine vision system collects data on crops without contact and with consistency immediately. In addition, we compared manual feature extraction models with various combinations of parameters to automatic feature extraction-based models to determine the best fresh weight prediction model. This on-site data-based model is expected to be better utilized in the field, and it is expected to help estimate the exact fresh weight state of the crop in real time and when to harvest.

2 Materials and methods

2.1 Field experiment site and target crop

The experiment was conducted inside a T-Farm2 (PlanTFarm Co., Ltd., Gyeonggi-do, Republic of Korea). Air temperature, relative humidity, and carbon dioxide concentration were maintained at 21.2 ± 3°C, 75.6 ± 15%, and 827 ppm on average, respectively, which fits the growth of butterhead lettuce since the optimal range of air temperature and air relative humidity for lettuce is 18°C–25°C and 60%–80%, respectively (Boros et al., 2023). In addition, carbon dioxide stayed in the range of 788 ppm–917 ppm, which is an appropriate concentration for cultivating the lettuce (Zhang et al., 2017). In the case of the nutrient solution, the pH was in the range of 6.29–6.72, with a mean value of 6.46.

A total of 159 Butterhead lettuce (*L, sativa* var. capitata L. nidus tenerrima) was cultivated for 102 days (from 30 September 2022 to 10 January 2023. The location of the cultivation rack of the butterhead lettuce for the experimental application was on the fifth level from the bottom of the cultivation racks to avoid interference from workers during cultivation (Figures 1, 2B).

2.1.1 Image acquisition system

An image acquisition system was designed based on a linear motion guide to attain consistent images automatically and stably on site under harsh environmental conditions, such as high humidity and confined space. The data acquisition system was produced by a company named Robowill from the Republic of Korea.

An image acquisition system was installed at the top of the cultivation bed. The height from the bottom to the ceiling of the cultivating bed was 330 mm (Figure 3A), and the LEDs were installed 25 mm away from the ceiling (Figure 3B). The height of the LEDs was 25 mm, and the distance between them was 100 mm. Cameras were installed 50 mm from the ceiling to avoid disturbing the illumination path.

The present system allowed multiple cameras to be installed within a narrow space, such as the top of a plant factory, by moving the installed terminal rail (Figure 4B) connected to cameras in the longitudinal or width direction to capture unstructured data of the crops located below. In addition to taking photos of images, driving parts (Figure 4A), such as motors, were safe under the high temperature and humidity of the growing environment. A clamp (Figure 4A) fixes the finishing plate provided at the end of the rail frame and is removably attached to the support pillar of the facility.

A pair of rail frames (Figure 4C) were arranged along the longitudinal direction to be spaced to a certain degree in the width direction. A terminal rail was equipped with multiple cameras at the bottom and installed in the width direction such that that both ends were placed on the rail frame. The first driving part is provided on one side of the rail frame, and the terminal rail is moved longitudinally (Supplementary Video 1). The second driving part was provided on the other side of the rail frame, and the terminal rail was moved in a longitudinal and width-wise direction (Supplementary Video 2).

A total of five cameras were connected to the camera rail: two Raspberry Pi camera modules (V2, Raspberry Pi Foundation, Cambridge, UK) taking RGB images with a resolution of 8-megapixel, two Pi NOIR cameras (Raspberry Pi Infrared Camera Module 2 NOIR) modules, taking infrared images with a resolution of 8-megapixel, and one depth camera of RealSense (D405, Intel, CA, USA) (Figures 2B, C). The Pi NOIR camera modules were connected to infrared lighting devices (YR-030) to collect images during the night, because the plant factory turned off the light for eight hours



FIGURE 1
Narrow passages and the height of moving racks for cultivating work in the industrial plant factory.

every day from 22:00 to 06:00. IR light was controlled by the MOSFET module for automatic switching on and off at the exact time: switch-on at 22:00 and switch-off at 06:00. The linear motion guide was coated with harmless substances (food-grade lubricants, SuperLube) that did not affect the health of the crop and helped smooth the movement and waterproofing of the motion guide.

The image acquisition system was attached to the highest level of cultivating racks nondestructively using clamps (Figures 2A, B).

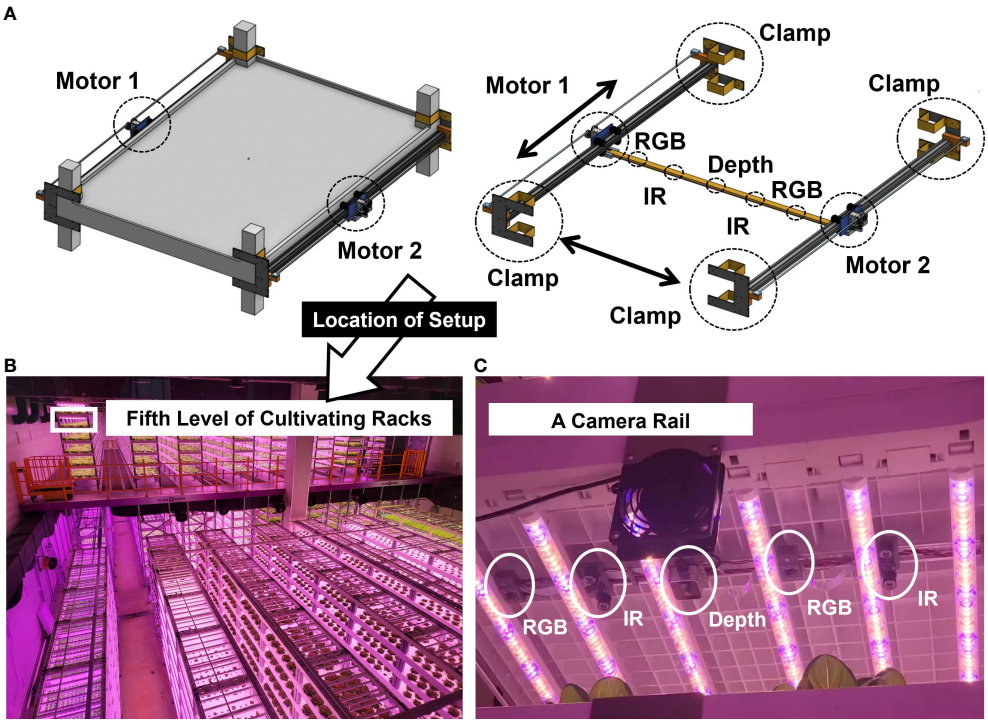


FIGURE 2
(A) A linear motion guide mounted on a cultivating rack and a camera rail with five different cameras (two RGBs, two IRs, and a depth) connected to the linear motion guide; **(B)** the location of the rack installed with the linear motion guide in the industrial plant factory; **(C)** setup of the connected camera rail with five different cameras to the linear motion guide.

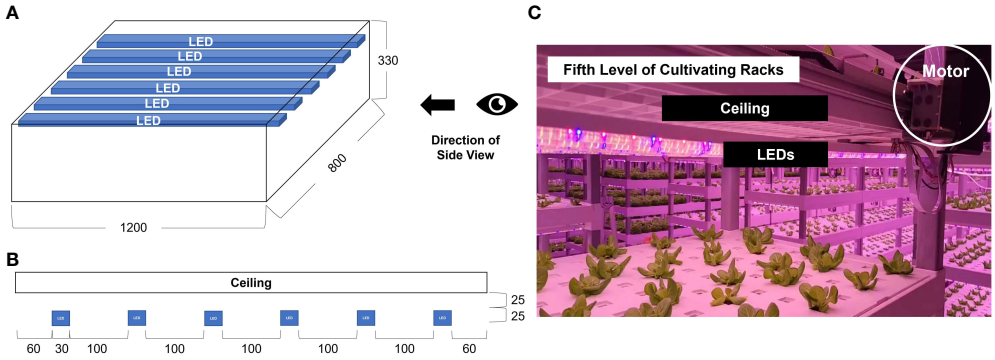


FIGURE 3
Sketch (unit: mm) of a confined cultivating bed used in the industrial plant factory; **(A)** A three-dimensional front view; **(B)** A side view showing the location of a camera rail, cameras, and LEDs.

Images were taken every hour, automatically moving in the horizontal and vertical directions (Figure 2C). For waterproofing, two parallel driving parts were covered in one case (Figures 4A, B). The control unit was attached to the front wall of driver. To specifically describe the control unit, Raspberry Pi 4 (Raspberry Pi Foundation, Cambridge, UK) connected to five cameras and a motor was covered with a case (Figures 4A, D). The timing belt and timing pulley were used to move vertically, and a rack and pinion frame were applied for horizontal movement (Figures 4B, C). A total of 2,040 aluminum profiles were used as the rail frames. A Robotis Dynamixel XM430-W350 actuators were used as a motorizing servo actuator and Robotis U2D2 was used as a Dynamixel for the USB communication module (Figure 4D).

A Secure Shell (SSH) was used to remotely control an image acquisition system that does not require a display for on-site applications. In addition, we used Tmux, a terminal multiplexer, to keep the system operating even if the SSH is disconnected due to problems in the plant factory. The image acquisition system saved images on Google Cloud and the local MCU in the PNG format with a pixel resolution of 3,280 × 2,464.

2.1.2 Manual fresh weight measurement

Images were collected for three months in a row for different stages of lettuce from 23 September 2022 to 10 January 2023, and 159 crops were destructively weighed 11 times (every 5 to 7 days) after removing the root (Table 1). Some images with overlapping or

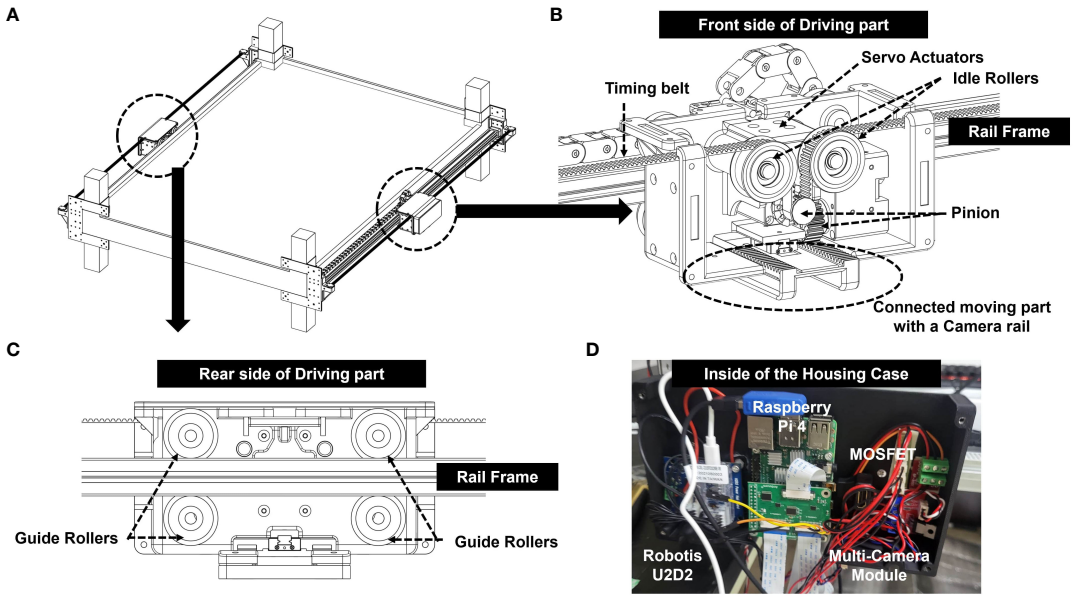


FIGURE 4
Overall framework and components of the Automatic Image Acquisition System; **(A)** A linear motion guide with two motors; **(B)** Front side of a driving part connected to a rail frame; **(C)** Rear side of a driving part connected to a rail frame; **(D)** Inside of the housing case attached to the outside surface of a driving part.

10.3389/fpls.2024.1365266

TABLE 1 The number of manually measured harvests and the number of corresponding images by date.

Date of Harvest	Days after transplanting (DAT)	Number of Weighed Harvest	Number of Images
30/09/2022	7	12	46
07/10/2022	15	12	45
12/10/2022	0	12	33
21/10/2022	9	12	25
28/10/2022	16	11	66
04/11/2022	23	12	36
04/11/2022	0	12	35
11/11/2022	7	9	9
25/11/2022	21	34	29
27/12/2022	0	9	9
03/01/2023	7	12	23
10/01/2023	14	12	20
Total	–	159	376

partially captured parts were not selected as datasets to develop the models. The number of chosen images matched to fresh weight was 376, as shown in Table 1. Images showing the entire part of the crop were selected and used repeatedly, and images showing only parts of the crop were not used because the data acquisition system can be designed to find ways to cover all shapes of each crop at the stage of detection in the future. The linear motion guide automatically moved and captured images at the same height, but at different angles, retaining the position of the top view per crop. Therefore, multiple images can be obtained from different angles for the same crop depending on the location of the cameras. For example, Figure 5 shows two examples of multiple images of the same plant taken from the location of the cameras. Two plants of 34 g (above images) and 2.4 g (below images) are shown as examples in Figure 5. This process produced varied unstructured data for each crop.

2.2 Image preprocessing and manual feature extraction

The collected and chosen images, free from overlapping or partial cut-off issues, were preprocessed using OpenCV 4 with Python to refine the datasets, remove complex backgrounds, and segment one target object from multiple objects in the raw data. All images were resized to 616 × 820 pixels to reduce computational resources. GrabCut, applied with the watershed method and median filter, was used to segment target objects from the background and remove unnoticed noise. This led to a lower failure at the subsequent stage of extracting contour features of the target objects from the images (Figure 6). From the contour features, we extracted the values of area (A), perimeter (P), and length of the major axis (MA) and minor axis (MI). A is the area inside the closed curve, and P is the length of the closed curve. MA and MI were extracted by fitting an ellipse; MA was the longest length, and MI was the shortest length of the fitted ellipse. All of these were used as input for fresh weight prediction models.

2.3 Fresh weight prediction models

There are two major methods for predicting the fresh weights of crops using computer vision: manual feature extraction (Mortensen et al., 2018) and automatic feature extraction (Lin et al., 2022; Moon et al., 2022a). In other words, manual feature extraction occurs when humans select and extract features manually to predict output

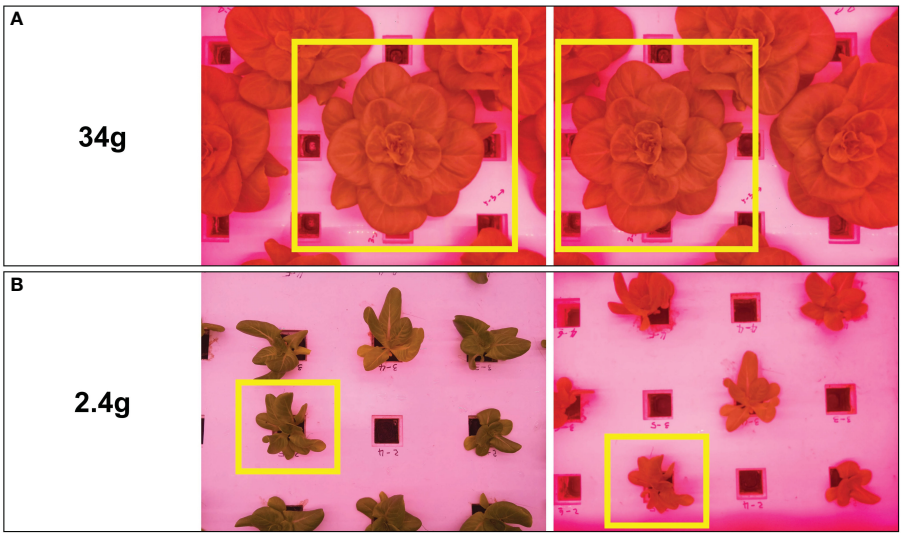


FIGURE 5
Two different images of the same crop from different locations of cameras with two examples; **(A)** a plant with a fresh weight of 34 g; **(B)** a plant with a fresh weight of 2.4 g.

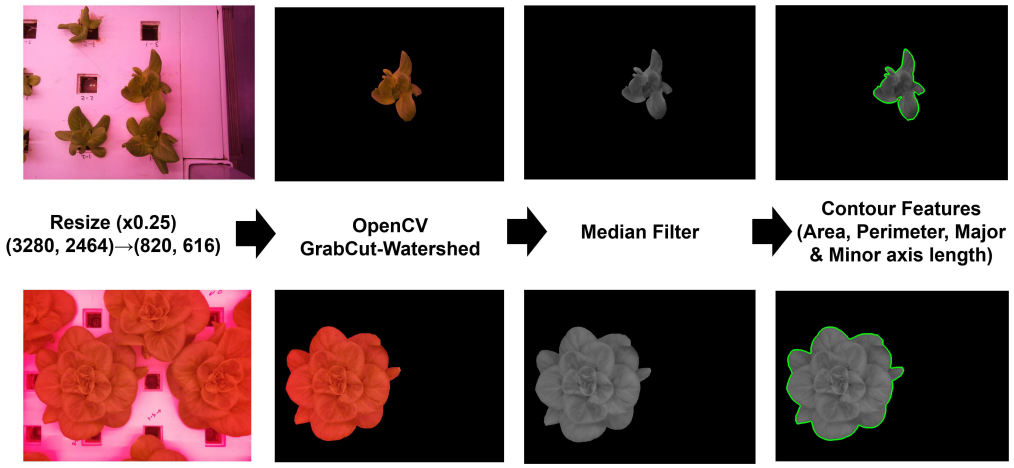


FIGURE 6
Image preprocessing workflow: Resize, image segmentation, median filtering, and feature extraction.

values. In contrast, automatic feature extraction occurs when computers select and extract features automatically.

The arranged data for developing the fresh weight estimation models were A, P, MA, MI, images, and fresh weight. The fresh weight values were the responsible (output) variables in the manual and automatic feature-extraction-based models. A, P, MA, and MI were used as explanatory (input) variables in the manual feature-extraction model-based models. For comparison, images were used as explanatory (input) variables in the automatic feature extraction model-based models.

2.3.1 Conventional linear regression models

2.3.1.1 Linear regression based on manual feature extraction

With four manually extracted variables (area, perimeter, major axis length, and minor axis length), linear regression with a single variable (Simple-LR, m=1 in Equation 2) and linear regression with multiple variables (Multi-LR) was performed (Equation 1). Multiple linear regression models were developed using a combination of two, three, and four independent variables. In total, four simple-LR models and 11 Multi-LR models were developed. All the regression models were developed in Keras.

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_p x_p + \epsilon \quad (1)$$

Where, $\hat{y}$ is the predicted value of the dependent variable, x_1 to x_p are p distinct predictor variables. β_0 represents $\hat{y}$ intercept, and ϵ is the residuals. β_1 to β_p denote the estimated regression coefficients.

2.3.1.2 Polynomial regression based on manual feature extraction

Polynomial regression is a form of linear regression that fits the nonlinear relationship between the dependent and independent variables. All polynomial regression models were conducted with a second degree (m = 2 in Equation 2) and third degree (m = 3 in Equation 2) polynomials to form quadratic and cubic expressions, respectively. Simple polynomial regression and multivariate polynomial regression were performed in the same way as above, with a combination of two, three, and four independent variables. In total, eight simple polynomial regression models and twenty-two multiple linear regression models were developed.

$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \ldots + \beta_m x^m + \epsilon \quad (2)$$

Where, $\hat{y}$ is the predicted value for the polynomial model with regression coefficients β_1 to β_m for each degree m and $\hat{y}$ intercept β_0 . It has m predictors raised to the power of I, where $i = 1$ until m. ϵ represents the model's error term.

2.3.2 Deep learning-based regression models using manual and automatic feature extraction

Several neural networks have been employed to recognize complex nonlinear functions better than traditional statistical regression models (Ong et al., 2008). The architecture of the neural networks is presented in Table 2.

For the four manually extracted variables as input, multilayer perceptron (MLP_1) had three fully connected linear layers (FC) with different numbers of nodes: 32, 15, and 1 each. The layers with 32 and 15 nodes were followed by the activation function of the rectified linear unit (ReLU). In a fully connected layer with 15 nodes, dropout was employed at a rate of 0.5.

MLP_2 and convolutional neural network (CNN) were applied to unstructured data for automatic feature extraction as regression models. CNN was adopted over other deep learning models because it excels in processing visual data due to its convolutional layers that effectively identify spatial hierarchies and patterns in images. Their architecture was specifically designed to handle the variability and complexity of image data, which is essential for accurate plant weight prediction. Additionally, CNN's ability to learn features directly from images without manual feature engineering makes them ideal for efficiently analyzing large datasets typically involved in our plant factory conditions. This high performance of CNN has been proven in many previous studies related to weight estimation models (Zhang et al., 2020; Gang et al., 2022a; Moon et al., 2022a).

TABLE 2 Architectures of deep learning models.

Model	MLP_1	MLP_2	CNN
Input variables	Structured data (A, P, MA, MI)	Unstructured data (Images)	Unstructured data (Images)
Input size	4	784	1 × 28 × 28
Layers	FC-32-ReLU	FC-1568-ReLU	Conv3–64-ReLU
	FC-15-ReLU	FC-392-ReLU	Conv3–128-ReLU
	Dropout(0.5)	Dropout(0.5)	MaxPool(2)
	FC-1-ReLU	FC-1-ReLU	Conv3–256-ReLU
			MaxPool(2)
			Conv3–512-ReLU
			Average Pool(7)
			FC-1
Output size	1 × 1		

FC represents a fully connected layer, a basic form of the neural network, and Conv is a convolution layer. Parameters for Conv are denoted as "{Type of layer}{Kernel size} – {Number of filters} – {Activation function}," and parameters for the other layers are denoted as "{type of layer} – {Number of nodes in the layer} – {Activation function}." MaxPool represents the maximum pooling.

The images were flattened as an input of MLP_2 such that the size of (1 × 28 × 28) was 784. MLP_2 had three fully connected linear layers (FC) with different numbers of nodes: 1,568, 392, and 1. The layers with 1,568 and 392 nodes were followed by an activation function of the rectified linear unit (ReLU). In a fully connected layer with 392 nodes, dropout was employed at a rate of 0.5.

Using Resnet-18, a skip connection is applied to avoid gradient vanishing (He et al., 2016) were used as the CNN model. The convolutional kernels were set to 3 × 3 with a stride of 1, and zero-padding was applied to maintain the same size of output with input size so that edge information could be used. Each layer contained different numbers of feature channels: 64, 128, 256, and 512. The ReLU follows each convolutional layer to ensure training stability. Additionally, maximum pooling was inserted twice after layer of 128-feature channels and 256-feature channels. After the layer of 512-feature channels, global average pool and flattening were performed, followed by a fully connected layer.

3 Experiments

3.1 Model training

All deep learning models were developed in Pytorch on the Windows Subsystem for Linux (WSL) with a CPU of i9–12900K, GPU of RTX3090 (NVIDIA, Santa Clara, CA, USA), and 64 GB memory.

In the case of automatic feature extraction, the dataset contained samples and labels, and it was split into a training set and a test set with a ratio of 7:3. Of the 376 datasets, 263 were used to develop the training set and 133 were used to develop the test set. In addition, a data loader was constructed with 512 batches to easily access the samples. All fresh weight values were adjusted in scale by multiplying by 0.01 for stability and convergence speed improvement. All input images were adjusted to (28, 28) in size with one channel and were randomly flipped horizontally and vertically.

In the case of manual feature extraction, scaling was performed using MinMax Scaler to ensure that all values existed in the range between 0 and 1 because all values of independent variables had considerable differences in scale. In addition, the Standard Scaler was performed on all response variable values to be standardized, as the distribution's standard deviation was equal to 1.

MLP_1, MLP_2, and CNN were conducted using the MSE loss function and Adam optimizer with a 0.001 learning rate as the regression models. In the case of MLP_2 and CNN, the number of iterations per epoch was one because the size of the training data was 263 and the batch size was set to 512.

3.2 Correlation analysis

The correlation is a statistical measure that expresses the strength of the relationship between two variables. If there are multiple variables, a correlation matrix, a table showing the correlation coefficients between a set of variables, is necessary to find the correlation between all variables. The Pearson's correlation coefficient (Equation 3), known as the correlation coefficient, is a statistical measure of the linear relationship between the two variables. Correlation heatmaps are essential for visualizing the strength of relationships between numerical variables through color coding of the cells. It also allows the identification of outliers and the detection of linear and nonlinear relationships. Correlations between variables were investigated using a heatmap to select the best-fit input variables for fresh weight prediction models.

$$\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad (3)$$

Where x and y represent two variables, y is the mean value of x, and $\bar{y}$ is the mean value of y. x_i and y_i represented different values of x and y.

Conventional regression models have various compositions of inputs, from one variable to four variables. All combinations of inputs were conducted for a total of 15 cases: (A), (P), (MA), (MI), (A, P), (A, MA), (A, MI), (P, MA), (P, MI), (MA, MI), (A, P, MA), (A, P, MI), (A, MA, MI), (P, MA, MI), and (A, P, MA, MI). MLP_1 was conducted with one set of inputs: (A, P, MA, and MI). MLP_2 and CNN received images as inputs. All models were designed to derive the value of the fresh weight as the output.

3.3 Model evaluation metrics

All the developed models were evaluated and compared in terms of model performance with values of the root mean square error (RMSE) and coefficient of determination (R^2), as shown in Equation 4

and plots, such as kernel density estimation (KDE), which is used to estimate the underlying probability density function of a dataset (Chen, 2017), allowing to explore the pattern of the data. In addition, the inference time per image (millisecond, ms) was measured because all models were developed for use in industrial plant factories, and the speed of weight estimation can be directly related to the efficiency of the on-site system.

$$\text{RMSE} = \frac{\sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\sqrt{\sum_{i=1}^{n}(y_i)^2}} \quad (4)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

where y represents the measured value, $\hat{y}$ is the predicted value by the models, $\bar{y}$ is the average value of the measured value, and n is the number of samples.

4 Results and discussion

4.1 Correlations between variables

Correlation coefficients were analyzed using a pair plot and heatmap to select the appropriate variables. As shown in Figure 7A, the relationships between the independent variables and weight are nonlinear. The area, major axis, and minor axis show a high correlation coefficient value to the weight, with a value over 0.85. Moreover, even the lowest correlation coefficient value to the fresh weight was 0.76 (Figure 7B). All variables can affect the accuracy of the models when used as input variables.

4.2 Performance evaluation and comparison of models

4.2.1 Conventional regression models based on manual feature extraction

The regression model results are shown as the R^2 and RMSE (g) values in Tables 3–5. The polynomial regression models performed better than the linear regression models. The Simple-LR (Table 3) showed the highest values of 0.77 as R^2 and 16.21 as RMSE (g) from one variable of A as input, and it had relatively poor performance compared to polynomial regression models with multivariable. Most of the third-degree models showed higher performance (0.90 of R^2 and 10.63 g of RMSE as best performed model with the combination of A and P, 0.81 of R^2 and 14.76 g of RMSE on average) than second-degree models (0.88 of R^2 and 11.87 g of RMSE as best performed model with the combination of A, P, and MA, 0.79 of R^2 and 15.39 g of RMSE on average). It tends to lower the accuracy of model performances when MI was included with other variables as input, except for one of the results in the Multi-LR models (Table 4), which showed the highest values of 0.84 (R^2) and 13.70 g (RMSE) from three variables: A, P, and MI. Therefore, even if MI is highly correlated with weight (0.87), it may provide redundant information already described in other variables to predict fresh weight and is not essential for fresh weight prediction. However, combinations with P, which showed the lowest correlation value (0.76) with other variables, improved the model performance in all the combinations of other variables, mostly over 0.80 of R^2, while other combinations without P showed poor model performance, mostly under 0.80 of R^2 and only P itself as input made the poorest performance, at around 0.6 of R^2. Therefore, combining P with other variables positively affected the fresh weight prediction by providing more information about crop conditions.

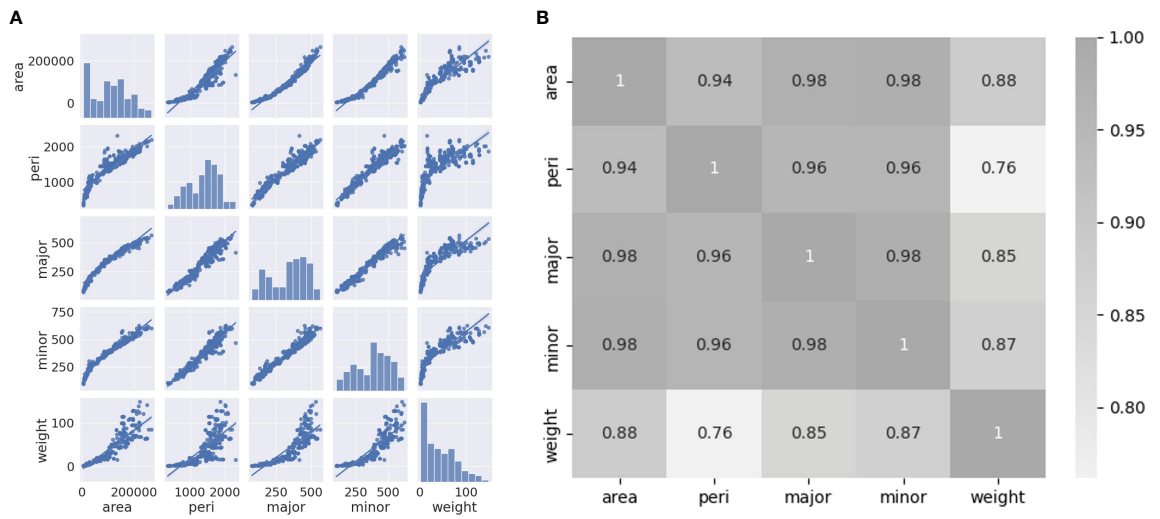


FIGURE 7
(A) Pair plot between variables (area, perimeter, major axis length, minor axis length, and weight); **(B)** Heatmap of Pearson's correlation coefficients between variables.

TABLE 3 Results of simple univariate linear and polynomial regression models based on manual feature extraction.

Variables / The degree of a polynomial	Area (A)		Perimeter (P)		Major Axis length (MA)		Minor axis length (MI)	
	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)
1 (Linear)	0.77	16.21	0.60	21.19	0.75	16.77	0.74	17.26
2	0.77	16.34	0.60	21.26	0.77	16.11	0.73	17.48
3	0.83	13.93	0.65	19.93	0.80	15.11	0.76	16.55

Overall, the best model in performance among conventional regression models was with the combination of A and P as input variables, resulting in 0.90 of R^2 and 10.63 g of RMSE in the third-degree polynomial regression model. In Simple-LR, A was the best performing variable and P was the poorest-performing variable for the models. However, the model with two variables (A and P) performed best among the conventional regression models. These results show that the variable showing a low correlation value with the target variable should not be excluded when developing models because it can still provide the necessary information for the prediction.

4.2.2 Deep learning regression models based on manual and automatic feature extraction

All the manually extracted features of A, P, MA, and MI were inserted into MLP_1 as input variables, and the R^2 value of the test set was 0.85 with an RMSE of 14.62 g (Table 5) at the epoch of 400 (Figure 8A). This was a better performance than simple linear

TABLE 4 Results of multivariate linear and polynomial regression models based on manual feature extraction.

Variables / The degree of a polynomial	(A, P)		(A, MA)		(A, MI)		(P, MA)	
	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)
1 (Linear)	0.80	14.86	0.76	16.61	0.77	16.20	0.80	14.87
2	0.86	12.56	0.79	15.53	0.79	15.61	0.83	14.01
3	0.90	10.63	0.80	15.23	0.74	17.24	0.86	12.55
Variables / The degree of a polynomial	**(P, MI)**		**(MA, MI)**		**(A, P, MA)**		**(A, P, MI)**	
	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)
1 (Linear)	0.80	15.04	0.73	17.62	0.82	14.16	0.84	13.70
2	0.79	15.66	0.69	18.70	0.88	11.87	0.86	12.65
3	0.83	13.97	0.76	16.65	0.88	11.47	0.85	13.30
Variables / The degree of a polynomial	**(A, MA, MI)**		**(P, MA, MI)**		**(A, P, MA, MI)**			
	R^2	RMSE (g)	R^2	RMSE (g)	R^2	RMSE (g)		
1 (Linear)	0.75	16.97	0.82	14.22	0.83	13.81		
2	0.76	16.66	0.84	13.46	0.85	12.90		
3	0.79	16.44	0.88	11.61	0.88	16.79		

TABLE 5 Results of deep learning models based on manual and automatic feature extraction.

Variables (Features) / Model	Manual (A, P, MA, MI)			Automatic from Images		
	R^2	RMSE (g)	Inference Time per Image (ms)	R^2	RMSE (g)	Inference Time per Image (ms)
MLP	0.85	14.62	0.002	0.93	9.35	0.003
CNN	–	–		0.95	8.06	0.026

regression models but lower than polynomial regression models, such as the model with the set of A and P as input (0.90 of R^2 and 10.63 g of RMSE). However, the CNN-applied model with unstructured data based on automatic feature extraction from the input images performed much better. The R^2 value of the test set was 0.95 with an RMSE of 8.06 g (Table 5) at the epoch of 300 (Figure 8C), and the ResNet18 model was used as the CNN architecture. This is a much better result than that of previous studies. The research using RGB image-based CNN_284 architecture showed a 0.92 of R^2 value (Xu et al., 2023). Moreover, it performed similarly to more intricate RGB-D images-based CNN models, which resulted in values of R^2 around 0.95 (Zhang et al., 2022; Gang et al., 2022a). MLP_2 with images performed relatively well, with an R^2 of 0.93 and RMSE of 9.35 g, although it showed slightly lower performance than the CNN model. The CNN model is the best, as it shows the highest accuracy of weight estimation. However, MLP_2 is more appropriate for the weight estimation system used in industrial plant factories because of the difference in inference time per image. The speed of the inference time per image is important because it is directly related to the efficiency of the on-site system. If it takes longer than a second to estimate the fresh weight of a plant, the company would be reluctant to use the system because of the slow inference. MLP_2 showed an approximately nine times faster inference time per image than CNN, with a slightly lower accuracy. Specifically, the inference time per image and the R^2 value of MLP_2 were 0.003 milliseconds (ms) and 0.93, respectively, while those of CNN showed 0.026 ms and 0.95 ms each. Because of the model performance, the input data type is essential for deep-learning models. Automatically extracted features from unstructured data can enhance the performance of deep learning models. In addition, light models that can be run with lower computing power devices, such as a microcontroller unit (MCU), must be used because industrial plant factories have the distinct characteristics of a confined space. Therefore, the on-site fresh-weight estimation model should be simpler. In this aspect, the generated models in this study were appropriate because only RGB and IR images were performed for model development and performed as well as any other complicated models in previous research.

In addition, the number of required epochs for Training MSE loss to converge was 300 in CNN (Figure 8C), smaller than MLP_1 (400) (Figure 8A) and MLP_2 (500) (Figure 8B). The convergence properties of MLP_1 were unstable at the end of the epochs, whereas MLP_2 and CNN exhibited stable convergence properties. Therefore, automatic feature extraction based on images reinforces models by recognizing more complicated interaction functions from the data.

We obtained probability density function graphs to estimate the characteristics of the probability distribution from fresh weight and to estimate the values that fresh weight can have and the possible degree of fresh weight on having that value (Figure 9). In the results of the KDE plots in Figure 9, MLP_1 showed the lowest similarity between true values of manually measured fresh weight (True) and predicted values of fresh weight (Prediction) over the entire range of fresh weight. Meanwhile, both MLP_2 and CNN showed a high similarity between the True and Prediction fresh weights. It is assumed that MLP_2 is more appropriate for use in the range of 0 g to 25 g and 60 g to 100 g than CNN, whereas CNN is more suitable for application in the range of 25 g to 60 g and over 100 g than MLP_2. Different models can be applied to more suitable ranges by considering the model performance. Moreover, it is expected that

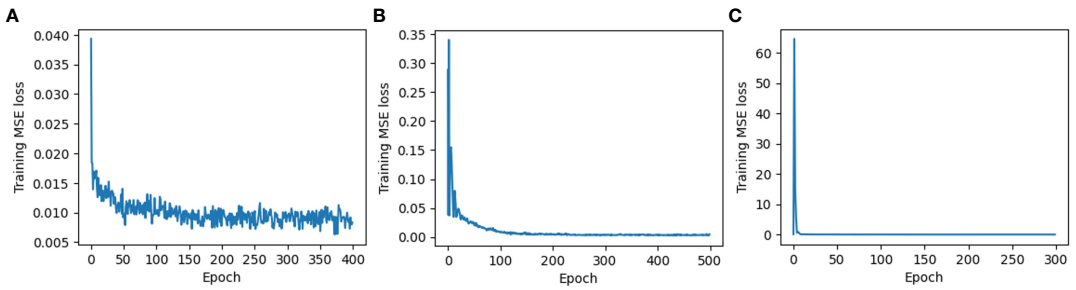


FIGURE 8
Training MSE loss: **(A)** Multilayer perceptron based on manual feature extraction (MLP_1); **(B)** Multilayer perceptron based on automatic feature extraction (MLP_2); and **(C)** Convolutional neural networks based on automatic feature extraction (CNN).

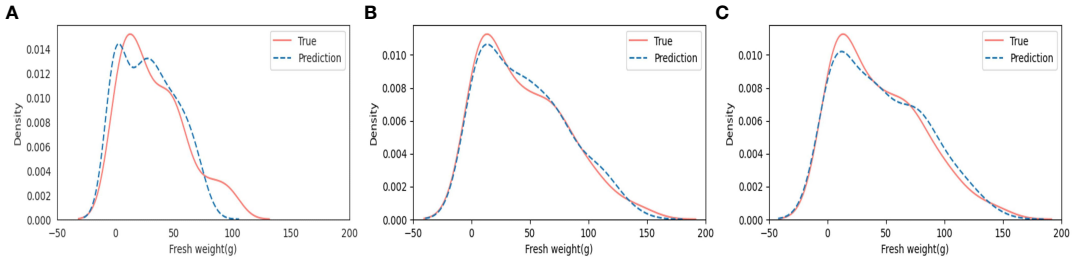


FIGURE 9
Results of Kernel density estimation **(KDE)** plots: **(A)** Multilayer perceptron based on manual feature extraction (MLP_1); **(B)** Multilayer perceptron based on automatic feature extraction (MLP_2); and **(C)** Convolutional neural networks based on automatic feature extraction (CNN).

all weak sections can be strengthened by adding more datasets during the stage of model training.

4.3 Feasibility of the data acquisition system in the industrial plant factory

The horizontally mobile data acquisition system in a narrow space is equipped with multiple cameras between a pair of rail frames and terminal rails spaced apart at regular intervals. As it moved in the width direction and along the rail frame in the longitudinal direction, the growth status of the crop was checked by adjusting the position of the camera. The first drive motor for longitudinal movement and the second drive motor for width direction movement were arranged at the same location. This made it possible to safely protect driving parts such as motors in high-temperature and high-humidity growing environments.

The first and second driving units operated in conjunction with each other, longitudinal movement was possible, and the position of the terminal rail could be automatically adjusted through the control box.

The timing belt of the first and second drive units was open type. Owing to its structure, there was no need to separate the first and second drive motors. The beneficial effect of improving the waterproofing effect was achieved by placing it in one waterproof case.

RGB and IR images were captured consistently and automatically in an industrial plant factory. Sometimes, the system instantly stopped when some debris fell into the gap of the rail frame because it caused the slip phenomenon of the motors, causing it to lose balance when moving in parallel, resulting in a stop in motion and not taking images. However, it operated well immediately after the rail frame was cleaned. Despite this issue, consistent images for developing models were adequately and automatically obtained from the system, resulting in good results from the developed models due to the high image quality. As mentioned in the *Materials and methods*, depth images were taken with a RealSense D405 depth sensor on the camera rail of the linear motion guide. However, three weeks later, the images were too corrupted for use in model development. The RealSense D405 depth sensor is assumed to be unsuitable in harsh, high-humidity environments because it operates well when brought back to laboratory conditions but not in the plant factory, even after the condition check. Apart from the depth images, all other images of RGB and IR were taken continuously.

In commercial plant factories where the experiment was conducted, crops were planted in zigzag positions and grown at an appropriate density to prevent the overlap of lettuce (Figure 10).

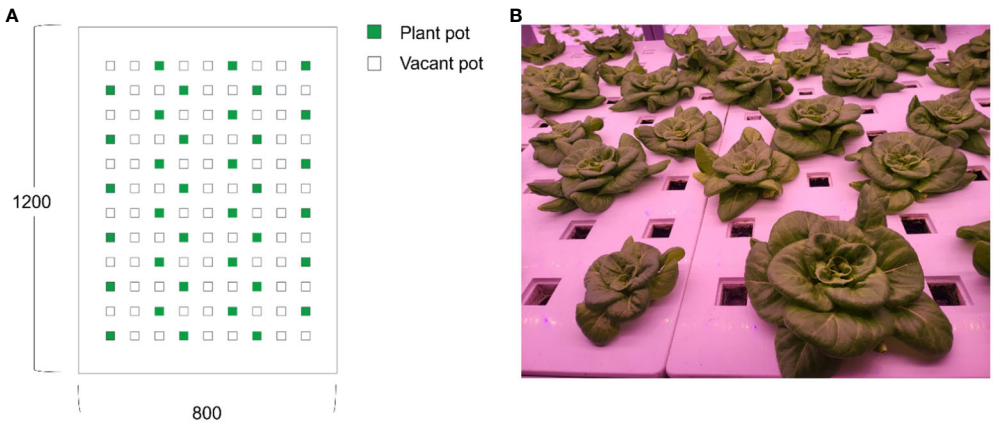


FIGURE 10
Positions of plant pots in the cultivating rack and grown crops: **(A)** zigzag positions of plants and vacant pots; **(B)** well-grown lettuce without overlapping problems.

A total of 36 crops were transplanted and cultivated in a 1,200 mm × 800 mm cultivation rack.

Except for the depth camera, it was feasible to operate in a commercialized plant factory with high humidity and a confined space to obtain data automatically and consistently. It would be more efficient if workers cleaned up the surrounding cultivating beds after touching crops to avoid falling debris in the gap of the aluminum profiles.

4.4 Future study

Several attempts have been made to improve this model. For example, it can be improved if the LSTM model is applied to develop time-series models that reflect the exact changes in plant growth on days. Integrating LSTMs can capture the sequential nuances of growth, offer a granular view of development cycles, and enable more targeted interventions. In addition, diverse types of images can be added for model development by collecting them with stable devices that can be utilized under harsh conditions. Enhancing the dataset with thermal or hyperspectral imaging could also unlock new correlations between visible symptoms and the plant's internal state, leading to more comprehensive growth data. For example, taking depth images with a stereo camera or taking images from the side view can further improve models because they can contain more information, such as the height and volume of plants and the number of leaves, including hidden leaves. These multidimensional data could facilitate a more sophisticated model that predicts weight and assesses plant health and maturity, thereby informing more accurate harvesting times. This information can be used as another variable in the model. Future iterations of the model could also leverage advancements in predictive algorithms to automate the detection of abnormal growth patterns, thereby offering early warnings for potential issues. In addition, the models can be developed into other forms of narrow environments, such as small-scale or home-based systems (Kim et al., 2022), and other varieties with different plant morphogenesis. In the case of diverse species, different morphogenesis can cause different vision-based model performances, with different values of RMSE and R2. Moon et al. performed a growth analysis of plant factory-grown lettuce such as Corbana, Caipira, and Fairy using deep neural networks based on automated feature extraction (Moon et al., 2022a). The result of the model adopting convolutional neural networks was 0.77 of R2. One reason for the low performance can be assumed to be the different morphologies of different crops.

5 Conclusion

In this study, we successfully developed a fresh weight prediction model for butterhead lettuce using computer vision that can also be used on-site in plant factories. This application represents the first development of a noninvasive weight estimation system based on automatic data acquisition, especially for commercialized plant factories with narrow, crowded, and high-humidity environments. Specifically, using a linear motion guide, the automatic data acquisition system is adequate for collecting consistent unstructured data onsite. This deployment of the automated system within a plant factory underscores the significant innovation in the data collection method in agricultural technology.

Automatic feature extraction with a convolutional neural network (CNN) based on images showed a high performance with an R2 of 0.95 and RMSE of 8.06 g compared to any other model for the fresh weight estimation of butterhead lettuce. However, MLP_2 can be more appropriate to be adopted on the spot in the industrial plant factory because the inference time per image was approximately nine times faster than CNN, with a slightly lower value of R2 (0.93) and a slightly higher value of RMSE (9.35 g). Therefore, the superior performance of MLP_2 introduces a breakthrough in precision agriculture, particularly in how data-driven models can be seamlessly integrated into operational workflows. The automatic feature extraction-based models using images as input through a multilayer perceptron (MLP) performed better than any other manual feature extraction-based models since the best performance was 0.90 of R2 and 10.63 g of RMSE from the third-degree of polynomial multivariable regression model with parameters of A and P. Therefore, the automatic feature extraction method using unstructured data is the most appropriate model for predicting fresh weight.

As an onsite automatic data acquisition system, the models should be light and available with lower computing power. The practical implementation of such efficient models in a commercial setting, without reliance on high-power computational resources, illustrates the feasibility and applicability of our approach. The model generated in this study uses automatic feature extraction with unstructured data, but it has a simple structure and shows sufficient performance compared to other models in previous studies. Simplifying complex data processing into a robust yet straightforward model that is accessible for on-site use is another innovative aspect of this study. The model can be applied to other fields such as small-scale home-based cultivation systems. In this study, workers involved in indoor farming were able to measure the fresh weight of crops in a non-destructive way and harvest at the appropriate time.

Data availability statement

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

Author contributions

J-SK: Conceptualization, Data curation, Formal Analysis, Investigation, Methodology, Project administration, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. SM: Writing – review & editing, Methodology, Software. JP: Methodology, Writing – review & editing. TK: Methodology, Software, Writing – review & editing. SC: Funding acquisition, Resources, Supervision, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the BK21 FOUR, Global Smart Farm Educational Research Center, Seoul National University, Seoul, Korea; and the New Faculty Startup Fund from Seoul National University (Project No. 500–20220186).

Acknowledgments

The authors also wish to thank PlanTfarm Corporation and Robowill Corporation in South Korea for assisting in the setup of the device and the process of experiments in an industrial plant factory on site.

Conflict of interest

SC reports financial support was provided by Seoul National University. SC, J-SK, and SM have patent Horizontally moving shooting platform in a small space pending to Seoul National University.

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/fpls.2024.1365266/full#supplementary-material

References

Adenäuer, M., Deuss, A., Elasri, A., Altendorf, S., René, S., Enciso, SRA., et al. (2023). *OECD-FAO Agricultural Outlook 2023–2032*. Paris: OECD Publishing. doi: 10.1787/08801ab7-en

Ben Hassen, T., and El Bilali, H. (2022). Impacts of the Russia-Ukraine war on global food security: towards more sustainable and resilient food systems? *Foods* 11, 1–17. doi: 10.3390/foods11152301

Boros, I. F., Székely, G., Balázs, L., Csambalik, L., and Sipos, L. (2023). Effects of LED lighting environments on lettuce (Lactuca sativa L.) in PFAL systems – A review. *Scientia Hortic.* 321, 112351. doi: 10.1016/j.scienta.2023.112351

Brain, R., Perkins, D., Ghebremichael, L., White, M., Goodwin, G., and Aerts, M. (2023). The shrinking land challenge. *Cite This: ACS Agric. Sci. Technol.* 2023, 152–157. doi: 10.1021/acsagscitech.2c00250

Chamara, N., Islam, M. D., Bai, G.(., Shi, Y., and Ge, Y. (2022). Ag-IoT for crop and environment monitoring: Past, present, and future. *Agric. Syst.* 203, 103497. doi: 10.1016/j.agsy.2022.103497

Chen, Y. C. (2017). A tutorial on kernel density estimation and recent advances. *Biostatistics Epidemiol.* 1, 161–187. doi: 10.1080/24709360.2017.1396742

Chen, W. T., Yeh, Y. H. F., Liu, T. Y., and Lin, T. (2016). An automated and continuous plant weight measurement system for plant factory. *Front. Plant Sci.* 7. doi: 10.3389/fpls.2016.00392

Farcas, A. C., Galanakis, C. M., Socaciu, C., Pop, O. L., Tibulca, D., Paucean, A., et al. (2020). Food security during the pandemic and the importance of the bioeconomy in the new era. *Sustainability* 13 (1), 150. doi: 10.3390/su13010150

Gang, M. S., Kim, H. J., and Kim, D. W. (2022a). Estimation of greenhouse lettuce growth indices based on a two-stage CNN using RGB-D images. *Sensors* 22, 5499. doi: 10.3390/s22155499

Hati, A. J., and Singh, R. R. (2021). Smart indoor farms: leveraging technological advancements to power a sustainable agricultural revolution. *AgriEngineering* 4), 728–767. doi: 10.3390/agriengineering3040047

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. Available online at: http://image-net.org/challenges/LSVRC/2015/.

Jiang, J.s., Kim, H. J., and Cho, W. J. (2018). On-the-go image processing system for spatial mapping of lettuce fresh weight in plant factory. *IFAC-PapersOnLine* 51, 130–134. doi: 10.1016/j.ifacol.2018.08.075

Kim, J. S. G., Jeong, W., Park, S., and Yang, M. (2022). Development of a fuzzy logic-controlled system for home cultivation of sweet basil. *Front Plant Sci.* 13, 999106. doi: 10.3389/fpls.2022.999106

Kozai, T., Niu, G., and Takagaki, M. (2019) Plant factory: an indoor vertical farming system for efficient quality food production. Available online at: https://books.google.com/books?hl=en&lr=&id=z-C7DwAAQBAJ&oi=fnd&pg=PP1&ots=zDtmsLhkfp&sig=4I5QCwwMEzkE-Rp9Se_-GEAb0oM.

Kozai, T., and Sasaki, S. (2013). Resource use efficiency of closed plant production system with artificial light: Concept, estimation and application to plant factory. *Proc. Japan Academy Ser. B* 89, 447–461. doi: 10.2183/pjab.89.447

Lee, J. W. (2008). MACHINE VISION MONITORING SYSTEM OF LETTUCE GROWTH IN A STATE-OF-THE-ART GREENHOUSE *. *Modern Phys. Lett. B* 22, 953–958. doi: 10.1142/S0217984908015668

Lin, Z., Fu, R., Ren, G., Zhong, R., Ying, Y., and Lin, T. (2022). Automatic monitoring of lettuce fresh weight by multi-modal fusion based deep learning. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.980581

Lou, M., Lu, J., Wang, L., Jiang, H., and Zhou, M. (2022). Growth parameter acquisition and geometric point cloud completion of lettuce. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.947690

Mitchell, C. A. (2022). History of controlled environment horticulture: indoor farming and its key technologies. *HortScience* 57, 247–256. doi: 10.21273/HORTSCI16159-21

Mokhtar, A., El-Ssawy, W., He, H., Al-Anasari, N., Sammen, S. S., Gyasi-Agyei, Y., et al. (2022). Using machine learning models to predict hydroponically grown lettuce yield. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.706042

Moon, T., Choi, W. J., Jang, S. H., Choi, D. S., and Oh, M. M. (2022a). Growth analysis of plant factory-grown lettuce by deep neural networks based on automated feature extraction. *Horticulturae* 8 (12), 1124. doi: 10.3390/horticulturae8121124

Moon, T., Park, J., and Son, J. E. (2020). Estimation of sweet pepper crop fresh weight with convolutional neural network. *Protected Horticulture Plant Factory* 29, 381–387. doi: 10.12791/KSBEC.2020.29.4.381

Mortensen, A. K., Bender, A., Whelan, B., Barbour, M. M., Sukkarieh, S., Karstoft, H., et al. (2018). Segmentation of lettuce in coloured 3D point clouds for fresh weight estimation. *Comput. Electron. Agric.* 154, 373–381. doi: 10.1016/j.compag.2018.09.010

Ojo, M. O., and Zahid, A. (2023). *Non-Destructive Biomass Estimation for Hydroponic Lettuce Production*. ASABE Paper No. 2300776. St. Joseph, MI.: ASABE. 1–10. doi: 10.13031/aim.202300776

Ong, H. C., Leong, C. H., and Tai, S. H. (2008). A functional approximation comparison between neural networks and polynomial regression. *WSEAS Trans. Mathematics* 7, 353–363.

Petropoulou, A. S., van Marrewijk, B., de Zwart, F., Elings, A., Bijlaard, M., van Daalen, T., et al. (2023). Lettuce production in intelligent greenhouses—3D imaging and computer vision for plant spacing decisions. *Sensors* 23 (2), 2929. doi: 10.3390/s23062929

Reyes-Yanes, A., Martinez, P., and Ahmad, R. (2020). Real-time growth rate and fresh weight estimation for little gem romaine lettuce in aquaponic grow beds. *Comput. Electron. Agric.* 179, 105827. doi: 10.1016/j.compag.2020.105827

Tong, Y., Oh, M. M., and Fang, W. (2023). Editorial: Advanced technologies for energy saving, plant quality control and mechanization development in plant factory. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1193158

World Population Prospects - Population Division - United Nations2022. Available online at: https://population.un.org/wpp/Graphs/.

Xu, D., Chen, J., Li, B., and Ma, J. (2023). Improving lettuce fresh weight estimation accuracy through RGB-D fusion. *Agronomy* 13 (10), 2617. doi: 10.3390/agronomy13102617

Zhang, R., Liu, T., and Ma, J. (2017). Plant factory: A new method for reducing carbon emissions. in *AIP Conference Proceedings* (AIP Publishing). 18201. doi: 10.1063/1.4977288

Zhang, L., Xu, Z., Xu, D., Ma, J., Chen, Y., and Fu, Z. (2020). Growth monitoring of greenhouse lettuce based on a convolutional neural network. *Horticulture Res.* 7, 124. doi: 10.1038/s41438-020-00345-6

Zhang, Q., Zhang, X., Wu, Y., and Li, X. (2022). TMSCNet: A three-stage multi-branch self-correcting trait estimation network for RGB and depth images of lettuce. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.982562

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 28 June 2024
DOI 10.3389/fpls.2024.1368885

Check for updates

OPEN ACCESS

EDITED BY
Hans Beeckman,
Royal Museum for Central Africa, Belgium

REVIEWED BY
Huahong Huang,
Zhejiang Agriculture and Forestry University, China
Alexandre Gontijo,
Brazilia Forest Service, Brazil
Maaike De Ridder,
Royal Museum for Central Africa, Belgium

*CORRESPONDENCE
Tuo He
tuohe@caf.ac.cn

RECEIVED 11 January 2024
ACCEPTED 10 June 2024
PUBLISHED 28 June 2024

CITATION
Liu S, Zheng C, Wang J, Lu Y, Yao J, Zou Z, Yin Y and He T (2024) How to discriminate wood of CITES-listed tree species from their look-alikes: using an attention mechanism with the ResNet model on an enhanced macroscopic image dataset.
Front. Plant Sci. 15:1368885.
doi: 10.3389/fpls.2024.1368885

COPYRIGHT
© 2024 Liu, Zheng, Wang, Lu, Yao, Zou, Yin and He. 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.

How to discriminate wood of CITES-listed tree species from their look-alikes: using an attention mechanism with the ResNet model on an enhanced macroscopic image dataset

Shoujia Liu[1,2], Chang Zheng[1,2], Jiajun Wang[1,2,3], Yang Lu[1,2], Jie Yao[4], Zhiyuan Zou[4], Yafang Yin[1,2] and Tuo He[1,2,5]*

[1]Department of Wood Anatomy and Utilization, Research Institute of Wood Industry, Chinese Academy of Forestry, Beijing, China, [2]Wood Collections, Chinese Academy of Forestry, Beijing, China, [3]National Centre for Archaeology, Beijing, China, [4]Beijing Information Science and Technology University, Beijing, China, [5]Wildlife Conservation Monitoring Center, National Forestry and Grassland Administration, Beijing, China

Introduction: Global illegal trade in timbers is a major cause of the loss of tree species diversity. The Convention on International Trade in Endangered Species of Wild Fauna and Flora (CITES) has been developed to combat the illegal international timber trade. Its implementation relies on accurate wood identification techniques for field screening. However, meeting the demand for timber field screening at the species level using the traditional wood identification method depending on wood anatomy is complicated, time-consuming, and challenging for enforcement officials who did not major in wood science.

Methods: This study constructed a CITES-28 macroscopic image dataset, including 9,437 original images of 279 xylarium wood specimens from 14 CITES-listed commonly traded tree species and 14 look-alike species. We evaluated a suitable wood image preprocessing method and developed a highly effective computer vision classification model, SE-ResNet, on the enhanced image dataset. The model incorporated attention mechanism modules [squeeze-and-excitation networks (SENet)] into a convolutional neural network (ResNet) to identify 28 wood species.

Results: The results showed that the SE-ResNet model achieved a remarkable 99.65% accuracy. Additionally, image cropping and rotation were proven effective image preprocessing methods for data enhancement. This study also conducted real-world identification using images of new specimens from the timber market to test the model and achieved 82.3% accuracy.

Conclusion: This study presents a convolutional neural network model coupled with the SENet module to discriminate CITES-listed species with their look-alikes and investigates a standard guideline for enhancing wood transverse image data, providing a practical computer vision method tool to protect endangered tree species and highlighting its substantial potential for CITES implementation.

KEYWORDS

wood identification, CITES, convolutional neural network, attention mechanism, data enhancement, macroscopic images

1 Introduction

As an integral ecosystem component, trees assume a pivotal role in purifying our atmosphere, serving as habitats for diverse organisms and actively combating climate change (Rivers, 2017). In the intricate fabric of Earth's ecological system, they are indispensable entities, ensuring the vitality and equilibrium of our planet's ecological balance. When a tree is harvested, the wood can be used in every stage of human society, such as papermaking, construction, and furniture manufacturing (Falk, 2009; Latib et al., 2020). Approximately 73,000 tree species exist worldwide, but excessive logging and timber over-exploitation have resulted in up to 30% of the world's tree species being at risk of extinction, along with biodiversity destruction, soil erosion, and other ecological problems (Brancalion et al., 2018; Cazzolla et al., 2022; IUCN, 2022). This threat is true for some tropical tree species, which are widely used in producing high-value furniture, musical instruments, and handicrafts due to their excellent physical processing properties and beautiful patterns (Wick, 2019; Atikah et al., 2021). Thus, driven by high profits, some tree species are being over-harvested, which has been recorded as extinct/extinct in the wild, i.e., *Lachanodes arborea* (Lambdon and Ellick, 2016). Benefits from illegal logging are estimated to account for 15%–30% of the international timber trade, accounting for $51 to $152 billion a year (INTERPOL, 2021).

In recent decades, the world has witnessed a distressing decline in global tree populations. To prevent international trade from threatening the survival of endangered wildlife, the Convention on International Trade in Endangered Species of Wild Fauna and Flora (CITES) came into effect on 1 July 1975 to ensure that international trade in wild animal and plant specimens does not threaten species survival by subjecting international trade in selected species to certain controls (Appendices I, II, and III), according to their needed degree of protection (Goldsmith, 1978). In November 2022, the 19th Conference of the Parties to the CITES was held in Panama, and numerous tropical tree species were newly listed in CITES appendices. To date, more than 34,310 plant species of 134 genera have been included in the CITES appendices, including approximately 670 tree species, and 80% of these species are internationally traded for their timbers (CITES, 2023). Thus, a fast and accurate wood identification method is needed to support CITES implementation and promote legal logging.

Wood, or secondary xylem, is composed of countless cells of different shapes, sizes, and arrangements, with a complex and anisotropic structure (Shmulsky and Jones, 2019). Species within the same genus has similar appearance and even wood anatomical structure. Traditional wood identification, the most mainstream recognition approach, relies on human examination of the anatomical features of wood samples and refers to an identification standard list of the macroscopic and microscopic characters compiled by the International Association of Wood Anatomists (IAWA) (Committee, 1989; Yin et al., 2022). This task can only be completed accurately by experienced wood anatomists who familiarize with wood anatomy and identification, with the help of identification tools and reference materials. From a general point of view, it is arduous for the traditional wood identification method to reach species-level discrimination by observing anatomical features. To compensate for the shortcomings of traditional wood identification methods and break through the wood identification bottleneck at the species level, some techniques, i.e., DNA barcoding (Jiao et al., 2020), mass spectrometry (Deklerck et al., 2019; Price et al., 2021), and near-infrared spectroscopy (Bergo et al., 2016; Pan et al., 2021), have been developed. However, the lack of reference data and the high cost to establish it have limited applications of these approaches.

Computer vision is an interdisciplinary field at the intersection of computer science and image processing that aims to bridge the gap between human visual perception and machines by endowing computers with the capacity to understand, interpret, and extract knowledge from digital visual data (Liu et al., 2017). Image classification is a fundamental computer vision task that attempts to comprehend an entire image to classify images by assigning them to a specific label. With the rapid development of computer vision research and computer hardware performance, several neural network architectures have been proposed for image classification (Wang et al., 2019). Advancements in deep learning and convolutional neural networks (CNNs) have enabled more accurate and robust object and pattern identification with visual data. It is also very appealing to many wood anatomists and has been widely used in wood classification. The automated wood

identification method combines deep learning and computer vision to extract structural features and detect key information hidden in wood images (Voulodimos et al., 2019; Hwang and Sugiyama, 2021).

CNNs are the most commonly used computer vision-based wood identification models. As research continues, an array of neural network architectures, including distinguished models such as LeNet, AlexNet, ResNet, and GoogLeNet, have been introduced to address the wood image classification task (Kwon et al., 2017; Ravindran et al., 2018; Oktaria et al., 2019). The performance of these convolutional architectures is boosted by increasing their depth while maintaining their gradient information. In several studies, ResNet has shown superior classification performance compared to other models (He et al., 2021; Wu et al., 2021). However, a wood image is a fine-grained texture image with the characteristics of large intraclass variation and small interclass variation. To accurately determine the wood species, spatial and channel information should be exploited more delicately. Recently, the squeeze-and-excitation network (SENet) was proposed to provide the unit with a mechanism to explicitly model dynamic, nonlinear dependencies between channels using global information (Hu et al., 2018). This mechanism enables the network to selectively amplify or suppress specific feature maps, improving model performance. In this way, SENet can be embedded in the CNN, and the model can be trained to achieve better results.

In many cases, obtaining wood images from reliable sources is difficult, especially for globally regulated tree species, resulting in an unsatisfactory number of high-quality images to satisfy the modeling requirements. Therefore, in the case of limited wood image datasets, transfer learning and data augmentation are effective methods for solving the problem of insufficient data volume (Dyk and Meng, 2001; Ravindran et al., 2018). Kırbaş and Çifci (2022) compared the impacts of several deep learning architectures, namely, ResNet-50, Inception V3, Xception, and VGG19, based on the WOOD-AUTH dataset. They found that Xception performs remarkably well in the transfer learning domain. Hengshuo et al. proposed a wood identification algorithm based on an improved residual CNN, which augments the data based on the self-similarity of wood cross-sectional macrostructure and uses an improved residual CNN model, i.e., ResNet101, based on block gradient weighting to extract the features of each sub-image (Su et al., 2021).

Although there have been many wood species identification studies based on computer vision (Ravindran et al., 2018; Ravindran and Wiedenhoeft, 2020), two main factors still limit the development and application of this technique: the self-collected wood image data from different sources before use have not been adequately processed, and the identification accuracy is mainly affected by the sensitivity of the model to the slight variability in the wood species. Room for improvement in the models used in the existing studies remains.

This study aimed to develop a fast and reliable computer vision-based deep learning model by exploiting spatial and channel information to discriminate CITES-listed tree species from their look-alikes. The specific aims of this study were to (1) construct a CITES-28 (14 commonly traded CITES-listed tree species and 14 of their look-alikes) wood transverse surface image dataset, while concurrently investigating the optimal image data processing approach from the perspective of data enhancement; (2) establish a state-of-the-art SE-ResNet model by embedding the SENet module in a CNN (ResNet); and (3) discriminate CITES-listed tree species from their look-alikes using SE-ResNet in real-world identification.

2 Materials and methods

2.1 Data preparation and augmentation

In this section, we first explain how to prepare and enhance the wood image data in the experimental preparation stage from the perspective of data enhancement. Second, based on the enhanced data, we select the currently commonly used CNN for model training. Finally, the model performance is evaluated for tree species identification.

2.1.1 Image dataset collection

In this study, 279 verified specimens of 14 commercially important CITES-listed species and 14 of their look-alike species that were often mixed with CITES-listed species in trade were collected from the Wood Collection of the Chinese Academy of Forestry (CAFw), the USDA Forest Products Laboratory Wood Collection of Madison (MADw), and the Samuel J. Record Collection (SJRw). The transverse surfaces of the wood samples were sanded at grits of 180, 240, 400, 800, and 1,000 to obtain a clear surface for image acquisition. Their macroscopic transverse images are shown in Figure 1. The nonoverlapping images of 2,048 × 2,048 pixels, representing 6.35 × 6.35 mm of tissue, were taken with a XyloTron (Ravindran et al., 2020). A total of 9,437 original images were collected to build a CITES-28 dataset of 14 CITES-listed tree species and 14 of their look-alikes based on previous studies (Table 1) (Ravindran et al., 2018; Yin et al., 2022).

2.1.2 Dataset partitioning and patch dataset creation

We divided the dataset before creating patches to avoid dividing the same image into both the training and testing sets. The CITES-28 dataset of the original 9,437 images was divided into 80%/10%/10% training/validation/test splits at the image level. To ensure that the errors are representative of the entire dataset, 10-fold cross-validation for each model was used and accuracy is reported as the average over the 10 folds. An image dataset with a high imbalance results in poor classification performance; hence, six kinds of patches were extracted from the CITES-28 dataset images in this study. More overlap exists in patches of classes (species) with low quantities of images to maintain a balanced distribution of classes (species) in our dataset. The details of the patch dataset used for training and testing are listed in Table 2.

2.1.3 Image turning

We constructed a specific data augmentation method to fully train the model according to the data characteristics, as shown in Equation 1.

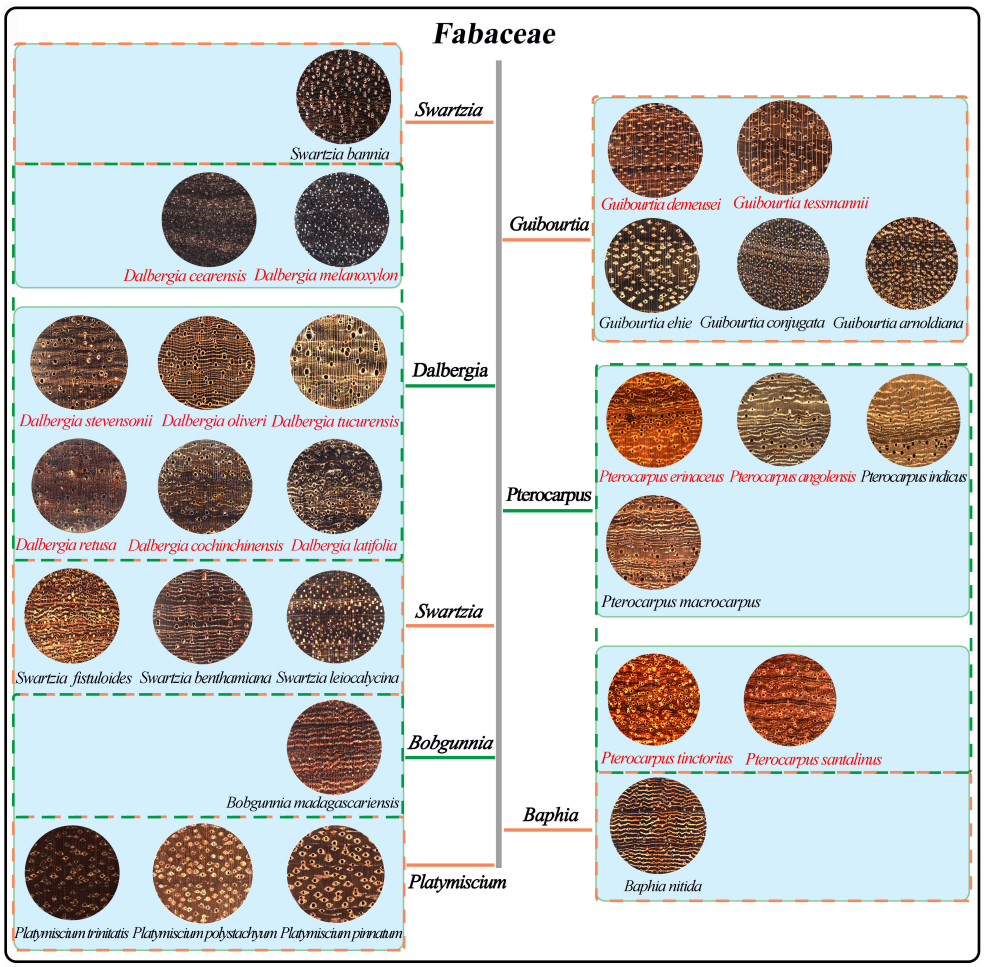


FIGURE 1
The transverse section of wood species in this research. The CITES-listed wood species and their look-alikes in the same blue frame are often confused (species in red text are CITES-listed). Species in the same dashed box are in the same genus.

TABLE 1 Detailed information of the CITES-28 image dataset.

Label	Species (class)	Protection level	Number of collected images	Number of xylarium specimens	Number of images/ xylarium specimens
0	*Dalbergia cearensis*	CITES II	321	14	23
1	*Dalbergia cochinchinensis*	CITES II	263	7	38
2	*Dalbergia latifolia*	CITES II	762	21	36
3	*Dalbergia melanoxylon*	CITES II	313	11	28
4	*Dalbergia oliveri*	CITES II	414	6	69
5	*Dalbergia retusa*	CITES II	698	16	44
6	*Dalbergia stevensonii*	CITES II	498	11	45
7	*Dalbergia tucurensis*	CITES II	461	12	38
8	*Platymiscium pinnatum*	/	201	10	20

(Continued)

TABLE 1 Continued

Label	Species (class)	Protection level	Number of collected images	Number of xylarium specimens	Number of images/ xylarium specimens
9	*Platymiscium polystachyum*	/	193	16	12
10	*Platymiscium trinitatis*	/	134	6	22
11	*Swartzia bannia*	/	122	5	24
12	*Swartzia benthamiana*	/	106	6	18
13	*Swartzia fistuloides*	/	111	4	28
14	*Swartzia leiocalycina*	/	115	7	16
15	*Bobgunnia madgascariensis*	/	110	5	22
16	*Pterocarpus erinaceus*	CITES II	373	9	41
17	*Pterocarpus santalinus*	CITES II	196	4	49
18	*Pterocarpus tinctorius*	CITES II	263	6	44
19	*Pterocarpus angolensis*	CITES II	805	19	42
20	*Pterocarpus indicus*	/	1,163	29	40
21	*Pterocarpus macrocarpus*	/	600	18	33
22	*Baphia nitida*	/	123	5	25
23	*Guibourtia demeusei*	CITES II	154	4	39
24	*Guibourtia tessmannii*	CITES II	249	7	36
25	*Guibourtia arnoldiana*	/	294	9	33
26	*Guibourtia ehie*	/	337	9	37
27	*Guibourtia conjugata*	/	58	3	19
	Total		**9,437**	**279**	~34

The meaning of the symbol "/" represents the species are not protected by CITES.
The bold values represents the total for each column.

TABLE 2 Prediction accuracies of two established models.

Model	Rotation mode	Image compensation	Accuracy (%)
SE-ResNet	Fixed	Y	**99.65**
	Fixed	N	99.55
	Random	Y	99.63
	Random	N	99.47
ResNet	Fixed	Y	**99.45**
	Fixed	N	99.41
	Random	Y	99.28
	Random	N	99.25

Y represents use 0 as a compensation item to complete the processed image.
N denotes that the rotated image is not processed.
The bold values represent the highest accuracy of SE-ResNet and ResNet.

$$aug(xi) = R(xi) + C \quad (1)$$

where *aug* is the enhancement method to be passed by the sample input to the model training. *R* is the image operation method for wood pictures proposed in this paper. During operation, image samples smaller than the input requirements of the model may be generated; thus, this paper introduces compensation factor *C* to compensate for the phenomenon of missing samples caused by the data enhancement operation.

In the actual test application process, controlling the acquisition angle of the test equipment and other factors is difficult. Therefore, we design a rule-based data enhancement rule with rotation as the core when building the data enhancement method. In addition, considering that the test angle accepted by the test equipment cannot be guaranteed in the actual test process, we propose a random rotation strategy to expand the diversity of training data further. The data enhancement operation can be represented by

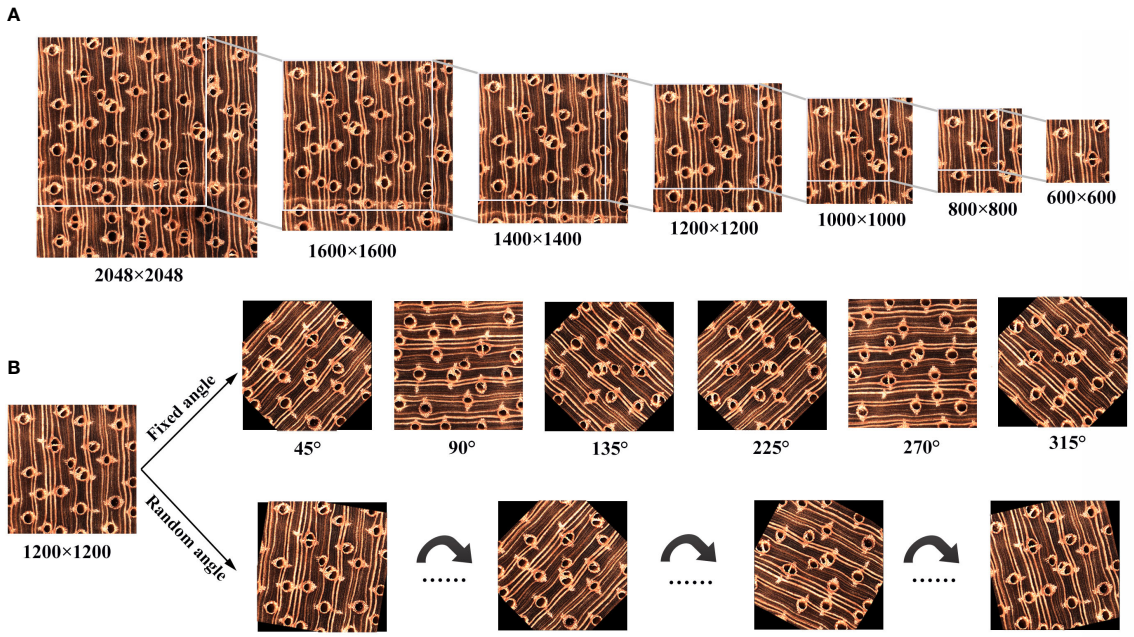


FIGURE 2
Data augmentation process. **(A)** Patch creation diagram. **(B)** Image rotation and compensation.

Equation 2.

$$R = R_{random} \cap R_{rule} \tag{2}$$

where R_{random} represents the random rotation strategy and R_{rule} represents the rule rotation strategy.

The data enhancement process is shown in Figure 2. When we enhanced the data based on fixed rotation, the image was rotated to the center with rotation angles of {45°, 90°, 135°, 225°, 270°, 315°}. When we enhanced the data based on a random rotation strategy, the image was rotated to the center, and the rotation angle was randomly sampled between 0° and 360°. After obtaining the rotated image, to meet the requirements that the model can input only the square image region, we cut the rotated image. We used 0 (black) as a compensation item to make up the exact area.

2.2 Architecture and training of convolutional neural networks

2.2.1 Convolutional neural network architecture

Based on the enhanced data, we selected ResNet, a widely used network, as the backbone. As He et al. (2016) discovered, a multilayer deep neural network can produce unexpected results, and the training accuracy drops as the layers increase, which is technically known as vanishing gradients. To address this problem, ResNet was proposed to help build a deeper neural network by utilizing skip connections or shortcuts to jump over some layers. SENet, as a classical attention mechanism, can be embedded in the CNN. In this study, we selected SE-ResNet to carry out our experiments.

To better obtain the feature expression of wood images in neural networks, more detailed analysis and processing were carried out on the features among image channels and the depth extraction of two-dimensional features. SENet provides the unit with a mechanism to explicitly model dynamic, nonlinear dependencies between channels using global information (Hu et al., 2018). SENet can ease the learning process and significantly enhance the representational power of the network. The SE-ResNet module includes a block, global average pooling, a fully connected (FC) layer, and an activation function layer with ReLU and sigmoid. The channel attention module is added after the residual module. The schematic architecture of the SE-ResNet model is shown in Figure 3.

Equation 3 shows that SENet mines and analyzes the weight information of different channels by studying the relationship between channels. First, it squeezes global spatial information into a channel descriptor by using global average pooling to generate channelwise statistics, which is ρ in Equation 3. Then, it learns to recalibrate the feature adaptively through two FC layers and uses two activation functions to learn nonmutually exclusive relationships. After obtaining the scalars, channelwise multiplication between the scalar and the feature map is carried out to obtain the final output.

$$s = \sigma(W_2\delta(W_1\rho(x))) \tag{3}$$

2.2.2 Optimization objective

We complete the forward pass and update the network by backpropagation. Globally, the loss function over iterations is still minimized until the loss converges, as shown in Equation 4, where θ

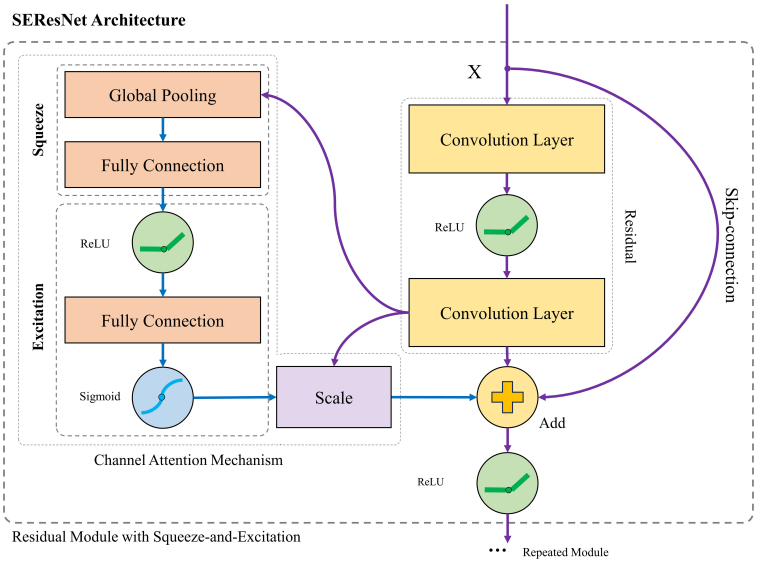


FIGURE 3
The schematic of the CNN architecture comprises a ResNet with an SENet employed for wood identification.

is the model parameter, X is the model input, and Y is the corresponding label.

Given an augmented selection, the model is optimized using perturbed features. The feature extraction ability is enhanced during the training process to obtain outstanding performance. In this sense, the augmentations help the model acquire more knowledge of potential input samples, increasing classification performance for real scenes.

$$\widehat{\theta} = argminL_{\theta}(\theta|aug(X), Y) \tag{4}$$

2.3 Model training

Model training is carried out in two phases. In the first phase, the model pretrained on the ImageNet dataset is studied for image classification via transfer learning. In the second phase, the model is trained to build classifiers for wood identification with the training set and tested at every epoch with the validation set. The initial learning rate of the model is 0.001, the momentum value is set to 0.9, and 24 epochs are trained sequentially. The epoch with the best result in the test set is taken to save the model. Stochastic gradient descent (SGD) is used to optimize the model. The model trained with the following hardware specifications: CPU Intel Core i9-14900K 6.0 GHz, 24 GB of RAM NVIDIA RTX 4090, and 96 GB of GPU. The experimental software environment is Ubuntu 20.04, Python 3.7.13, PyTorch 1.12.1, and NVIDIA CUDA 10.2.

In this study, the following process investigated the optimal image processing method (image rotation mode and patch size) for model accuracy. First, the model was trained with six different image patches extracted from the original image dataset (600 pixels × 600 pixels, 800 pixels × 800 pixels, 1,000 pixels × 1,000 pixels, 1,200 pixels × 1,200 pixels, 1,400 pixels × 1,400 pixels, and 1,600 × 1,600 pixels) to determine the most appropriate size. For the most appropriate patch size dataset, we rotated the patches with two strategies, fixed rotation and random rotation, to achieve the expansion of the image dataset. Compensated or uncompensated processing was performed for images obtained with different rotations. Then, image datasets obtained from different processing strategies were used to train the model and compare the impact of different data enhancement methods on model wood identification performance.

2.4 Evaluation

The performance of the trained models was evaluated using a test set. The highest identification accuracy of all models based on test set images is reported. A confusion matrix is given to better understand the classification results of wood species given by the model and to analyze the causes of species discrimination errors in terms of wood anatomy. The confusion matrix contains information about the true and predicted values of the classification and reflects the wood species classification results. Accuracy usually describes model performance on all sample categories and is used when all sample categories are equally important. The higher the value is, the better the performance of the classification model. Accuracy is calculated using true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs), which are shown in Equation 5.

$$Accuracy\ (\mathrm{A}) = \frac{TN + TP}{TN + TP + FP + FN} \tag{5}$$

To further test the generalization ability of the trained model, for each species, we captured 10 images from each sample purchased from the Guangdong Yuzhu Timber Market to conduct real-world identification.

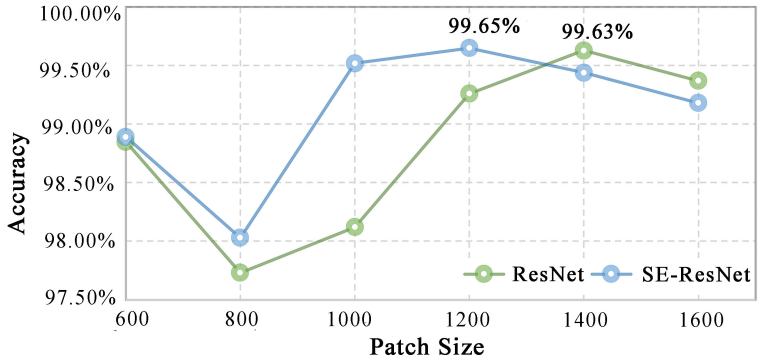


FIGURE 4
The model performance when using images of different patch sizes.

3 Results and discussion

3.1 Appropriate image data augmentation for wood species identification models

3.1.1 Image cropping

Cropping is the common means in the present study. Figure 4 displays the model performance of different patch sizes in this experiment. This figure shows that the ResNet and SE-ResNet models can achieve high identification accuracy when the patch size is larger than 1,200 pixels × 1,200 pixels, which aligns with previous research (He et al., 2020). The appropriate perceptual field of view size is an important factor affecting the classification performance of the model. The computational cost of high-resolution images is too high; thus, the image size must be adjusted but kept within the range where the desired features can be extracted. In addition, information loss should be accounted for when choosing the size of the wood image. For macroscopic images of wood cross-sections, the size of the image feature field of view needs to be balanced with the number of images. Therefore, to obtain as much image data as possible with low overlap while ensuring model accuracy, we consider a patch size of 1,200 pixels × 1,200 pixels to be the optimal size for model training.

3.1.2 Image rotation

The ResNet and SE-ResNet models were established based on the most suitable patch size. The validity of different image rotation methods on wood identification models was compared for the first time. Table 3 shows that the fixed rotation method reached higher accuracy for the ResNet (99.65%) and SE-ResNet (99.45%) models. Moreover, these models exhibited relatively high accuracy after using 0 as a compensation item to complete the cropped image; both values were above 99.28%. The results showed that using a fixed rotation method and image compensation to obtain adequate training data is feasible. The accuracy and loss curves of the SEResNet model during the training and test process are shown in Figure 5. It showed that learning rate decays by 10 times and tended to be flat when training to 15, respectively. At the same epoch, the test set reached the highest accuracy with a loss value of less than 0.05 (Figure 5B). It showed that this model has strong generalization and stability ability and is able to conduct wood species identification.

Some existing publicly accessible datasets provide numerous images that can be used in computer vision research, but no more attention is given to preprocessing images from different sources before use. Geus et al. (2021) conducted an experiment that applied a rotation of 1° to each image within the training set, covering a range from −15° to 15°. Nevertheless, this experiment did not give specific conclusions about wood image preprocessing methods, and no further research has been carried out. In this experiment, the image undergoes segmentation, followed by rotation over a comprehensive 360° range and subsequent compensation. The results serve to validate the efficacy of the proposed preprocessing approach in substantially expanding the available dataset.

TABLE 3 Image dataset details with the different patch sizes.

Patch size	Number of train patch	Number of val patch	Number of test patch	Number of total patch
1,600	32,776	3,790	3,889	40,455
1,400	32,157	4,296	4,002	40,455
1,200	32,408	4,029	4,018	40,455
1,000	32,204	3,988	4,263	40,455
800	32,286	3,857	4,312	40,455
600	32,215	4,170	4,070	40,455

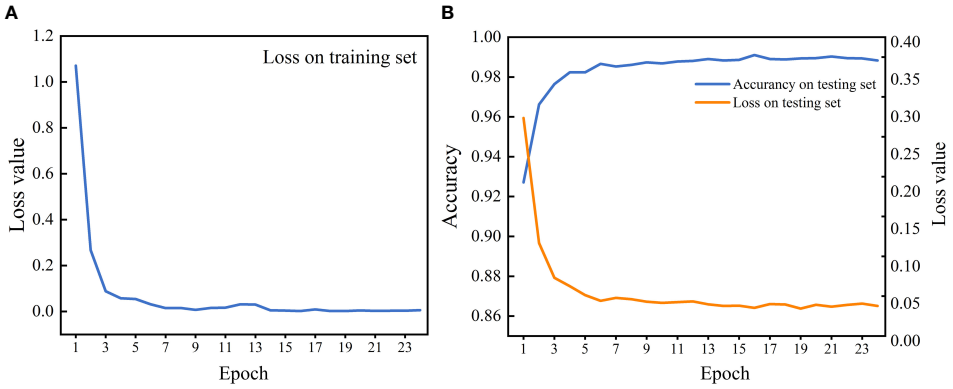


FIGURE 5
The loss value and accuracy of the SEResNet model across each epoch. **(A)** The model training. **(B)** The model test.

3.2 Model accuracy improvement by adding an attention channel

CNN models commonly used in previous studies, such as AlexNet, InceptionV3, and DenseNet, have shown strong feature extraction capabilities when processing two-dimensional images (Geus et al., 2021; Wagle and Harikrishnan, 2021); however, the model performance is unsatisfactory when targeting wood image data. The wood section image contains features with strong regularity. In contrast, the variations between different wood species are subtle. In addition, the difference in wood section images is mainly reflected in factors such as texture, color, and control distribution. The attention mechanism can extract key features of similar images better. In this experiment, the channel attention mechanism is introduced to improve wood species identification accuracy at the species level.

As shown in Table 3, the model accuracy of SE-ResNet is generally higher than that of ResNet under various rotation and compensation methods. Compared with ResNet, the performance of SE-ResNet was better and suitable for identifying the CITES-listed wood species with their look-alikes. ResNet incorporates SENet, which enhances the capacity of the network to learn identification keys by adaptively recalibrating the channelwise feature responses. The attention mechanism in computer vision is inspired by the human attention mechanism, which imitates the manner in which people focus more on specific information in an image while ignoring the rest (Lu et al., 2023). As a module that can affect the performance of the model, the attention mechanism can focus limited attention on only the most essential information to save computational resources and obtain the most effective information quickly. In particular, when the original model is underfitting with fewer parameters and cannot fully learn the training data rules, adding the attention module enhances the expressive power of the model, improving the underfitting problem and accuracy.

3.3 Discrimination of CITES-listed species from their look-alikes

In this study, the SE-ResNet model obtained the highest accuracy for wood species identification on a dataset with fixed and complementary rotation. The confusion matrix in Figure 6 shows the classification results of the model for each wood species in the test set. Overall, the classification accuracy of the 28 tree species reached 99.65%, with 20 wood species correctly classified. Surprisingly, the predictions made by the SE-ResNet model for five *Guibourtia* species were perfect. When discriminating CITES-listed *Dalbergia* species, they were identified with an accuracy of over 95% with this model. The CITES-listed *Dalbergia* species was completely distinguished from eight look-alike species, including *Platymiscium*, *Swartzia*, and *Bobgunnia*. However, approximately 3.23% of the *Dalbergia cearensis* images were misclassified as *Pterocarpus angolensis*.

Within *Pterocarpus*, all the CITES-listed *Pterocarpus* species were completely discriminated from their look-alikes, except for *Pterocarpus angolensis*, which was confused with *Pterocarpus indicus* and *Pterocarpus macrocarpus*. For the closely related *Pterocarpus* species, *Pterocarpus angolensis*, *Pterocarpus erinaceus*, *Pterocarpus macrocarpus*, and *Pterocarpus indicus* were misjudgments between each other. These four species have large wood anatomical similarities, such as the axial parenchyma arrangement and ray type, which make them difficult to distinguish completely and accurately on the basis of macroscopic images of a single cross-section. Not surprisingly, *Pterocarpus santalinus* and *Pterocarpus tinctorius* appeared to be misidentified, with 5.88% of the images of *Pterocarpus tinctorius* misclassified as *Pterocarpus santalinus*. With the traditional methods, it is extremely difficult to distinguish between these two species because of their highly similar macroscopic and microscopic structural characteristics. Therefore, *Pterocarpus tinctorius* was listed in CITES Appendix II in 2019 due to its over-exploitation and extreme similarity with *Pterocarpus santalinus*, listed in CITES Appendix II in 2007.

The vast majority of wood and wood products have lost the key identification characteristics, such as leaves and bark, which increases the difficulty of wood identification. In particular, wood species of the same genus or even closely related genera have high similarities in appearance and anatomical characteristics. Not surprisingly, compared to human vision-based methods, the results demonstrated in this experiment can provide higher identification accuracy (Wiedenhoeft et al., 2019). Simultaneously, the deep learning model effectively eradicates human subjectivity

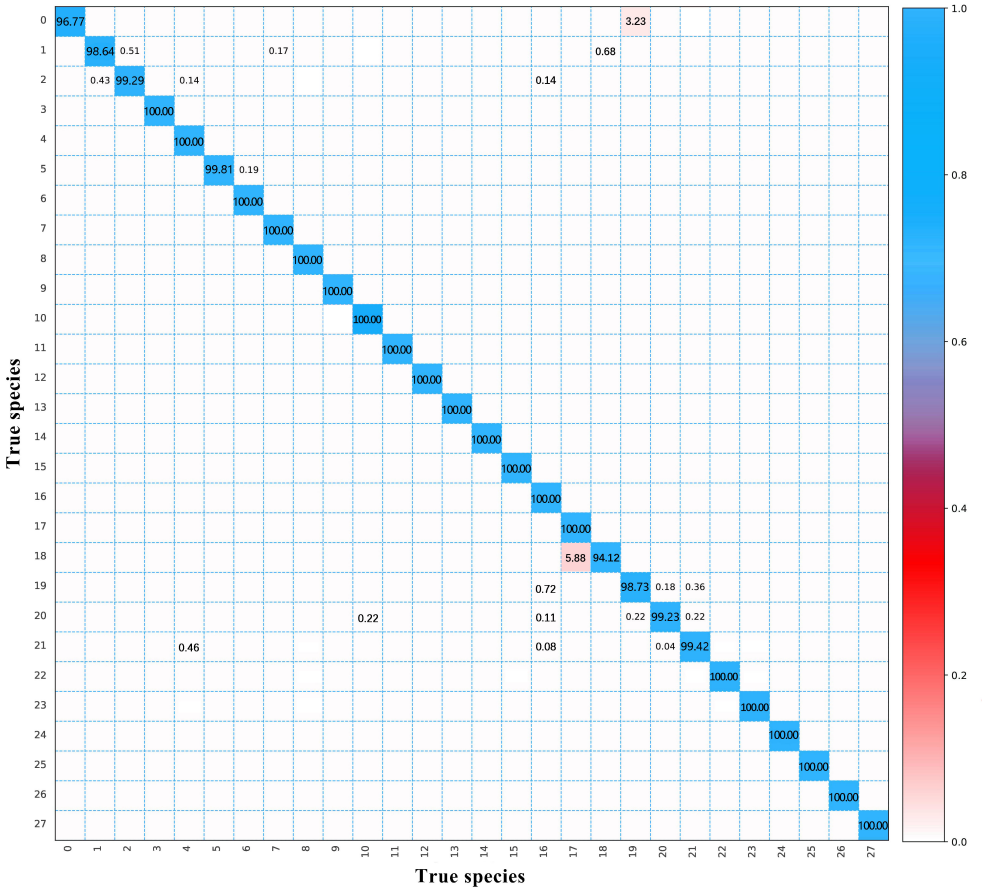


FIGURE 6
The classification results of the SE-ResNet model with the test dataset. 0—Dalbergia cearensis; 1—Dalbergia cochinchinensis; 2—Dalbergia latifolia; 3—Dalbergia melanoxylon; 4—Dalbergia oliveri; 5—Dalbergia retusa; 6—Dalbergia stevensonii; 7—Dalbergia tucurensis; 8—Platymiscium pinnatum; 9—Platymiscium polystachyum; 10—Platymiscium trinitatis; 11—Swartzia bannia; 12—Swartzia benthamiana; 13—Swartzia fistuloides; 14—Swartzia leiocalycina; 15—Bobgunnia madgascariensis; 16—Pterocarpus erinaceus; 17—Pterocarpus santalinus; 18—Pterocarpus tinctorius; 19—Pterocarpus angolensis; 20—Pterocarpus indicus; 21—Pterocarpus macrocarpus; 22—Baphia nitida; 23—Guibourtia demeusei; 24—Guibourtia tessmannii; 25—Guibourtia arnoldiana; 26—Guibourtia ehie; 27—Guibourtia conjugate.

TABLE 4 The identification performance of the models for independent images.

Label	Species (class)	SE-Resnet		ResNet	
		Accuracy%	Misidentified labels	Accuracy%	Misidentified labels
0	*Dalbergia cearensis*	100	/	100	/
1	*Dalbergia cochinchinensis*	80	*Dalbergia retusa* *Dalbergia stevensonii*	0	*Pterocarpus santalinus* *Dalbergia latifolia* *Dalbergia stevensonii*
2	*Dalbergia latifolia*	100	/	100	/
3	*Dalbergia melanoxylon*	100	/	100	/
4	*Dalbergia oliveri*	80	*Dalbergia stevensonii*	100	/
5	*Dalbergia retusa*	100	/	80	*Dalbergia oliveri*
6	*Dalbergia stevensonii*	100	/	0	*Dalbergia tucurensis* *Dalbergia retusa*
7	*Dalbergia tucurensis*	80	*Dalbergia stevensonii*	70	*Pterocarpus indicus* *Dalbergia oliveri*

(Continued)

TABLE 4 Continued

Label	Species (class)	SE-Resnet		ResNet	
		Accuracy%	Misidentified labels	Accuracy%	Misidentified labels
8	*Platymiscium pinnatum*	40	*Platymiscium polystachyum* *Platymiscium trinitatis*	70	*Platymiscium trinitatis*
9	*Platymiscium polystachyum*	20	*Platymiscium pinnatum*	40	*Platymiscium pinnatum*
10	*Platymiscium trinitatis*	20	*Dalbergia melanoxylon* *Dalbergia retusa*	50	*Dalbergia stevensonii* *Platymiscium polysta*
11	*Swartzia bannia*	100	/	60	*Dalbergia stevensonii* *Dalbergia oliveri*
12	*Swartzia benthamiana*	60	*Dalbergia cochinchinensis* *Dalbergia stevensonii*	70	*Pterocarpus angolensis*
13	*Swartzia fistuloides*	100	/	100	/
14	*Swartzia leiocalycina*	40	*Platymiscium trinitatis* *Swartzia fistuloides*	10	*Swartzia bannia* *Platymiscium trinitatis*
15	*Bobgunnia madgascariensis*	100	/	90	*Dalbergia cearensis*
16	*Pterocarpus erinaceus*	100	/	100	/
17	*Pterocarpus santalinus*	90	*Pterocarpus macrocarpus*	90	*Dalbergia retusa*
18	*Pterocarpus tinctorius*	90	*Pterocarpus santalinus*	100	/
19	*Pterocarpus angolensis*	100	/	100	/
20	*Pterocarpus indicus*	100	/	90	*Dalbergia oliveri*
21	*Pterocarpus macrocarpus*	80	*Pterocarpus angolensis* *Pterocarpus indicus*	50	*Dalbergia oliveri*
22	*Baphia nitida*	100	/	100	/
23	*Guibourtia demeusei*	100	/	90	*Pterocarpus tinctorius*
24	*Guibourtia tessmannii*	100	/	10	*Guibourtia demeusei*
25	*Guibourtia arnoldiana*	40	*Guibourtia ehie*	0	*Guibourtia ehie*
26	*Guibourtia ehie*	100	/	100	/
27	*Guibourtia conjugata*	100	/	100	/
	Total	**82.3**		**70.4**	

The meaning of the symbol "/" represents the species don't misidentified as other wood species.
The bold values represents the average accuracy of SE-ResNet and ResNet.

and assumes the role of a professional with greater efficiency and precision.

3.4 Prospect of wood species identification based on a deep learning model

Table 4 presents the discriminative results of the SE-ResNet model for images collected from the timber markets. For completely unfamiliar samples, the recognition accuracies of SE-Resnet and Resnet are 82.3% and 70.4%, respectively; the generalization ability of SE-Resnet model is superior. For SE-Resnet, the classification results show that images of 16 species were fully identified and 6 species had 8 or 9 out of 10 images accurately identified. In this regard, the CITES-listed tree species of *Pterocarpus* and *Guibourtia* were all distinguished from their look-alikes. *Dalbergia* spp., *D. cearensis*, and *D. cochinchinensis* were also accurately identified. However, this model performed worse when discriminating *Platymiscium* spp., with images of all three species at a level of poor stability. Although the number of samples is large, the effective area of the image that can be acquired for each sample is small and there is a relatively small number of images of *Platymiscium* used for modeling. Such a phenomenon also appears in *Swartzia benthamiana* with *Swartzia leiocalycina*. It demonstrated that even if more images could be obtained through data enhancement strategies, fully covering the variability within species would not have been possible.

Based on this result, we considered that the model proposed in this experiment could identify 22 tree species accurately and consistently. The SE-ResNet model has great potential for generalization, and the model gives better recognition for wood

species with more specimens. Thus, it is recommended that more specimens of the same species be collected to train the model to cover as much variability as possible and that, where possible, many specimens be used to test the generalization ability of the model in future studies, which can enhance the accuracy, generalization ability, and applicability of the trained model.

Our experiment explored the effect of image cropping and rotation on the performance of deep learning models for wood image identification and added an attention channel module into the ResNet-50 model. It was found that fixed rotation and cropping are effective data enhancement methods for wood images. In addition, we added only the attention channel module into the models, and other attention mechanisms, such as multi-head self-attention, self-attention, and convolutional block attention modules, were not tested in wood species identification (Woo et al., 2018; Guo et al., 2022). Follow-up research should explore more data enhancement methods, such as mirroring and scaling. The generative adversarial network (GAN) technique is a new computer vision method. The GAN-based oversampling technique not only increases the minority class representation to solve class imbalance problems but also may help to prevent overfitting (Sampath et al., 2021). However, GANs have not been reported in wood image recognition.

Numerous practical tools have been developed to address the specific field application requirements, exemplified by solutions such as MyWood-ID (Tang et al., 2018) and XyloPhone (Wiedenhoeft, 2020). Considering the comprehensive research conducted in this paper, an intelligent wood identification system was devised and successfully deployed for on-site inspections carried out by customs officials. Compared to other emerging techniques (NIRS, DART-MS, and DNA barcode), the computer vision-based wood identification approach demonstrates considerable promise in achieving species-level precision in field screening for wood species.

At this research stage, the dataset images are mainly processed on the surface of the wood by both knifing and sanding in the laboratory so that the anatomical characteristics of the wood can be fully exposed before the images are acquired. The experimentally acquired images after fine sanding are idealized, and the models trained based on these image data may not match the practical application. Therefore, exploring the deep learning models with wood images of rough surfaces is necessary to reduce the workload of wood surface treatment by processing rough surface images.

4 Conclusion

The traditional wood species identification method has a history of nearly a century, forming a complete identification process and norms, and still occupies a dominant position. However, in the new era of demand, traditional methods need to be complemented by new technologies to achieve species-level wood species identification. Faced with the current conservation pressure of endangered tree species, a more accurate, convenient, and economically friendly method is urgently needed for the identification of wood species. Computer vision is the most feasible technique for traded wood species identification coupled with deep learning, especially for the field inspection at import and export ports.

The results of this study showed that the 1,200 pixel × 1,200 pixel patch size can be applied as the best choice for the training model and that the number of wood images can be effectively expanded by image cropping and rotation. In addition, the channel attention mechanism (SENet) module is added to the CNN structure to identify CITES-listed tree species and their look-alikes with an accuracy of over 99%. It shows a relatively satisfactory performance in real-world identification. For the public, the model does not require extensive knowledge of wood anatomy and experience in species identification, which greatly reduces the complexity of the traditional wood identification process.

This work not only provides a CNN model along with added attention channels for successful identification but also provides a standard guideline for image data enhancement when conducting wood species identification. Further studies are needed to explore the interspecific wood anatomy features with deep learning models and feature visualization. The results of this study show that the wood transverse image dataset coupled with the SE-ResNet model can accurately discriminate CITES-listed species from their look-alikes to combat illegal timber trade and contribute to tree species conservation.

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

SL: Data curation, Writing – original draft, Writing – review & editing, Methodology, Visualization. CZ: Methodology, Software, Writing – review & editing. JW: Data curation, Writing – review & editing. YL: Data curation, Writing – review & editing. JY: Methodology, Writing – review & editing. ZZ: Software, Writing – review & editing. YY: Funding acquisition, Writing – review & editing. TH: Funding acquisition, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the Project of Natural Science Foundation of China (32201496) and the Project of Natural Science Foundation of Beijing (6224064).

Acknowledgments

We would like to thank Alex C. Wiedenhoeft, Center for Wood Anatomy Research, USDA Forest Service, Forest Products Laboratory, for providing wood specimens used to collect images and giving good suggestions on experiment design. We wish to acknowledge the assistance of Prof. Xiaomei Jiang in participating in the discussion. We also express our gratitude to Dr. Jie Wang, Miss. Jiabao Chen, and

Mr. Yupei Wei for their help with the figures in this paper, and Dr. Yu Sun for his contributions to image collection.

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

Atikah, T., Wardani, W., and Surya, D. (2021). "Population, economic value and the trade of Dalbergia latifolia in Lombok, West Nusa Tenggara, Indonesia," in *In IOP Conference Series: Earth and Environmental Science*, Vol. 762. 012015. doi: 10.1088/1755-1315/762/1/012015

Bergo, M. C., Pastore, T. C., Coradin, V. T., Wiedenhoeft, A. C., and Braga, J. W. (2016). NIRS identification of *Swietenia macrophylla* is robust across specimens from 27 countries. *IAWA J.* 37, 420–430. doi: 10.1163/22941932-20160144

Brancalion, P. H. S., Almeida, D. R. A. D., Vidal, E., Molin, P. G., Sontag, V. E., Souza, S. E. X. F., et al. (2018). Fake legal logging in the Brazilian Amazon. *Sci. Adv.* 4, eaat1192. doi: 10.1126/sciadv.aat1192

Cazzolla, G. R., Reich, P. B., Gamarra, J. G., Crowther, T., Hui, C., Morera, A., et al. (2022). The number of tree species on Earth. *Proc. Natl. Acad. Sci. U.S.A.* 119, e2115329119. doi: 10.1073/pnas.2115329119

CITES (2023). *Appendices I, II and III*. Available online at: https://cites.org/eng/app/appendices.php (Accessed June 25, 2023).

Committee, I. H. (1989). IAWA list of microscopic features for hardwood identification. *IAWA J.* 10, 219–332. doi: 10.1163/22941932-90000349

Deklerck, V., Mortier, T., Goeders, N., Cody, R. B., Waegeman, W., Espinoza, E., et al. (2019). A protocol for automated timber species identification using metabolome profiling. *Wood Sci. Technol.* 53, 953–965. doi: 10.1007/s00226-019-01111-1

Dyk, D. A. V., and Meng, X. L. (2001). The art of data augmentation. *J. Comput. Graph Stat.* 10, 1–50. doi: 10.1198/10618600152418584

Falk, B. (2009). Wood as a sustainable building material. *For Prod J.* 59, 6–13. doi: 10.5176/2301-394x_ace18.43

Geus, A. R. D., Backes, A. R., Gontijo, A. B., Albuquerque, G. H. Q., and Souza, J. R. (2021). Amazon wood species classification: a comparison between deep learning and pre-designed features. *Wood Sci. Technol.* 55, 857–872. doi: 10.1007/s00226-021-01282-w

Goldsmith, E. I. (1978). The convention on international trade in endangered species of wild fauna and flora. *J. Med. Primatol.* 7, 122–124. doi: 10.1159/000459796

Guo, M. H., Xu, T. X., Liu, J. J., Liu, Z. N., Jiang, P. T., Mu, T. J., et al. (2022). Attention mechanisms in computer vision: A survey. *Comput. Visual media* 8, 331–368. doi: 10.1007/s41095-022-0271-y

He, T., Liu, S. J., Lu, Y., Zhang, Y. G., Jiao, L. C., and Yin, Y. F. (2021). iWood: An automated wood identification system for endangered and precious tree species using convolutional neural networks. *Scientia Silvae Sinicae.* 57, 152–159. doi: 10.11707/j.1001-7488.20210915

He, T., Lu, Y., Jiao, L. C., Zhang, Y. G., Jiang, X. M., and Yin, Y. F. (2020). Developing deep learning models to automate rosewood tree species identification for CITES designation and implementation. *Holzforschung.* 74, 1123–1133. doi: 10.1515/hf-2020-0006

He, K., Zhang, X., Ren., S., and Song, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, USA. 770–778. doi: 10.1109/CVPR.2016.90

Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. *Proc. Proc. IEEE Conf. Comput. Vision Pattern Recognit.* 2018, 7132–7141. doi: 10.1109/CVPR.2018.00745

Hwang, S. W., and Sugiyama, J. (2021). Computer vision-based wood identification and its expansion and contribution potentials in wood science: a review. *Plant Methods* 17, 1–21. doi: 10.1186/s13007-021-00746-1

INTERPOL (2021). *Forestry Crime Fact sheet 2021*. Available online at: https://www.interpol.int/content/download/17367/file/Forestry%20Crime%20Fact%20sheet%202021.pdf (Accessed May 15, 2023).

IUCN (2022). *The IUCN Red List of Threatened Species*. Available online at: https://www.iucnredlist.org (Accessed December 15, 2022).

Jiao, L. C., Lu, Y., He, T., Guo, J., and Yin, Y. F. (2020). DNA barcoding for wood identification: Global review of the last decade and future perspective. *IAWA J.* 41, 620–643. doi: 10.1163/22941932-bja10041

Kırbaş, İ., and Çifci, A. (2022). An effective and fast solution for classification of wood species: A deep transfer learning approach. *Ecol. Inform.* 69, 101633. doi: 10.1016/j.ecoinf.2022.101633

Kwon, O., Lee, H. G., Lee, M. R., Jang, S., Yang, S. Y., Park, S. Y., et al. (2017). Automatic wood species identification of Korean softwood based on convolutional neural networks. *Korean J. Wood Sci. Technol.* 45, 797–808. doi: 10.5658/WOOD.2017.45.6.797

Lambdon, P. W., and Ellick, S. (2016). *Lachanodes arborea* (The IUCN Red List of Threatened Species) (Accessed 06 June 2024).

Latib, H. A., Liat, L. C., Ratnasingam, J., Law, E. L., Azim, A. A. A., Mariapan, M., et al. (2020). Suitability of paulownia wood from Malaysia for furniture application. *BioResources.* 15, 4727–4737. doi: 10.15376/biores.15.3.4727-4737

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., and Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing* 234, 11–26. doi: 10.1016/j.neucom.2016.12.038

Lu, S., Liu, M., Yin, L., Yin, Z., Liu, X., and Zheng, W. (2023). The multi-modal fusion in visual question answering: a review of attention mechanisms. *PeerJ Comput. Sci.* 9, e1400. doi: 10.7717/peerj-cs.1400

Oktaria, A. S., Prakasa, E., and Suhartono, E. (2019). Wood species identification using convolutional neural network (cnn) architectures on macroscopic images. *J. Inf Comput. Sci.* 4, 274–283. doi: 10.25126/jitecs.201943

Pan, X., Li, K., Chen, Z., and Yang, Z. (2021). Identifying wood based on near-infrared spectra and four gray-level co-occurrence matrix texture features. *Forests.* 12, 1527. doi: 10.3390/f12111527

Price, E. R., Miles-Bunch, I. A., Gasson, P. E., and Lancaster, C. A. (2021). *Pterocarpus* wood identification by independent and complementary analysis of DART-TOFMS, microscopic anatomy, and fluorescence spectrometry. *IAWA J.* 42, 397–418. doi: 10.1163/22941932-bja10064

Ravindran, P., Costa, A., Soares, R., and Wiedenhoeft, A. C. (2018). Classification of CITES-listed and other neotropical Meliaceae wood images using convolutional neural networks. *Plant Methods* 14, 1–10. doi: 10.1186/s13007-018-0292-9

Ravindran, P., Thompson, B. J., Soares, R. K., and Wiedenhoeft, A. C. (2020). The XyloTron: flexible, open-source, image-based macroscopic field identification of wood products. *Front. Plant Sci.* 11. doi: 10.3389/fpls.2020.01015

Ravindran, P., and Wiedenhoeft, A. C. (2020). Comparison of two forensic wood identification technologies for ten meliaceae woods: computer vision versus mass spectrometry. *Wood Sci. Technol.* 54, 1139–1150. doi: 10.1007/s00226-020-01178-1

Rivers, M. (2017). The Global Tree Assessment–red listing the world's trees. *BGjournal.* 14, 16–19. Available at: https://www.jstor.org/stable/26369172.

Sampath, V., Maurtua, I., Martín, J. J. A., and Gutierrez, A. (2021). A survey on generative adversarial networks for imbalance problems in computer vision tasks. *J. Big Data.* 8, 27. doi: 10.1186/s40537-021-00414-0

Shmulsky, R., and Jones, P. D. (2019). ""Tree growth and production of woody tissue."," in *The Forest Products and Wood Science* (Wiley, Chichester), 1–20. doi: 10.1002/9781119426400

Su, H. S., Lv, J., Ding, Z. P., Tang, Y. J., Chen, X. D., Zhou, Q., et al. (2021). Wood identification algorithm based on improved residual neural network. *Chin. J. Sci. Silvae Sin.* 57, 147–154. doi: 10.11707/j.1001-7488.20211215

Tang, X. J., Tay, Y. H., Siam, N. A., and Lim, S. C. (2018). "MyWood-ID: Rapid and robust automated macroscopic wood identification system using smartphone with macro-lens," in *In Proceedings of the 2018 International Conference on Computational Intelligence and Intelligent Systems—CIIS 2018*. 37–43.

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. (2019). Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.* 2018, 1–13. doi: 10.1155/2018/7068349

Wagle, S. A., and Harikrishnan, R. (2021). Comparison of plant leaf classification using modified alexNet and support vector machine. *Trait Signal.* 1, 38. doi: 10.18280/ts.380108

Wang, W., Yang, Y., Wang, X., Wang, W., and Li, J. (2019). Development of convolutional neural network and its application in image classification: a survey. *Opt Eng.* 8, 040901–040901. doi: 10.1117/1.OE.58.4.040901

Wick, E. (2019). "Impacts of the Convention on International Trade in Endangered species of Wild Fauna and Flora (CITES) on various stakeholders in the music industry," in *Swiss Wood Solution*. (Swiss: Swiss Wood Solution).

Wiedenhoeft, A. C. (2020). The XyloPhone: toward democratizing access to high-quality macroscopic imaging for wood and other substrates. *IAWA J.* 41, 699–719. doi: 10.1163/22941932-bja10043

Wiedenhoeft, A. C., Simeone, J., Smith, A., Parker-Forney, M., Soares, R., and Fishman, A. (2019). Fraud and misrepresentation in retail forest products exceeds US forensic wood science capacity. *PloS One* 14, e0219917. doi: 10.1371/journal.pone.0219917

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). "Cbam: Convolutional block attention module," in *In Proceedings of the European conference on computer vision (ECCV)*. 3–19.

Wu, F., Gazo, R., Haviarova, E., and Benes, B. (2021). Wood identification based on longitudinal section images by using deep learning. *Wood Sci. Technol.* 55, 553–563. doi: 10.1007/s00226-021-01261-1

Yin, Y. F., Jiao, L. C., He, T., and Jiang, X. M. (2022). *Identification manual of endangered and precious timber species common in trades* (Beijing: Science Press).

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 09 December 2024
DOI 10.3389/fpls.2024.1373104

Check for updates

OPEN ACCESS

EDITED BY
Jeffrey Too Chuan Tan,
Genovasi University College, Malaysia

REVIEWED BY
Huahong Huang,
Zhejiang Agriculture and Forestry University, China
Naveen Kumar Mahanti,
Dr. Y.S.R. Horticultural University, India
Muhammad Aqib,
PMAS-Arid Agriculture University Rawalpindi, Pakistan
Yunchao Tang,
Dongguan University of Technology, China

*CORRESPONDENCE
Quan Gao
gaoq@ynau.edu.cn
Tong Li
tli@ynu.edu.cn

RECEIVED 19 January 2024
ACCEPTED 29 October 2024
PUBLISHED 09 December 2024

CITATION
Ye R, Gao Q and Li T (2024) BRA-YOLOv7: improvements on large leaf disease object detection using FasterNet and dual-level routing attention in YOLOv7.
Front. Plant Sci. 15:1373104.
doi: 10.3389/fpls.2024.1373104

COPYRIGHT
© 2024 Ye, Gao 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.

BRA-YOLOv7: improvements on large leaf disease object detection using FasterNet and dual-level routing attention in YOLOv7

Rong Ye[1,2], Quan Gao[2,3]* and Tong Li[2,3]*

[1]College of Food Science and Technology, Yunnan Agricultural University, Kunming, Yunnan, China, [2]The Key Laboratory for Crop Production and Smart Agriculture of Yunnan Province, Yunnan Agricultural University, Kunming, Yunnan, China, [3]College of Big Data, Yunnan Agricultural University, Kunming, Yunnan, China

Tea leaf diseases are significant causes of reduced quality and yield in tea production. In the Yunnan region, where the climate is suitable for tea cultivation, tea leaf diseases are small, scattered, and vary in scale, making their detection challenging due to complex backgrounds and issues such as occlusion, overlap, and lighting variations. Existing object detection models often struggle to achieve high accuracy in detecting tea leaf diseases. To address these challenges, this paper proposes a tea leaf disease detection model, BRA-YOLOv7, which combines a dual-level routing dynamic sparse attention mechanism for fast identification of tea leaf diseases in complex scenarios. BRA-YOLOv7 incorporates PConv and FasterNet as replacements for the original network structure of YOLOv7, reducing the number of floating-point operations and improving efficiency. In the Neck layer, a dual-level routing dynamic sparse attention mechanism is introduced to enable flexible computation allocation and content awareness, enhancing the model's ability to capture global information about tea leaf diseases. Finally, the loss function is replaced with MPDIoU to enhance target localization accuracy and reduce false detection cases. Experiments and analysis were conducted on a collected dataset using the Faster R-CNN, YOLOv6, and YOLOv7 models, with Mean Average Precision (mAP), Floating-point Operations (FLOPs), and Frames Per Second (FPS) as evaluation metrics for accuracy and efficiency. The experimental results show that the improved algorithm achieved a 4.8% improvement in recognition accuracy, a 5.3% improvement in recall rate, a 5% improvement in balance score, and a 2.6% improvement in mAP compared to the traditional YOLOv7 algorithm. Furthermore, in external validation, the floating-point operation count decreased by 1.4G, FPS improved by 5.52%, and mAP increased by 2.4%. In conclusion, the improved YOLOv7 model demonstrates remarkable results in terms of parameter quantity, floating-point operation

count, model size, and convergence time. It provides efficient lossless identification while balancing recognition accuracy, real-time performance, and model robustness. This has significant implications for adopting targeted preventive measures against tea leaf diseases in the future.

KEYWORDS

tea leaf diseases, dual-level routing dynamic sparse attention mechanism, FasterNet, YOLOv7 algorithm, lightweight model

1 Introduction

Yunnan is internationally recognized as the birthplace of tea trees, and the tea industry is a characteristic advantage industry in Yunnan. Yunnan's tea plantation area and the comprehensive associated output value of the industry have consistently ranked among the top in the country for many years. Yunnan has recently listed it as the province's top priority among its eight key agricultural industries. The tea industry plays a crucial role in consolidating the achievements of poverty alleviation efforts and promoting the implementation of the rural revitalization strategy, which holds significant political, social, and economic significance (Li et al., 2022; Sun et al., 2023). Most of Yunnan's tea gardens are located in mountainous areas, where production conditions are poor and mechanization levels are relatively low. The most serious issue is the insufficient investment in tea leaf scientific research, which leads to a low rate of transformation of research achievements.

Traditional agricultural producers often rely on manual experience to determine tea diseases, which is inefficient and prone to misjudging the disease cycle, resulting in the inability to take targeted protective measures in advance. This greatly reduces the accuracy and scientific nature of tea disease identification (Zhang et al., 2023). During the growth period, diseases can further intensify their spread, and new diseases are likely to occur, leading to missing the optimal treatment period (Rajathi and Parameswari, 2022).

In recent years, deep learning and image processing have been widely applied in crop disease diagnosis (Waheed et al., 2020) and gene identification (Hong et al., 2020). Applying artificial intelligence methods to crop disease diagnosis can provide a new solution for sustainable crop development and is of great significance for ensuring healthy crop growth. Disease identification generally involves four steps: image preprocessing, image segmentation, disease image feature extraction, and disease identification. Hossain et al. (Hossain et al., 2018) developed an image processing method that can analyze 11 features of tea diseases and used a support vector machine classifier to identify and classify the two most common tea diseases: tea brown blight and tea leaf spot. Sun et al. (Sun et al., 2018) improved the method of extracting significant disease maps of tea diseases from complex environments by combining simple linear iterative clustering (SLIC) and support vector machines (SVM). Hu et al. (Hu et al., 2021) developed a model for analyzing the severity of tea withering disease in natural scene photos. They used an SVM classifier to segment the disease spot location from tea withering disease leaf images to calculate the initial disease severity (IDS) index. Xu et al. (Xu et al., 2020) used an improved Faster R-CNN algorithm to identify tea bud images, but the model had poor universality and slow segmentation speed. As mentioned earlier, deep neural network technology has been proven to be effective in detecting and identifying tea diseases, but most of them are limited to diagnosing or classifying simple crop disease images. With the complexity of background images in current natural scenes, the upgrading of tea varieties, and the growth changes of multiple diseases, some traditional deep learning models have a large number of parameters and slow operation speed, making it difficult to achieve an effective balance between recognition efficiency and accuracy, which does not match the actual scenario.

With the development of deep learning, target detection algorithms are mainly divided into two categories: one-stage and two-stage detection algorithms. One-stage algorithms, such as the YOLO (Redmon et al., 2016; Zhang et al., 2022; Lin et al., 2023; Lv et al., 2023; Soeb et al., 2023; Zhao et al., 2023) series, extract features only once and are widely used in agriculture due to their evolution in the era of deep learning. Bai et al. (Bai et al., 2024) designed a lightweight and efficient T-YOLO model for the rapid and accurate detection of tea vegetative buds. This model incorporates the lightweight module C2fG2 and the efficient feature extraction module DBS into the backbone and neck of the YOLOv5 baseline model. Furthermore, the head network of the model is pruned, effectively reducing the number of parameters. Xue et al. (Xue et al., 2023) integrates self-attention and convolution (ACmix) with the Convolution Block Attention Module (CBAM) based on YOLOv5, enabling the improved YOLO-Tea model to more effectively focus on tea diseases and insect pests. Consequently, the detection results of the enhanced model are significantly superior to those of the original.

Tea gardens often have complex environmental conditions, with soil, pests, or diseases that have similar colors overlapping and causing difficulties in target detection due to the presence of irrelevant features. Therefore, several aspects need to be considered during the recognition process: 1) in natural environmental

conditions, tea leaves are often subjected to intense lighting and moderate wind speeds, which can affect the extraction of disease features; 2) the color and texture distribution of disease spots in tea leaf images vary, and multiple disease spots may coexist and overlap, causing uncertainty in the boundary between normal pixels and diseased pixels; 3) the use of multi-scale convolution and attention mechanism modules should effectively adjust the receptive field size to enhance the ability of image feature extraction by parameter tuning.

Due to the real-time image processing capability and superior training efficiency compared to other models in the YOLO series, the YOLOv7 model is considered for target detection in tea leaf disease images. Considering the presence of a large number of invalid background areas and redundant information in the samples, as well as issues such as varying resolutions, leaf deficiency, and non-uniform image quality in the same tea leaf disease image, this paper adopts YOLOv7 as the base model for object detection and conducts research and algorithm optimization specifically for the real scenes of tea leaves to improve the accuracy of tea leaf disease image recognition.

2 Data and methods

2.1 Image capture

In Yunnan region, large-leaf tea plantation covers more than 80% of the national plantation area. This article focuses on the Hekai Base in Menghai County, Xishuangbanna Prefecture, Yunnan Province (latitude 21.5, longitude 100.28) as the research object. The tea plantation is shown in Figure 1. Due to the suitable temperature and high humidity in Yunnan, the occurrence of large-leaf tea diseases is highly seasonal, with the highest incidence in autumn (Sun et al., 2020). Therefore, the shooting time for this study was from July 1st to July 15th, 2022. Considering the influence of light intensity on the disease dataset, photos were taken respectively from 9 to 11 am and from 3 to 5 pm. The image capture device used was a Canon EOS 800D, with a photo resolution of 4608×3456, saved in.PNG format.

To meet the requirements of diverse pest detection in complex environments and to ensure the authenticity of the growth environment, the captured images have the following conditions: slight occlusion, severe occlusion, overlap, natural light angles, side light angles, back light angles, etc. Examples of tea disease samples are shown in Figure 2.

2.1.1 Image preprocessing and dataset partitioning

A total of 3,246 tea disease images were collected, which included different diseases, lighting conditions, degrees of occlusion, and overlapping diseases. After screening, 2,789 qualified images were selected. Among them, 10% of the images were randomly chosen as the validation set to evaluate the generalization of the detection model, while the remaining 2,510 images were randomly divided into a training set (2,259 images) and a test set (251 images) in a 9:1 ratio. Care was taken to ensure that there were no duplicate images among the training, validation, and test sets to prevent overfitting of the model (Halstead et al., 2018). The distribution of the sample dataset is shown in Table 1. The annotation software, LabelImg, was used for manual annotation of tea disease targets in the training set. The annotations were made based on the minimum bounding rectangle around the disease to minimize the inclusion of background areas. The annotated files were saved in XML format (Jintasuttisak et al., 2022). The visualization analysis of the annotated tea disease files is shown in Figure 3. From Figure 3, it can be observed that the sizes of the bounding boxes are uneven, but the ratios are mostly distributed between 0.04 and 0.4. Small-sized disease targets are more abundant and are not easy to detect.

In order to enhance the model's generalization ability, data augmentation was performed on the images of Yunnan large-leaf sun-dried green tea diseases. Specifically, 1) image brightness



FIGURE 1
Tea plantation.



FIGURE 2
Example of tea disease samples.

adjustment was applied by increasing and decreasing the brightness by 1.4 times and 0.6 times respectively. Through these brightness transformations, the model becomes more suitable for complex tea plantation environments with changing lighting conditions; 2) image contrast adjustment was applied by increasing and decreasing the contrast by 1.4 times and 0.6 times respectively. This helps to improve the clarity, grayscale, and texture details of the tea leaf images; 3) Gaussian blur and random rotation were applied. Gaussian blur enhances the details in disease images and increases image smoothness, while random rotation enhances the adaptability of the detection model. After applying brightness and contrast enhancement, Gaussian blur, and random rotation to the selected disease images in the dataset, the total number of images reached 15534. Figure 4 illustrates the results of data augmentation.

2.2 The improved YOLOv7 network model

In single-stage object detection algorithms, YOLOv7 performs well and is the most optimized model in terms of inference speed and recognition performance among the YOLO series. Due to its shallow network depth and smaller feature map width, it achieves fast inference speed and is widely used in real-time detection of diseases in practical scenarios. YOLOv7 consists of four components: Input, Backbone, Neck, and Head.

2.3 Optimize loss function

When solving object detection problems using CNNs, regardless of whether it's a regression or classification problem, a loss function is essential and also a major factor affecting the accuracy of the results. In this paper, the Mean Position-Density IoU (MPDIoU) loss function (Xu and Jeongyoung, 2021; Ma and Xu, 2023; Ma et al., 2023) is used to replace the original YOLOv7 network model's object regression (CIoU) loss function. MPDIoU includes regression of both overlapping and non-overlapping bounding boxes, center point distance loss, and deviations in width and height. During the training process, it accurately optimizes the bounding box regression process when the predicted box and annotated box have the same center point overlap and proportional height and width deviations. This is illustrated in Figures 5, 6

In the training phase, the objective of this model optimization is to make each predicted box

$B_{prd}[x_{prd}, y_{prd}, w_{prd}, h_{prd}]^T$ as close as possible to the annotated box $B_{gt} = [x_{gt}, y_{gt}, w_{gt}, h_{gt}]^T$, minimizing the loss function L as shown below:

$$\mathcal{L} = \overset{\min}{\Theta} \sum_{B_{gt} \in \mathbb{B}_{gt}} \mathcal{L}(B_{gt}, B_{prd} | \Theta) \quad (1)$$

Where B_{gt} is the set of annotated boxes, Θ is the parameter of the regression deep model. Based on this, the penalty term of the

TABLE 1 Distribution of the sample dataset.

Types of Tea Plant Diseases	Total Number of Datasets	Train Set	Test Set	Validation Set
Tea Cloud Spot Blight	932	766	79	87
Tea Red Spot Disease	746	605	65	76
Tea White Star Disease	594	477	56	61
Tea Leaf Spot Disease	517	411	51	55

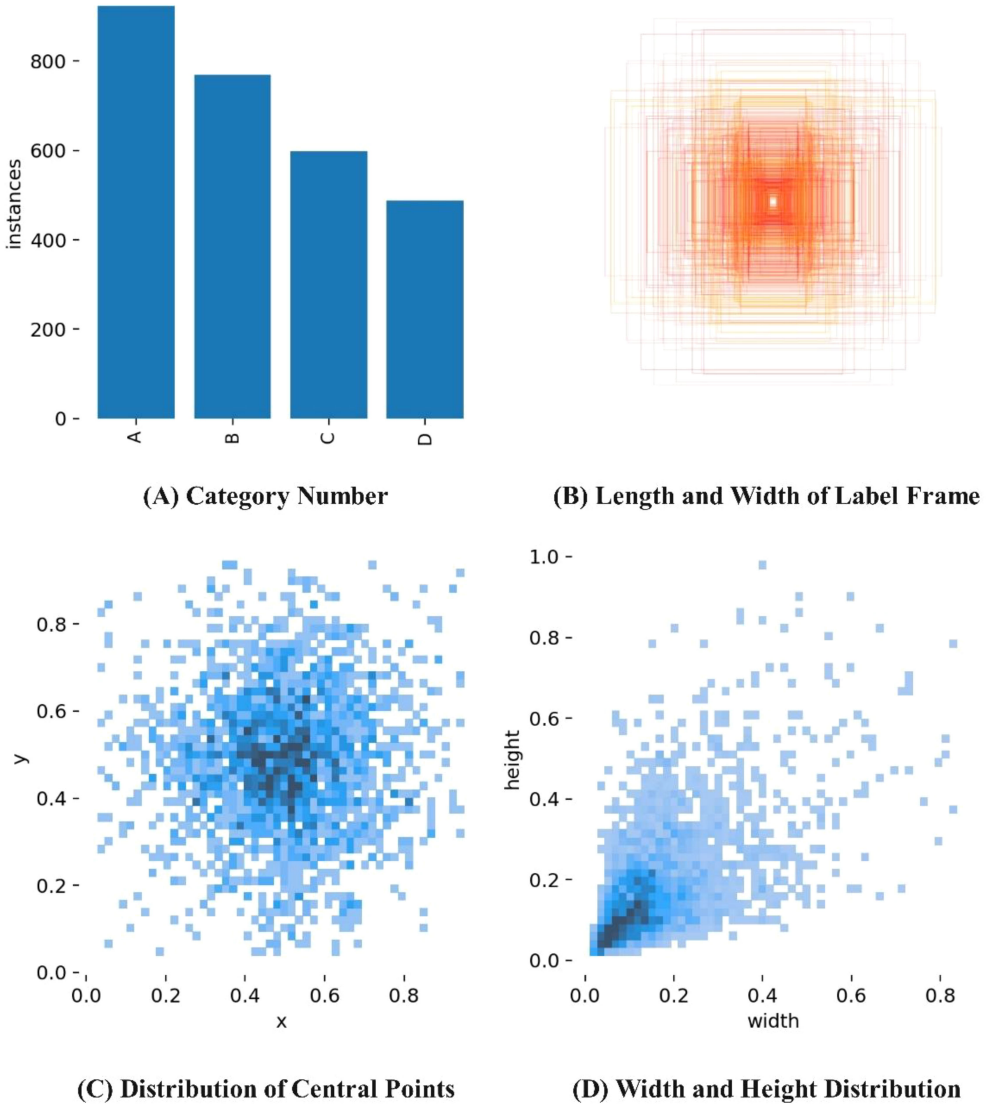


FIGURE 3
Visualization analysis of annotated tea disease files. **(A)** Category Number **(B)** Length and Width of Label Frame **(C)** Distribution of Central Points **(D)** Width and Height Distribution.

bounding box regression (MPDIoU) loss function is formulated as follows:

$$\mathcal{L}_{MPDIoU} = 1 - MPDIoU \tag{2}$$

$$MPDIoU = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2 + h^2} - \frac{d_2^2}{w^2 + h^2} \tag{3}$$

$$d_1^2 = (x_1^B - x_1^A)^2 + (y_1^B - y_1^A)^2 \tag{4}$$

$$d_2^2 = (x_2^B - x_2^A)^2 + (y_2^B - y_2^A)^2 \tag{5}$$

In Equations 2–5, $MPDIoU$ represents the regression boundary, A and B represent the predicted box and the ground truth box, (x_1^A, y_1^A) and (x_2^A, y_2^A) represent the coordinates of the top-left and bottom-right corners of box A, (x_1^B, y_1^B) and (x_2^B, y_2^B) represent the coordinates of the top-left and bottom-right corners of box B.

2.4 PConv

In addition to model accuracy, the calculation power (FLOPs) and parameter size required during forward propagation are also important factors in accelerating the inference speed of neural networks. By reducing the demands on GPU performance and memory usage, we can design a faster YOLOv7 neural network. In this study, we introduced PConv and FasterNet to replace the original network structure of YOLOv7.In the main network, we introduced a new type of convolution called PConv (Partial Convolution) (Chen et al., 2023), which reduces redundant calculations and memory accesses. The structure of PConv is shown in Figure 7. Compared to conventional convolutions Figure 7A and depth-wise convolutions Figure 7B, PConv only applies filters to a few input channels, while leaving the rest of the channels unchanged. By exploiting the redundancy in feature maps,

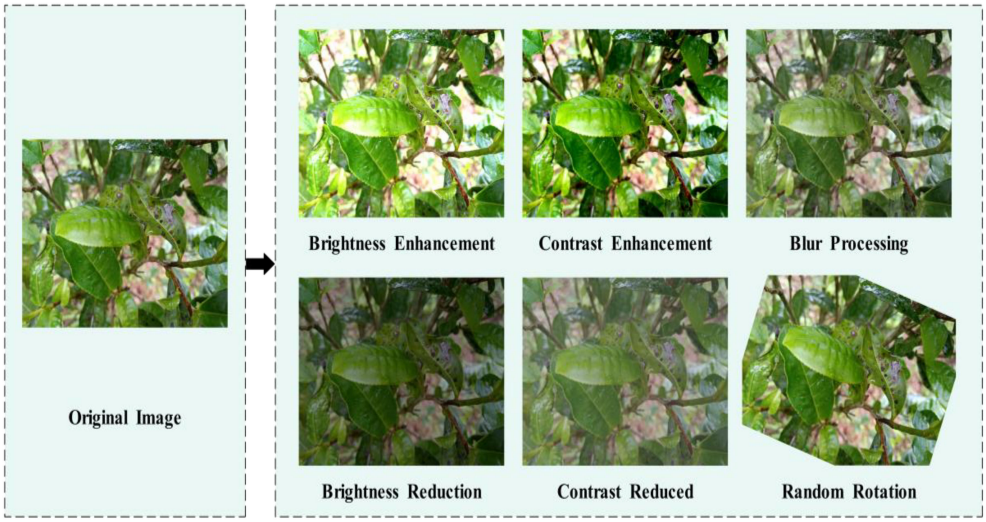


FIGURE 4
Image enhancement processing.

we systematically apply regular convolutions (Conv) to a subset of input channels while keeping the remaining channels intact.

PConv can be considered to have the same number of channels in the input and output feature maps without loss of generality. The floating point operations of PConv are shown in Equation 6, and the memory access is relatively low, as shown in Equation 7. Therefore, for a conventional ratio of $r = C_p/C = 1/4$, PConv has only 1/16 and 1/4 of the floating point operations and memory access compared to conventional convolution

$$h \times w \times k^2 \times c_p^2 \tag{6}$$

$$h \times w \times 2c_p + k^2 \times c_p^2 \approx h \times w \times 2c_p \tag{7}$$

PConv has lower FLOPs and higher FLOPS compared to conventional convolutions and depthwise convolutions. FLOPS stands for Floating Point Operations per Second and serves as a measure of effective computing speed. PConv better utilizes the computational power of devices and is also effective in spatial feature extraction.

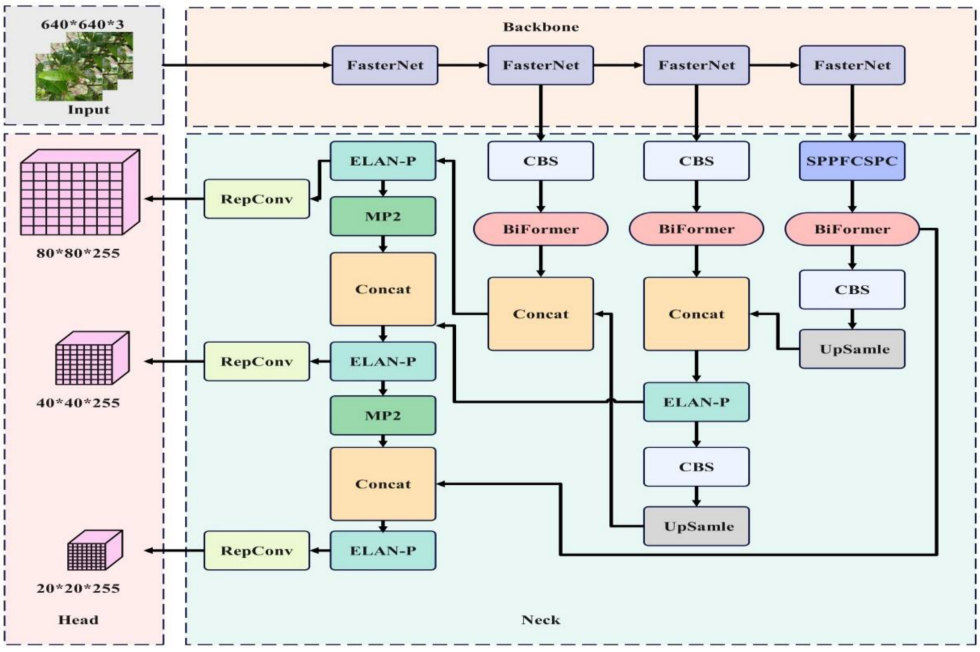


FIGURE 5
BRA-YOLOv7 network architecture.

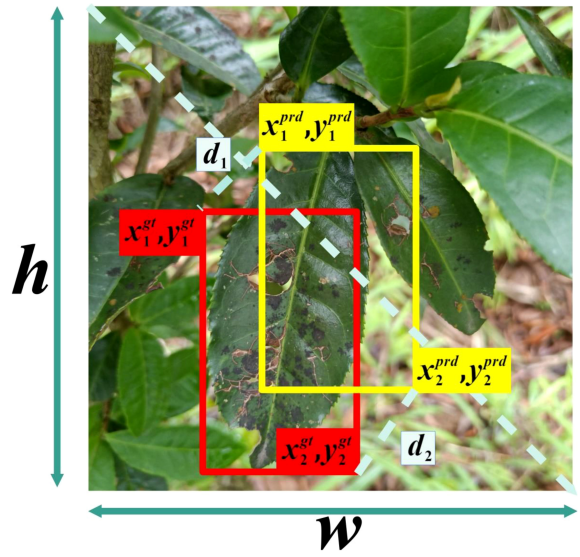


FIGURE 6
Illustration of factors in MPDIoU calculation.

The ELAN module in the backbone network can effectively improve the learning ability of the network without disrupting the original gradient path. However, the ELAN module heavily relies on CBS convolutional layers, which have a large number of parameters. Additionally, during feature extraction, the ELAN module can lead to isolated feature channels, which affects the model's detection efficiency. To enhance the feature extraction capability of the ELAN module, this paper replaces the CBS convolutional layers with PConv, which has fewer parameters. The resulting ELAN-P structure is shown in Figure 8.

2.5 Fusion of PConv with FasterNet module

FasterNet is a new family of neural networks that run faster and achieve higher accuracy on multiple processing platforms, surpassing other neural networks. FasterNet is mainly composed of four levels and its structure is shown in Figure 9. Each FasterNet Block consists of a PConv layer and two PWConv layers, presenting an inverted residual block. Stage 3 and Stage 4 layers have an expanded number of channels and higher floating-point operation efficiency per second. FasterNet performs well and is generally fast on various devices, including GPUs, CPUs, and ARM processors.

2.6 Introduction of dual-level routing in the dynamic sparse attention mechanism

In the visual Transformer, attention mechanism is a crucial part. Considering the scalability issues in terms of model computation and memory requirements, we noticed that the multi-head self-attention mechanism can enable the model to better capture discriminative features from different perspectives, thereby improving the model's performance (Gao et al., 2023; Li et al., 2023). Taking reference from YOLOv7 in Tea Tree Disease Detection training, the model performs poorly when there are occluded disease parts. Therefore, we introduce a double-layer routing-based dynamic sparse attention mechanism to achieve more flexible computation allocation and content perception.

Double-layer routed attention (Kwan-Wu et al., 2016; Jiang et al., 2023; Zhu et al., 2023) is a dynamic and query-aware sparse attention mechanism. The main idea is to filter out most irrelevant key-value pairs at a coarse-grained level and calculate coarse-grained routing features through average pooling. After computing and reading the relevance, scattered key-value pairs are collected to calculate fine-grained attention from token to token, leaving only a small number of fine-grained routing regions, as shown in Figure 10.

First, the disease image is segmented into $S\times S$ non-overlapping regions, where each region contains a feature vector of size $\frac{H\times W}{S^2}$. Here, H represents the height of the original image, and W represents the width of the original image. The feature vectors are then linearly mapped to obtain Q, K, V, as shown in Equation 8. In this equation, $X^r \in \mathbb{R}^{S^2\times\frac{HW}{S^2}\times C}$ epresents the sub-region of the feature map, W^q 、 W^k 、 W^v represent the projection weights for query, key, and value respectively. By calculating the mean values of each region, $Q^r, K^r \in \mathbb{R}^{S^2\times C}$ re obtained. The adjacency matrix of the correlation between Q^r and K^r is computed, as shown in Equation 9. By multiplying the transposed matrices of Q^r and K^r, A^r is obtained, which represents the level of correlation between two regions. we obtain A^r as shown in Equation 10. Q^r represents the region-level query, K^r represents the region-level key, and T represents the transpose operation. For coarse-grained region-level routing computation, a routing index matrix $I^r \in \mathbb{N}^{S^2\times k}$ is used. This matrix stores the indices of the top k connections for each region, while eliminating the weaker correlations. To efficiently process the collected key K and value V tensors, a public key normalization operation is applied, as shown in Equations 11, 12. Here, K^g represents the aggregated tensor for keys, K represents the original keys, I^r represents the routing index matrix, V^g represents the aggregated tensor for values, and V represents the original values. Finally, the attention mechanism is applied to K^g and V^g to obtain the feature map O, as shown in Equation 13. O represents the fine-grained attention from token to token, and $LCE(V)$ represents the local context enhancement term.

$$Q = X^r W^q, \qquad K = X^r W^k, \qquad V = X^r W^v \tag{8}$$

$$A^r = Q^r (K^r)^T \tag{9}$$

$$I^r = topkIndex(A^r) \tag{10}$$

$$K^g = gather(K, I^r) \tag{11}$$

$$V^g = gather(V, I^r) \tag{12}$$

$$O = Attention(Q, K^g, V^g) + LCE(V) \tag{13}$$

10.3389/fpls.2024.1373104

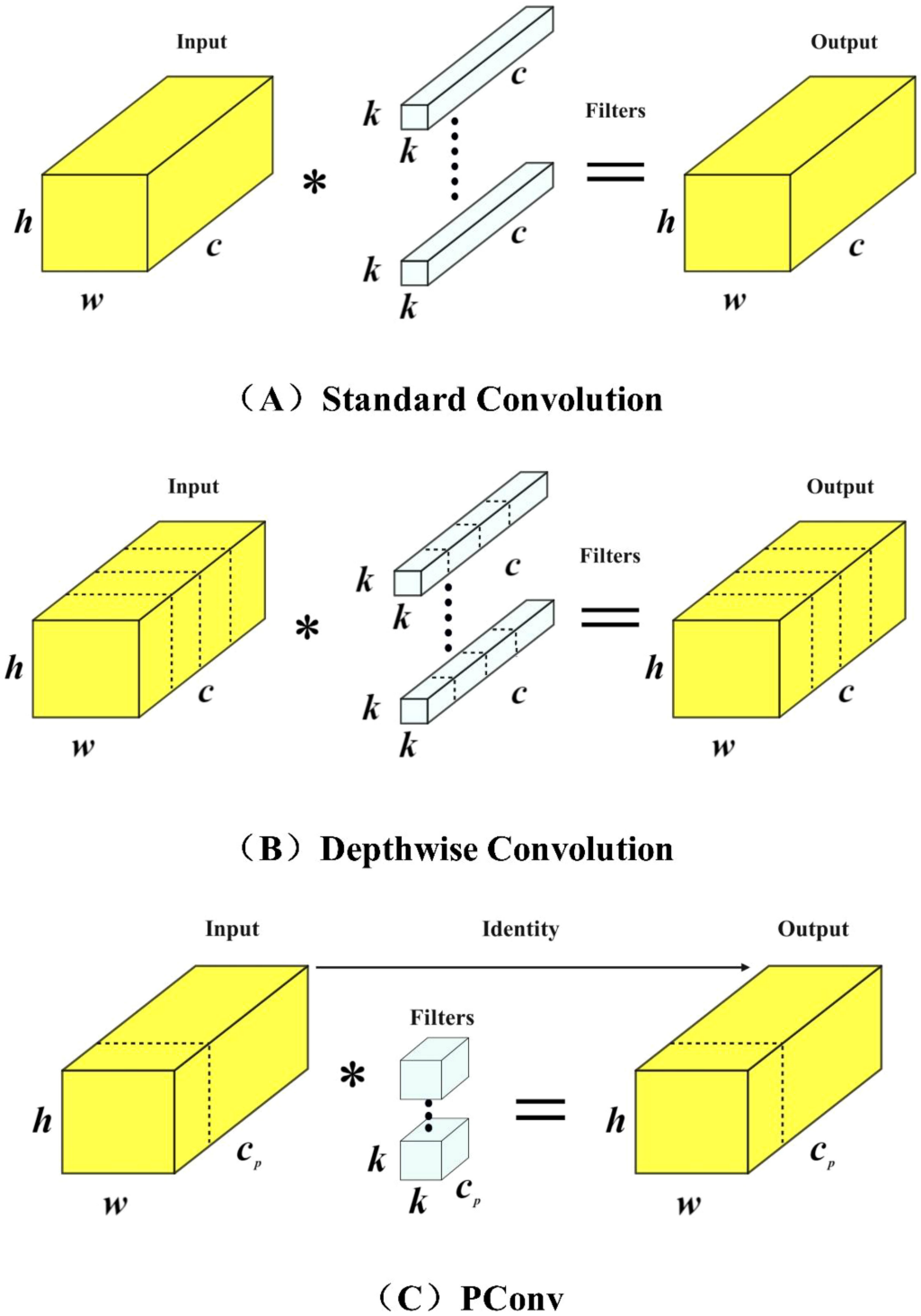


FIGURE 7
Structures of different convolutional networks. **(A)** Standard Convolution **(B)** Depthwise Convolution **(C)** PConv.

3 Experiments and discussions

To verify the detection effectiveness of BRA-YOLOv7 on the detection of tea leaf diseases, including tea leaf blight, tea red spot disease, tea white spot disease, and tea gray blight, this study conducted three comparative experiments with BRA-YOLOv7 and three popular network models: YOLOv7, Faster-RCNN, and SSD. The experiments were performed on Ubuntu 18.04.5 LTS operating system with an Intel® Xeon® Gold 5220RCPU@2.20GHz CPU and an NVIDIA Quadro RXT 5000 GPU with 32GB memory. The deep learning framework used was Pytorch 1.12.1 with CUDA 11.2. To ensure the scientific rigor of the model testing results, the hardware devices and software environment used in this study were identical.

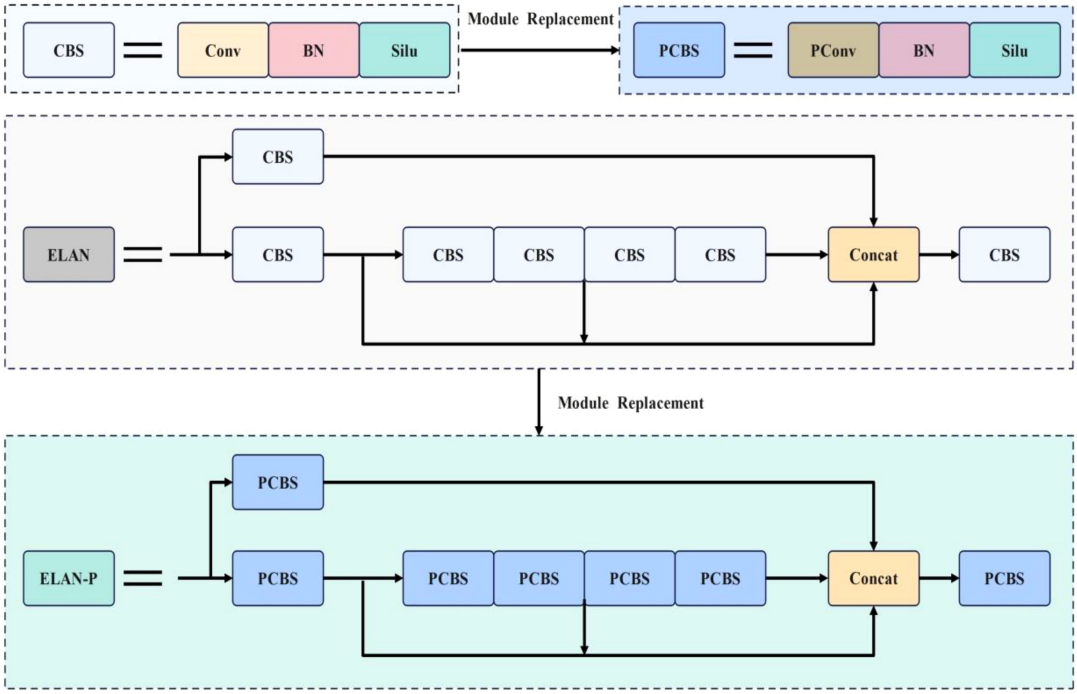


FIGURE 8
ELAN-P network structure diagram.

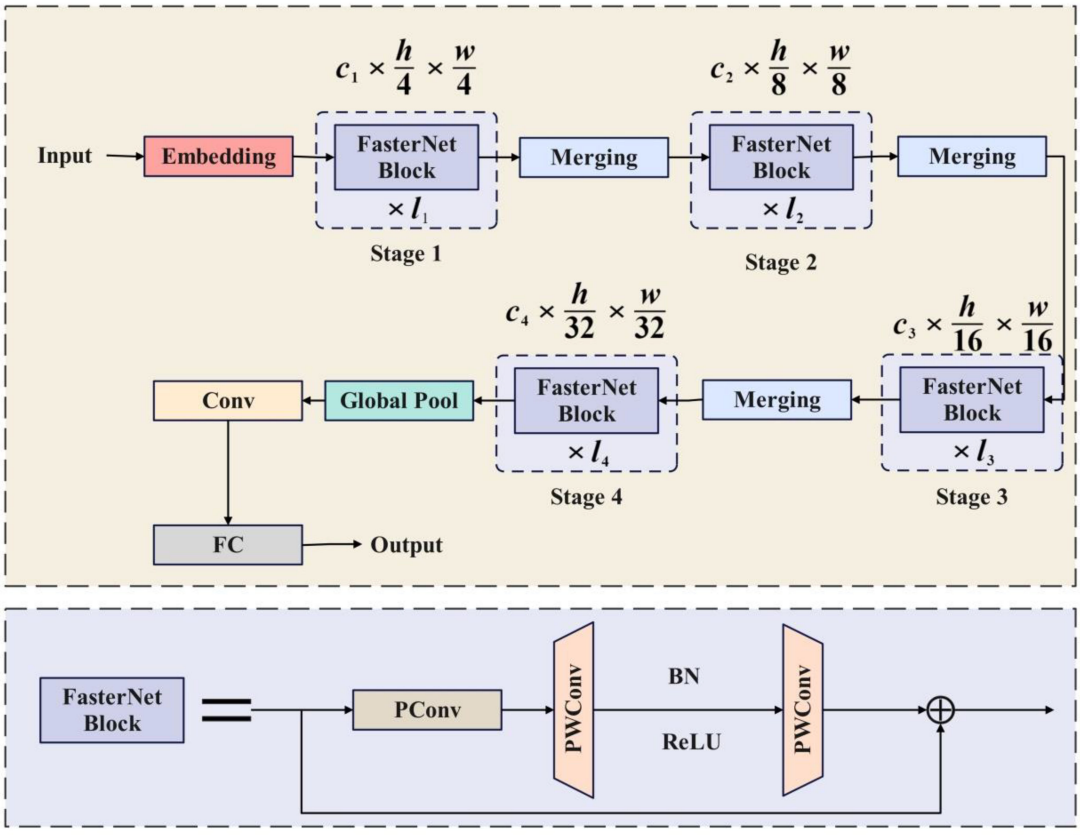


FIGURE 9
FasterNet architecture diagram.

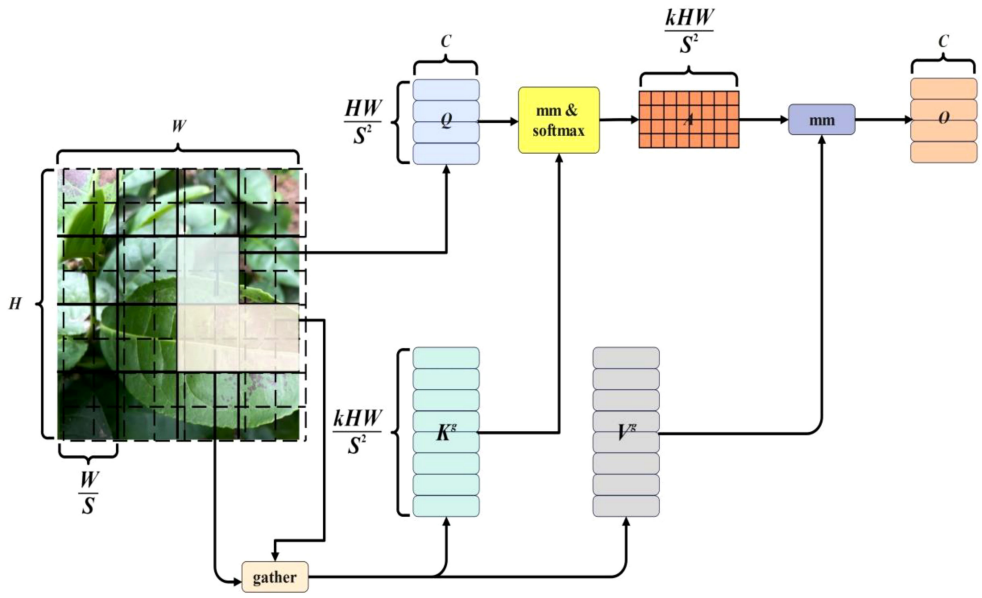


FIGURE 10
Bi-level routing attention mechanism.

3.1 Training process and analysis

The loss function (Wen et al., 2021; Ali et al., 2023) is an important indicator that measures the difference between the predicted results and the actual results of a model. A smaller value of the loss function indicates a better performance of the model, as it means the predicted results are closer to the actual results. As shown in Figure 11, during the initial stage of training, BRA-YOLOv7 exhibits a fast descent in the loss function. However, after 50 epochs, the descent speed slows down and the oscillation of the curve becomes more pronounced. As the training continues, the curve gradually flattens, indicating the convergence of the loss function. Eventually, the total loss on the training set stabilizes below 2%, while the total loss on the validation set stabilizes below 8%. By comparing the change in the loss function curves between the original YOLOv7 and the improved YOLOv7, it is evident that the improved YOLOv7 shows significant reductions in the loss of predicted box position, predicted box confidence, and classification. The most significant reduction is observed in the predicted box position loss, which decreases by more than 20% in both the training and testing sets.

The main model performance evaluation metrics used in this article include precision, recall, F1 score, average precision (AP), and mean average precision (mAP), as shown in Equations 14–18 (Lee et al., 2020; Han et al., 2024).

$$Precision = \frac{T_P}{T_P + F_P} \tag{14}$$

$$Recall = \frac{T_P}{T_P + F_N} \tag{15}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{16}$$

$$AP = \int_0^1 Precision(Recall) dRecall \tag{17}$$

$$mAP = \frac{\sum_{i=1}^{C} AP(i)}{C} \tag{18}$$

Where T_P represents the number of test images in the tea disease category that are correctly identified by the model as belonging to that category, F_P represents the number of test images in other categories of tea diseases that are incorrectly identified by the model as belonging to the current category, F_N represents the number of test images in the current category of tea diseases that are incorrectly identified by the model as belonging to other categories, and C represents the number of categories of tea diseases in the test set.

From the perspective of prediction results, precision is a metric used for statistics. It reflects the proportion of samples that are predicted as a certain class and actually belong to that class, which is also known as the 'classification accuracy'. Recall, on the other hand, measures the ability of the model to retrieve samples correctly among all the samples in that class. The balanced score is a comprehensive measure based on precision and recall, using their harmonic mean. As shown in Figure 12, BRA-YOLOv7 has achieved significant improvements in detection performance. Compared to the YOLOv7 model, Precision, Recall, and F1 have improved by 6.37%, 6.14%, and 6.25% respectively.

AP (Average Precision) represents the average accuracy of a specific class at different IOU thresholds. mAP (mean Average

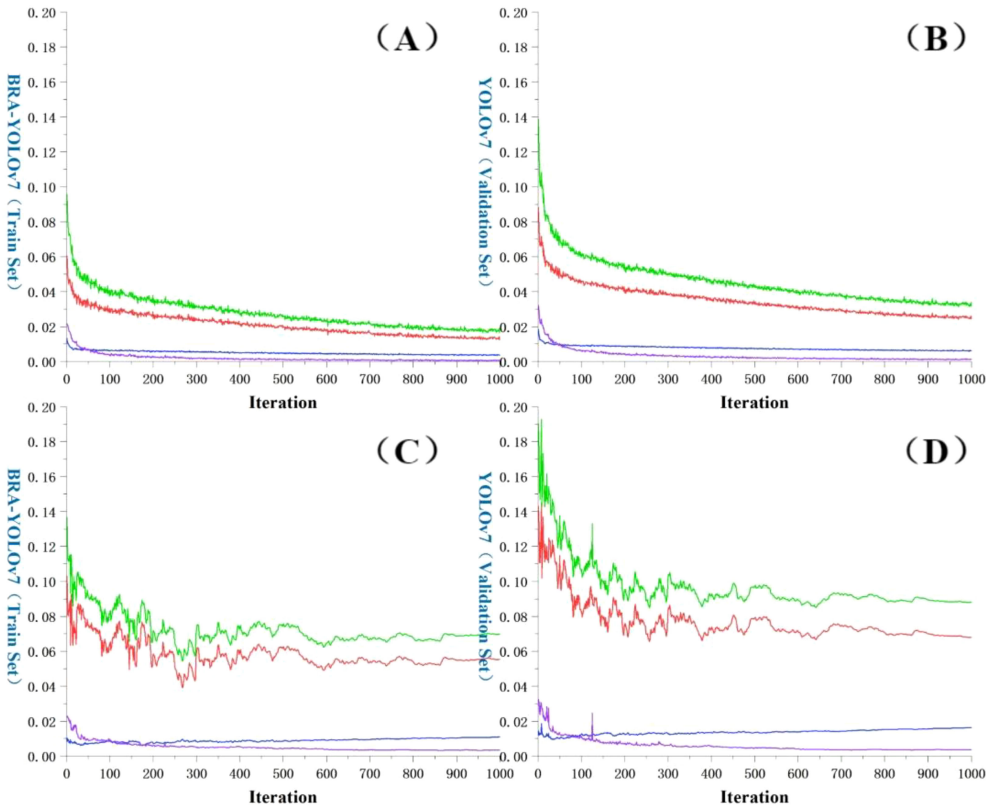


FIGURE 11
Comparison of loss function change curves. **(A)** BRA-YOLOv7 (Training set); **(B)** BRA-YOLOv7 (Validation set); **(C)** YOLOv7 (Training set); **(D)** YOLOv7 (Validation set); Red: Val Box; Blue: Val Objectness; Purple: Val Classification; Green: Total loss value.

Precision) refers to the mean value of AP for various classes. As shown in Figure 13, the BRA-YOLOv7 model demonstrates improvements in tea disease recognition compared to YOLOv7, Faster-RCNN, and SSD. For Single Target Unobstructed recognition, the AP gains are 4.76%, 14.71%, 5.98% respectively. For Single Target Occlusion recognition, the AP gains are 4.72%, 14.4%, 5.63% respectively. For Multiple Target Unobstructed recognition, the AP gains are 5.69%, 15.7%, 7.93% respectively. For a, the AP gains are 5.26%, 15.27%, 8.04% respectively. The overall mAP improvements are 4.71%, 14.69%, 6.95% respectively.

3.2 Model validation experiment

In order to further verify the advantages of the improved model in this study, different lighting intensities were used to detect and identify Tea blight disease, Tea red star disease, Tea white star disease, and Tea wheel spot disease under the conditions of single-target and multi-target with and without occlusion. To ensure the reliability of the results, BRA-YOLOv7, YOLOv7, YOLOv8 (Tian et al., 2022), Faster-RCNN (Cheng and Li, 2023), and SSD (Wang et al., 2023) networks were trained and tested using the same external validation set, while the platform configurations for training were also kept consistent. The final comparison results are shown in Figure 14. A represents Tea blight disease, B represents Tea red star disease, C represents Tea white star disease, and D represents Tea wheel spot disease.

In the test, the four models can successfully detect single-object occlusion and multi-object occlusion in both strong and decreasing light conditions. It is observed that the confidence level decreases as the light intensity decreases, indicating that light intensity has an impact on the model's detection. Among the models, BRA-YOLOv7 and YOLOv7 exhibit the highest confidence in the detection results. The BRA-YOLOv7 model can address the issue of disease localization deviation and avoid repeated detection, showing an average confidence improvement of over 3% compared to the original YOLOv7 model. In the case of multi-object occlusion, the Faster-RCNN model has the lowest confidence in the detection results, leading to missed detections and incorrect recognition. Although SSD can correctly recognize tea diseases, its model accuracy is relatively low. Overall, BRA-YOLOv7 performs better than the other three models in detecting small target diseases.

Table 2 presents a comparison of external parameters for five models in this experiment, including mAP value, floating-point operation count (FLOPs), and frames per second (FPS) during external validation. After incorporating FasterNet, dynamic sparse attention mechanism, and MPDIoU loss function, this study reduced the floating-point operation count by 15.5G compared to the original model, increased the FPS by 5.51% compared to YOLOv7, and

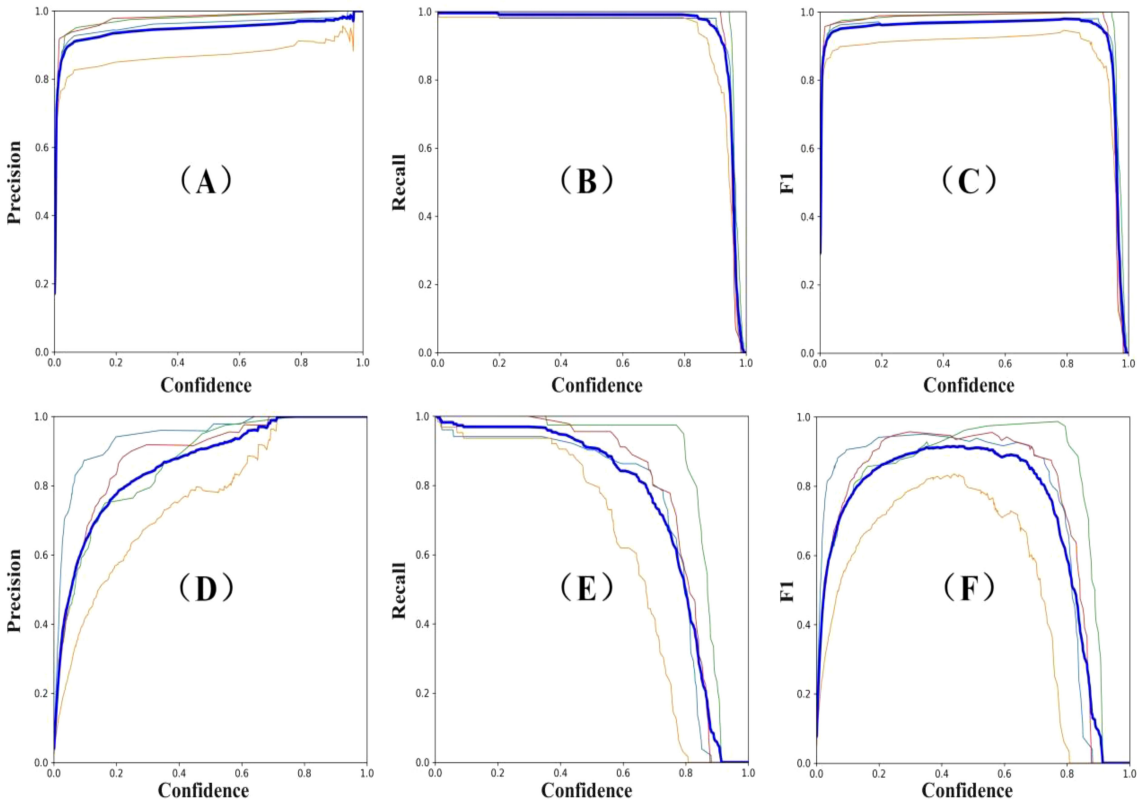


FIGURE 12
Curves depicting changes in accuracy, recall rate, and balanced score. **(A)** YOLOv7 precision; **(B)** YOLOv7 recall; **(C)** YOLOv7 F1 score; **(D)** BRA-YOLOv7 precision; **(E)** BRA-YOLOv7 recall; **(F)** BRA-YOLOv7 F1 score. Different colored thin lines represent the values for Tea Cloud Spot Blight, Tea Red Spot Disease, Tea White Star Disease, and Tea Leaf Spot Disease, respectively. The thick blue line indicates the average value of these four diseases.

improved the mAP value by 4.2% compared to YOLOv7. Overall, BRA-YOLOv7 outperforms the original YOLOv7, YOLOv8, Faster-RCNN, and SSD in terms of detection accuracy and speed. It provides support for the intelligent recognition of edge devices and tea plantation drones in future deployments.

3.3 Ablation experiment

To verify the effectiveness of different improvement modules in the Neck layer of the YOLOv7 model proposed in this article, in the same platform and parameter settings, ablation experiments were conducted on the dataset set to compare the detection accuracy of the BRA-YOLOv7 model with the RFE-YOLOv7 (Tian and Tian, 2023), FRCB-YOLOv7 (), and LW-YOLOv7; () models. The experimental results are shown in Table 3.

From Table 3, it can be seen that in terms of detection speed performance, there is not much difference between BRA-YOLOv7, RFE-YOLOv7, and FRCB-YOLOv7. However, in comparison to RFE-YOLOv7 and FRCB-YOLOv7, the BRA-YOLOv7 model has improved mAP values by 15.46% and 6.416% respectively. The higher mAP values of BRA-YOLOv7 compared to the other two methods demonstrate the effectiveness of this approach. The ablative experiments confirmed the effectiveness of the proposed improvement strategy relative to YOLOv7. Therefore, considering the detection accuracy, memory, and runtime requirements under the same experimental conditions, the BRA-YOLOv7 algorithm was selected for further research.

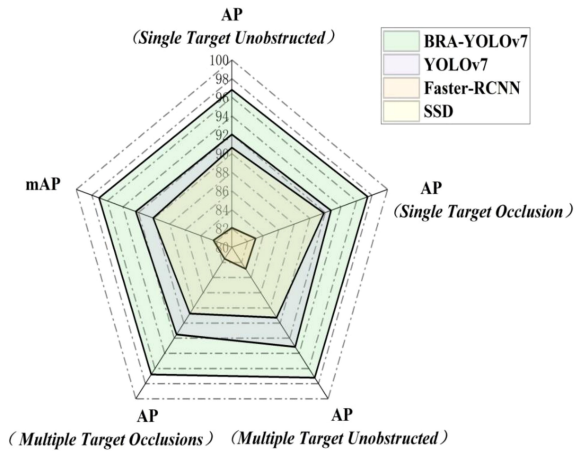


FIGURE 13
Comparison of AP and mAP for different models.

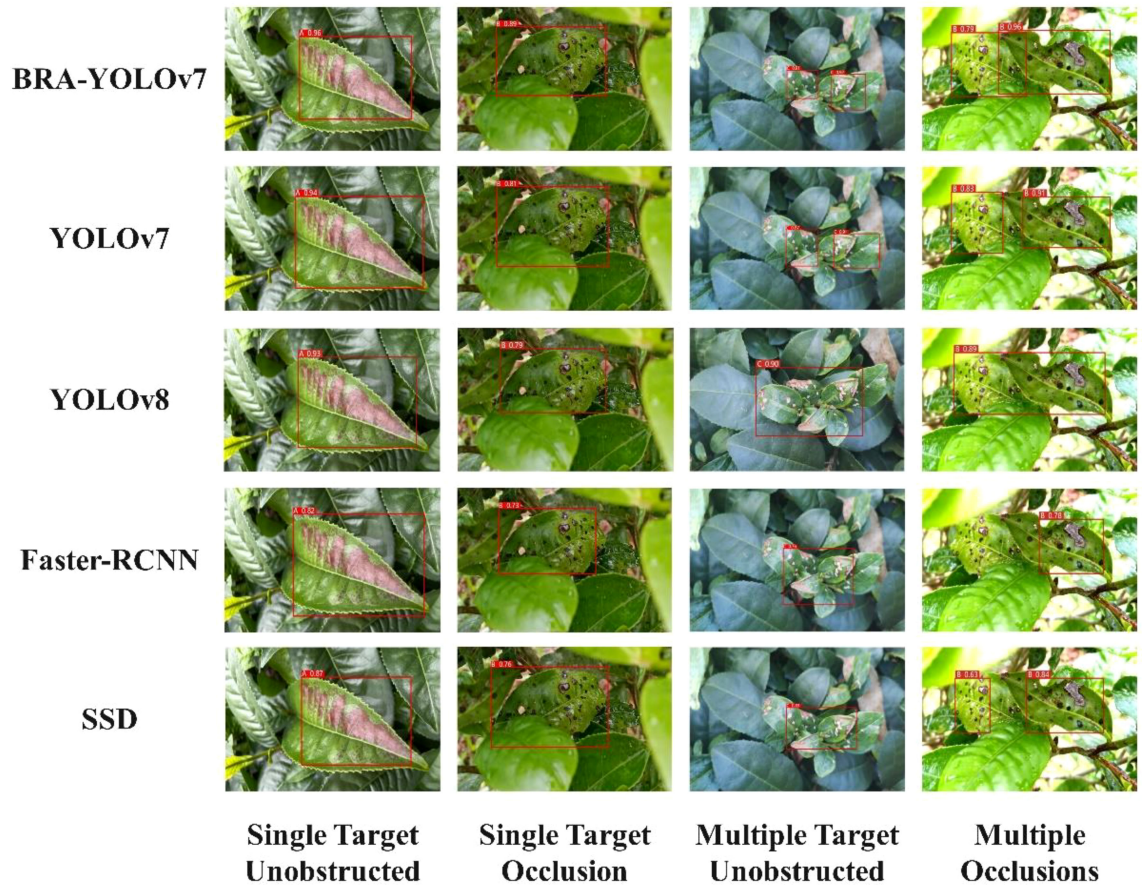


FIGURE 14
Comparison of detection results for different models.

4 Discussion

4.1 Impact of MPDIou on YOLOv7 network

Localization is an important part of object detection, usually achieved through bounding box regression. When training deep models for object detection and instance segmentation, we found that the same disease exhibits similar shape and size characteristics, making MPDIoU more suitable for measuring bounding box similarity. Therefore, this study combines horizontal rectangle geometry features and proposes a new MPDIoU loss function based on minimum point distance. It overcomes the limitations of common loss functions such as CIoU, DIoU, and EIoU. It can still converge when the width and height values are different, and its convergence speed is higher than the CIoU in the YOLOv7 network. This not only simplifies the computation process to a certain extent and improves the model's convergence speed, but also makes the regression results more accurate.

4.2 Influence of PConv and FasterNet on YOLOv7 network

In order to reduce the complexity of the model and achieve faster running speed for the YOLOv7 model, the FasterNet block is introduced in combination with partial convolution (PConv). This

TABLE 2 External validation parameters for comparing models.

Model	mAP/%	FLOPs/G	FPS/Hz
BRA-YOLOv7	94.9	89.7	46.08
YOLOv7	90.7	105.2	43.67
YOLOv8	90.4	165.7	36.71
Faster-RCNN	81.3	346.6	7.03
SSD	88.6	285.4	18.97

TABLE 3 Comparison of ablation results.

Model	P%	R%	mAP@0.5%	F1	FPS
BRA-YOLOv7	90.1	92.3	93.46	91.17	64
RFE-YOLOv7	81.9	80.2	78.0	81.04	69
FRCB-YOLOv7	86.7	83.2	87.3	84.91	74
LW-YOLOv7	89.3	85.5	93.2	87.36	90

allows for maintaining high FLOPS and low FLOPs, utilizing the redundancy in feature maps, and systematically applying conventional convolution (Conv) only on a portion of input channels to extract spatial features, while keeping the rest of the channels unchanged. This helps to reduce information redundancy and facilitate information aggregation. The YOLOv7 model improves detection speed by incorporating the FasterNet Block module into the backbone network.

4.3 The impact of dual-path routing attention mechanism on the YOLOv7 network

Traditional attention mechanisms require computing pairwise interactions between tokens in all spatial positions, resulting in significant computational and memory costs. Therefore, they excel in capturing long-range object detection. However, in the case of disease object detection, it is often difficult to obtain complete features due to overlapping occlusions and smaller disease objects, leading to potential omissions and recognition errors. With the proposed Dual-route Attention mechanism, by leveraging BiFormer's ability to adaptively focus on a small subset of relevant tokens without interference from irrelevant tokens, it enables more flexible computation allocation and enhances content-awareness.

5 Conclusions

This article presents an improved BRA-YOLOv7 algorithm for tea disease target detection in complex scenes. It introduces PConv and FasterNet to replace the original backbone network structure, improving floating point operation efficiency and detection speed. Additionally, a dual-layer route attention mechanism is utilized to filter out irrelevant key-value pairs at the coarse region level, making use of sparsity to save computation and memory. Lastly, a more efficient bounding box loss function called MPDIou is introduced to accelerate model convergence. The experimental results show that:

1. BRA-YOLOv7 network has a total loss stable below 2% on the training set and below 7% on the validation set, which is a more than 2% decrease compared to the original YOLOv7 network. Additionally, in the improved network, there are significant decreases in bounding box position loss, bounding box confidence loss, and classification loss. Among them, the decrease in bounding box position loss is the most significant, with a decrease of over 20% in both the training and testing sets.
2. From the perspective of detection performance, BRA-YOLOv7 has achieved effective improvement in accuracy while reducing the number of parameters. Compared to the YOLOv7 network, the accuracy of BRA-YOLOv7 has improved by 6.37%, the recall rate has improved by 6.14%, and the balanced score has increased by 6.25%. In addition, BRA-YOLOv7 has improved the average precision (AP) of four types of diseases by 4.76%, 4.72%, 5.69%, and 5.26% respectively, resulting in an overall mAP improvement of 4.71%.
3. After external data verification, BRA-YOLOv7 network reduces floating-point operations by 15.5G compared to YOLOv7. The FPS is improved by 5.51% compared to the original model, and the mAP value in actual detection is increased by 4.2%.

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

RY: Data curation, Software, Visualization, Writing – original draft, Writing – review & editing. QG: Data curation, Methodology, Resources, Writing – review & editing. TL: Supervision, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

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, J., Anuran, M., and Devavrat, S. (2023). *Shah federated optimization of smooth loss functions* (IEEE Transactions on Information Theory). doi: 10.1109/tit.2023.3317168

Bai, B., Wang, J., Li, J., Long, Y., Wen, J., and Han, Y. (2024). T-YOLO: a lightweight and efficient detection model for nutrient buds in complex tea-plantation environments. *J. Sci. Food Agric.* 104, 5698–5711. doi: 10.1002/jsfa.v104.10

Chen, J., Kao, S., He, H., Zhuo, W., Wen, S., Lee, C., et al. (2023). "Run, don't walk: chasing higher FLOPS for faster neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12021–12031. doi: 10.1109/cvpr52729.2023.01157

Cheng, Z., and Li, Y. (2023). Improved YOLOv7 algorithm for detecting bone marrow cells. *Sensors*. 23 (17), 7640. doi: 10.3390/s23177640

Gao, P., Huang, H., and Fan, Y. (2023). Interactive speech emotion recognition using fused coordinates and multi-head attention mechanism. *Comput. Appl.*, 1–7.

Halstead, M., McCool, C., Denman, S., Perez, T., and Fookes, C. (2018). Fruit quantity and ripeness estimation using a robotic vision system. *IEEE Robot Autom. Let* 3, 2995–3002. doi: 10.1109/lra.2018.2849514

Han, R., Shu, L., and Li, K. (2024). *A method for plant disease enhance detection based on improved YOLOv8.2024 IEEE 33rd international symposium on industrial electronics (ISIE)* (IEEE), 1–6.

Hong, H., Jiang, S., Li, H., Du, G., Sun, Y., Tao, H., et al. (2020). DeepHiC: A generative adversarial network for enhancing Hi-C data resolution. *PloS Comput. Biol.* 16, e1007287. doi: 10.1371/journal.pcbi.1007287

Hossain, M. S., Mou, R. M., Hasan, M. M., Chakraborty, S., and Abdur Razzak, M. (2018). "Recognition and detection of tea leaf 's diseases using support vector machine," in *Proceedings of the 2018 IEEE 14th international colloquium on signal processing and its application, CSPA* (Institute of Electrical and Electronics Engineers Inc), 150–154.

Hu, G., Wei, K., Zhang, Y., Bao, W., and Liang, D. (2021). Estimation of tea leaf blight severity in natural scene images. *Precis. Agric.* 22, 1239–1262. doi: 10.1007/s11119-020-09782-8

Jiang, M., Chin, K.-W., He, T., He, T., Soh, S ., and Wang, L. (2023). *Joint link scheduling and routing in two-tier RF-energy-harvesting ioT networks* (IEEE Internet of Things Journal). doi: 10.1109/jiot.2021.3085862

Jintasuttisak, T., Edirisinghe, E., and Elbattay, A. (2022). Deep neural network based date palm tree detection in drone imagery. *Comput. Electron. Agric.* 192, 106560. doi: 10.1016/j.compag.2021.106560

Kwan-Wu, C., Wang, L., and Sieteng, S. (2016). *Joint routing and links scheduling in two-tier multi-hop RF-energy harvesting networks* (IEEE Communications Letters). doi: 10.1109/lcomm.2016.2590463

Lee, S. H., Lin, S. R., and Chen, S. F. (2020). Identification of tea foliar diseases and pest damage under practical field conditions using a convolutional neural network. *Plant Pathol.* 69, 1731–1739. doi: 10.1111/ppa.13251

Li, H., Shi, H., Du, A., Mao, Y., Fan, K., Wang, Y., et al. (2022). Symptom recognition of disease and insect damage based on Mask R-CNN, wavelet transform, and F-RNet. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.922797

Li, Y., Liu, W., and Huang, D. (2023). Image denoising network model with fused multi-head attention mechanism. *Comput. Sci.* 50, 326–333.

Lin, W., Zhang, J., and He, N. (2023). Real-time detection method for larch tree insect damage based on improved YOLOv4. *Trans. Chin. Soc. Agric. Machinery* 54, 304–312, 393. doi: 10.6041/j.issn.1000-1298.2023.04.031

Lv, Z., Zhang, F., Wei, X., Huang, Y., Li, J., and Zhang, Z. (2023). Collaborative recognition of tomato flowers and fruits in a greenhouse using an enhanced combination of YOLOX-ViT. *J. Agric. Eng.* 39, 124–134.

Ma, S., and Xu, Y. (2023). The bounding box regression loss function for minimum point distance and its application. *Small Microcomputer Syst.*, 1–8.

Ma, S., and Xu, Y. (2023). *MPDIoU: A loss for efficient and accurate bounding box regression. arXiv - CS - computer vision and pattern recognition*. arxiv-2307.07662. doi: 10.48550/arxiv.2307.07662

Rajathi, N., and Parameswari, P. (2022). Early stage prediction of plant leaf diseases using deep learning models. *Comput. Vis. Mach. Learn.Agric*, 245–260. doi: 10.1007/978-981-16-9991-7-15

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 (CVPR)*, 779–788. doi: 10.1109/cvpr.2016.91

Soeb, M. J. A., Jubayer, M. F., Tarin, T. A., Mamun, M. R. A., Ruhad, F. M., Parven, A., et al. (2023). Tea leaf disease detection and identification based on YOLOv7 (YOLO-T). *Sci. Rep.* 13, 6078. doi: 10.1038/s41598-023-33270-4

Sun, C., Shao, Y., Zhou, H., Xu, J., and Xu, D. (2020). Research progress on six important leaf diseases of tea plants. *Tea* 46, 71–76. doi: 10.3969/j.issn.0577-8921.2020.02.002

Sun, Y., Jiang, Z., Zhang, L., Dong, W., and Rao, Y. (2018). SLIC_SVM based leaf diseases saliency map extraction of tea plant. Comput. *Electron. Agric.* 157, 102–109. doi: 10.1016/j.compag.2018.12.042

Sun, Y., Wu, F., Guo, H., Li, R., Yao, J., and Shen, J. (2023). TeaDiseaseNet: multi-scale self-attentive tea disease detection. *Front. Plant Sci.* 14, 1257212. doi: 10.3389/fpls.2023.1257212

Tian, L., Zhang, H., Liu, B., Zhang, J., Duan, N., Yuan, A., et al. (2022). VMF-SSD: A Novel v-space based multi-scale feature fusion SSD for apple leaf disease detection. *IEEE/ACM Trans. Comput. Biol. Bioinf.* 20, 2016–2028.

Tian, Y., and Tian, X. (2023). A new lunar dome detection method based on improved YOLOv7. *Sensors*. 23 (19), 8304. doi: 10.3390/s23198304

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassanien, A. E., and Pandey, H. M. (2020). An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. *Comput. Electron. Agric.* 175, 105456. doi: 10.1016/j.compag.2020.105456

Wang, Y., Xiao, M., and Wang, S. (2023). Detection of famous tea buds based on improved YOLOv7 network. *Agriculture*. 13 (6), 1190. doi: 10.3390/agriculture13061190

Wen, C., Yang, X., Zhang, K., and Zhang, J. (2021). Improved loss function for image classification. *Compu. Intell. Neurosci.* 2021, 1–8. doi: 10.1155/2021/6660961

Xu, G., Zhang, Y., and Lai, X. (2020). Tea leaf bud image recognition method based on Faster R-CNN deep network. *Optoelectronics·Laser* 31, 1131–1139. doi: 10.16136/j.joel.2020.11.0164

Xu, W., and Jeongyoung, S. (2021). ICIoU: improved loss based on complete intersection over union for bounding box regression. *IEEE Access*. doi: 10.1109/access.2021.3100414

Xue, Z., Xu, R., Bai, D., and Lin, H. (2023). YOLO-tea: A tea disease detection model improved by YOLOv5. *Forests* 14, 415. doi: 10.3390/f14020415

Zhang, F., Sun, H., Xie, S., Dong, C., Li, Y., Xu, Y., et al. (2023). A tea bud segmentation, detection and picking point localization based on the MDY7-3PTB model. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1199473

Zhang, Q., Li, Q., Hu, J., and Xie, X. (2022). Real-time detection of navel orange fruits based on improved algorithm of PP-YOLO. *J. Beijing Union Univ.* 36, 58–66. doi: 10.16255/j.cnki.ldxbz

Zhao, M., Yang, G., Wang, J., and Gong, Z. (2023). Improved real-time detection algorithm for safety helmet using modified YOLOv7-tiny. *Radio Eng.* 53 (08), 1741–1749. doi: 10.3969/j.issn.1003-3106.2023.08.001

Zhu, L., Wang, X., Ke, Z., Zhang, W., and Lau, R. (2023). *BiFormer: vision transformer with bi-level routing attention[C]//2023 IEEE conference on computer vision and pattern recognition*. (Vancouver: IEEE Press), 10323–10333.arxiv-2303.08810

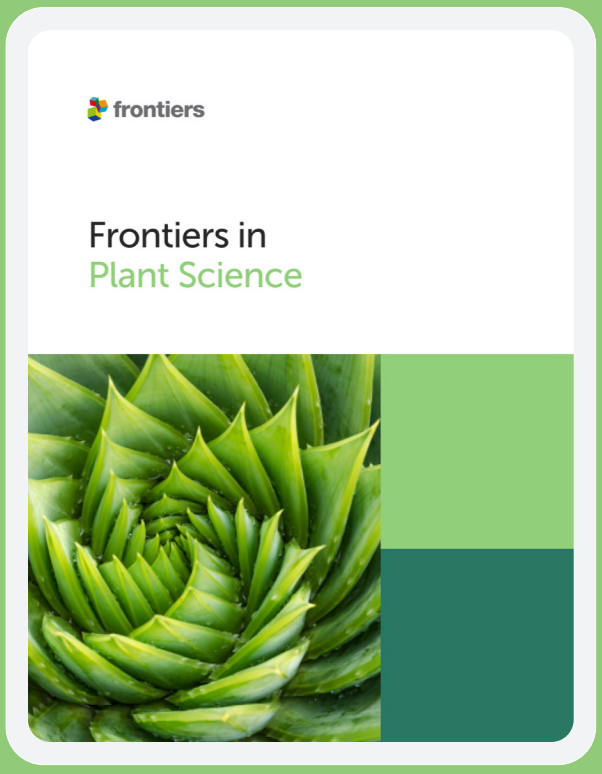
Frontiers in
Plant Science

Cultivates the science of plant biology and its applications

The most cited plant science journal, which advances our understanding of plant biology for sustainable food security, functional ecosystems and human health.

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