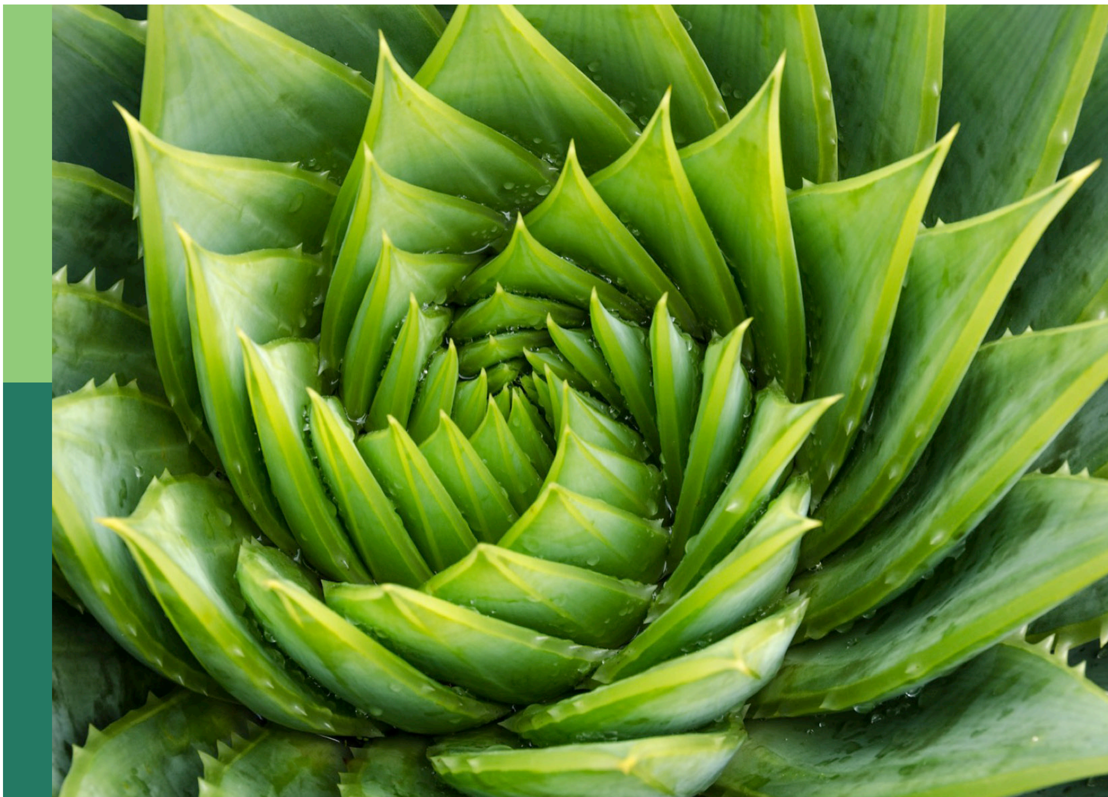
frontiers | Research Topics

Artificial intelligence -of- things (AIoT) in precision agriculture

Edited by
Yaqoob Majeed, Longsheng Fu and Long He

Published in
Frontiers in Plant Science



ISSN 1664-8714
ISBN 978-2-8325-4431-0
DOI 10.3389/978-2-8325-4431-0

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

Artificial intelligence-of-things (AIoT) in precision agriculture

Topic editors

Yaqoob Majeed – University of Agriculture, Faisalabad, Pakistan
Longsheng Fu – Northwest A&F University, China
Long He – The Pennsylvania State University (PSU), United States

Citation

Majeed, Y., Fu, L., He, L., eds. (2024). *Artificial intelligence-of-things (AIoT) in precision agriculture*. Lausanne: Frontiers Media SA.
doi: 10.3389/978-2-8325-4431-0

Table of contents

Frontiers in Plant Science

TYPE Editorial
PUBLISHED 26 January 2024
DOI 10.3389/fpls.2024.1369791

OPEN ACCESS

EDITED AND REVIEWED BY
Roger Deal,
Emory University, United States

*CORRESPONDENCE
Yaqoob Majeed
yaqoob.majeed@uaf.edu.pk
yaqoob.majeed@tamu.edu

RECEIVED 12 January 2024
ACCEPTED 22 January 2024
PUBLISHED 26 January 2024

CITATION
Majeed Y, Fu L and He L (2024) Editorial: Artificial intelligence-of-things (AIoT) in precision agriculture.
Front. Plant Sci. 15:1369791.
doi: 10.3389/fpls.2024.1369791

Editorial: Artificial intelligence-of-things (AIoT) in precision agriculture

Yaqoob Majeed[1,2]*, Longsheng Fu[3] and Long He[4]

[1]Department of Food Engineering, University of Agriculture Faisalabad, Faisalabad, Punjab, Pakistan, [2]Texas A&M AgriLife Research, Texas A&M University System, Dallas, TX, United States, [3]College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling, Shaanxi, China, [4]Department of Agricultural and Biological Engineering, Pennsylvania State University, University Park, PA, United States

KEYWORDS

precision agriculture, internet-of-things (IoT), artificial intelligence, deep learning, digital agriculture, machine vision

Editorial on the Research Topic
Artificial intelligence-of-things (AIoT) in precision agriculture

Precision agriculture is becoming critically important for sustainable food production to meet the growing food demand. In recent decades, technical advances in AI (artificial intelligence) and IoT (internet-of-things) can help solve various agricultural field problems and optimize resource utilization (e.g. water, pesticide, fertilizer, seed, energy), improve production management and productivity, and reduce labor dependency. AI and IoT-enabled applications are increasingly implemented for precision agriculture applications such as crop growth monitoring, weed removal control, pest and disease detection, planting, crop yield estimation, targeted spraying and pollination, smart irrigation and nutrient management, field analysis, and plant phenotyping. For example, IoT-based applications using machine learning and deep learning models are widely used to recognize fruits, vegetables, weeds, pests, and diseases, and measure soil quality and nutrients. Such information helps inform better crop management practices. Despite the progress of AI and IoT technologies in precision agriculture, the combined use of these technologies in the form of AIoT are still in early stages with numerous challenges in the form of data acquisition and connectivity, and optimization of AI algorithms based on edge computing processing capabilities that still need to be addressed.

This Research Topic focuses on the recent advancement in the area of AI and IoT applications on precision agricultural technologies for both field and specialty crops. This Research Topic attracted nine research articles and three review articles. These articles reveal the research advancements and trends of applied machine learning and deep learning techniques for various precision agriculture applications.

Robotic harvesting plays an important role in addressing the labor shortage problems for manual labor-intensive and time-sensitive harvesting operations. For example, Sun et al. propose the YOLO-P to detect the pears for robotic harvesting in natural orchard environment. They propose the shuffle block integrated with convolutional block attention module (CBAM) as the backbone of YOLOv5 network. A total of 5,257 images consisting of various backgrounds and illumination conditions were used to train and test the proposed approach. Different ablation experiments were performed to check the robustness and generalization and obtained the 0.961 F1-score with 32 FPS (frames per second). To facilitate autonomous driving of robot and roadside fruit harvesting, Zhou et al. proposed the framework for synchronous road extraction and roadside fruit recognition. Gray factor optimization approach was adopted to extract the unstructured roads from images while YOLOv7 was employed to detect the wine grapes. The proposed synchronous approach helped to increase fruit detection by 23.84%.

In another study, Tang et al. estimated the tree-level almond yield using aerially captured multispectral images and convolutional neural networks. They used approximately 2000 almond trees for the yield monitoring. Multispectral aerial images were collected at a height of 6,000 ft with 0.3 m spatial resolution. Then, convolutional neural network (CNN) with spatial attention module was proposed to estimate the yield estimation at tree-level. Their proposed approach achieved the R^2 and RMSE (root mean square value) of 0.96 and 6.6%, respectively. Similarly, Ren et al. introduce the mobile robotic platform for indoor farming to monitor strawberry yield. They first developed the autonomous mobile robot platform (AMR) that uses the AprilTag and inertial navigation to autonomously navigate the structural environment of indoor farms. Then, they used the multilayer perception robot (MPR), mounted on ARM, to collect the temporal-spatial data of the strawberry plants within the strawberry indoor farm. Their MPR achieved the positioning accuracy of 13.0 mm while navigating the plant factory with 6.26% error rate in yield monitoring performance.

Precision pest management is another area in precision agriculture which involves accurate pest detection and identification for the precise pesticide applications. For example, Peng et al. employed an ensemble learning technique to fuse the selective kernel unit, representative batch normalization module, and ACON activation with the Dense-Net-121 networks, naming it MADN, to detect and identify the crop pests. Their proposed approach helped to achieve F1-score of 0.7528 in identifying the pests.

To optimize coconut breeding, Liu et al. introduced a non-destructive approach to segment the internal organs of coconuts using Computed Tomography (CT) scanning and semantic segmentation. They scanned the coconut during different stages using the CT scan and constructed the CIDCO dataset. Then DeepLabv3+ based semantic segmentation was employed by introducing dense atrous spatial pyramid pooling and CBAM modules. Their improved model helped to achieve F1-score of 0.905 to segment the internal organs of coconuts. Similarly, non-destructive and automatic detection of defective kiwifruit is critically important to maintain the postharvest quality of kiwifruits and for consumer acceptability. To address this issue, Wang et al. focused on detecting the defective kiwifruits for grading lines by employing YOLOv5. They constructed a multiple-defect kiwifruit dataset consisting of healthy, leaf-rubbing, damaged, healed cuts or scarred, and sun-burn kiwifruits. Then, spatial-depth and depth-wise separable convolutional modules were combined with YOLOv5 to improve the detection performance of the defective kiwifruits. Their approach helped to achieve an average detection accuracy of 97.7% with 8.0 ms detection time.

It has been always a challenge for dataset availability and its manual labeling to train AI based algorithms to solve the specific precision agriculture application. To address this problem, Wang et al. introduce a deep reinforcement learning based augmentation framework for the leaf rust images. Their proposed approach consists of Deep Q-Learning (DQN) for selecting optimal augmentation approach based on individual image, extracting geometric and pixel indicators, and DeepLabv3+ to authenticate augmented image and feedback the rewards. Experimental results showed that the proposed approach helped to achieve Intersection-over-Union (IoU) of 0.8426 in correctly classifying leaf rust spots compared to the union of expected and predicted rust spots.

Measurement of plant phenotypic traits is critical in selecting the high-yield crop varieties and timely identifying the need in actions for optimal plant growth. To measure the soybean plant phenotyping traits, He et al. proposed a generalized regression neural network based approach. First, SfM (structure from motion) algorithm was used to reconstruct the soybean plants. Then, different filtering (lowpass filter and gaussian filter) and Laplacian smoothing methods were used to segment different parts of soybeans (e.g. plants, stem, and leaves). Ultimately, a generalized regression neural network was employed to measure the phenotypic traits of the soybeans. Results indicated that their proposed approach helped to achieve R^2 of 0.9775, 0.9785, and 0.9487 for measuring the plant height, lead length, and leaf width, respectively compared to ground truth measurements.

In addition to the above-mentioned studies, there are further areas in which AI-assisted technologies could be used for precision agriculture applications. For example, Nawaz et al. reviewed the latest trends in applying data processing and deep learning algorithms for remote sensing data. Furthermore, Estrada et al. explored and reviewed machine learning applications for remote forestry health assessment. Similarly, Johnson & Cheein presented a comprehensive review on the use of mechatronics, AI and IoT applications for potato harvesting.

With the papers published in this Research Topic ranging from different precision agriculture applications and covering latest advancements in the AI application to solve various agricultural challenges, we hope readers will gain insights into the state-of-the-art developments in rapidly growing precision and digital agriculture domain and will provide further opportunities for scientists and industries to take on the collective challenges faced by this sector. The papers published in this Research Topic proved the critical role of AI and IoT applications to address global food security issues and meet the sustainable agriculture goals in the context of declining and aging agricultural labor. However, more studies will be needed with continuous innovations, and collective efforts from scientists and industries working in the precision and digital agriculture domain.

Author contributions

YM: Writing – original draft, Writing – review & editing. LF: Writing – review & editing. LH: Writing – review & editing.

Conflict of interest

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

Publisher's note

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

TYPE Systematic Review
PUBLISHED 18 November 2022
DOI 10.3389/fpls.2022.1041514

Check for updates

OPEN ACCESS

EDITED BY
Nieves Goicoechea,
University of Navarra, Spain

REVIEWED BY
Huan Yu,
Chengdu University of Technology, China
Sijia Yu,
The State University of New Jersey - Busch Campus, United States

*CORRESPONDENCE
Jingbing Li
jingbingli2008@hotmail.com

This article was submitted to Sustainable and Intelligent Phytoprotection, a section of the journal Frontiers in Plant Science

SPECIALTY SECTION
RECEIVED 13 September 2022
ACCEPTED 07 October 2022
PUBLISHED 18 November 2022

CITATION
Nawaz SA, Li J, Bhatti UA, Shoukat MU and Ahmad RM (2022) AI-based object detection latest trends in remote sensing, multimedia and agriculture applications.
Front. Plant Sci. 13:1041514.
doi: 10.3389/fpls.2022.1041514

AI-based object detection latest trends in remote sensing, multimedia and agriculture applications

Saqib Ali Nawaz[1,2], Jingbing Li[1,2]*, Uzair Aslam Bhatti[1,2], Muhammad Usman Shoukat[3] and Raza Muhammad Ahmad[4]

[1]School of Information and Communication Engineering, Hainan University, Haikou, China, [2]State Key Laboratory of Marine Resource Utilization in the South China Sea, Hainan University, Haikou, China, [3]School of Automotive Engineering, Wuhan University of Technology, Wuhan, China, [4]College of Cyberspace Security, Hainan University, Haikou, China

Object detection is a vital research direction in machine vision and deep learning. The object detection technique based on deep understanding has achieved tremendous progress in feature extraction, image representation, classification, and recognition in recent years, due to this rapid growth of deep learning theory and technology. Scholars have proposed a series of methods for the object detection algorithm as well as improvements in data processing, network structure, loss function, and so on. In this paper, we introduce the characteristics of standard datasets and critical parameters of performance index evaluation, as well as the network structure and implementation methods of two-stage, single-stage, and other improved algorithms that are compared and analyzed. The latest improvement ideas of typical object detection algorithms based on deep learning are discussed and reached, from data enhancement, *a priori* box selection, network model construction, prediction box selection, and loss calculation. Finally, combined with the existing challenges, the future research direction of typical object detection algorithms is surveyed.

KEYWORDS
deep learning, object detection, transfer learning, algorithm improvement, data augmentation, network structure

1 Introduction

Computer vision, also known as machine vision, uses an image sensor that replaces the human eye to obtain an image of an object, converts the image into a digital image, and uses computer-simulated human discrimination criteria to understand and recognize the image, to analyze the image, and draw conclusions. This technology gradually emerged on the basis of the successful application of remote sensing image processing and medical image processing technology in the 1970s and has been applied in many fields. At present, the application of computer vision technology in agriculture is increasing day by day. Object detection is widely used in different areas of agriculture and getting importance these days in fruits, diseases, and scene classification (Zhang et al., 2020; Bhatti et al., 2021).

The primary goal of this work is to find all of the objects of interest in a specified image with high accuracy and efficiency and to use the rectangular bounding box to determine the spot and size of the detected object, which is connected to object classification, semantic segmentation, and instance. In the process of object detection, due to the different appearance, posture, shape, and quantity of various target objects in the image, as well as the interference of multiple factors such as illumination and occlusion, the target is distorted, and the difficulty of object detection (Chen and Wang, 2014; Bhatti et al., 2019).

Deep learning-based object detection algorithms are mainly divided into traditional and detection algorithms. Traditional detection approaches rely on hand-crafted features and shallow trainable architectures, which are ineffective when creating complicated object detectors and scene classifiers that combine many low-level image features and high-level semantic information. Traditional object detection algorithms mainly include the deformable parts model (DPM) (Dollár et al., 2009), selective search (SS) (Uijlings et al., 2013), Oxford-MKL (Vedaldi et al., 2009), and NLPR-HOGLBP (Yu et al., 2010), etc. Traditional object detection algorithm basic structure mainly includes the following three-part: 1) region selector, first, a sliding window of different sizes and proportions is set for a given image, and the entire image is traversed from left to right and top to bottom to frame a specific part of the image to be detected as a candidate region; 2) feature extraction, extract visual features of candidate regions, such as scale-invariant feature transform (SIFT) (Bingtao et al., 2015), Haar (Lienhart and Maydt, 2002), histogram of oriented gradient (HOG) (Shu et al., 2021) commonly used in face and standard object detection, and other features to extract features for each region; 3) classifier classification, use the trained classifier to identify the target category of the feature, such as the commonly used deformable part model (DPM), adaboot (Viola and Jones, 2001), support vector machines (SVM) (Ashritha et al., 2021) and other classifiers. However, these three parts achieved certain results while exposing their inherent flaws, such as using a sliding window for region selection will result in high time complexity and window redundancy, the uncertainty of illumination change and the diversity of background will result in poor robustness of the guide design feature technique (Cao et al., 2020a), poor generalization, and complex algorithm stages will result in slow detection efficiency and low accuracy (Wu et al., 2021). As a result, classic object detection approaches have struggled to match people's demands for high-performance detection.

However, there are still some complications in applying an object detection algorithm based on deep learning, such as too small detection objects, insufficient detection accuracy, and insufficient data volume. Many scholars have improved algorithms and also formed a review by summarizing these improved methods. Tong et al. (2020) analyzed and outlined the improved techniques from the aspects of multi-scale features, data enhancement and context information but ignored the performance improvement of the feature extraction network for small object detection; moreover, the data enhancement part only considers improving the small object detection performance by increasing the number and type of small targets in the data set, which lacks diversity. Xu et al. (2021) and Degang et al. (2021) respectively introduced and analyzed the typical algorithms of object detection for the detection framework based on regression and candidate window. However, because the optimization scheme of the algorithm is not well classified in the text, they cannot clearly understand when and how to apply the improvement idea to the detection algorithm. The mainstream deep learning object detection algorithms are mainly separated into two-stage detection algorithms and single-stage detection algorithms, as shown in Figure 1.

In Figure 1, the two-stage detection algorithm is based on candidate regions represented by the R-CNN series; the single-stage detection algorithm is a regression analysis-based object detection algorithm defined by YOLO and SSD. This review is based on different object detection techniques approaches, and the main contribution of this paper is as follows:

- Firstly, this review organized the standard data sets and evaluation indicators. The list of datasets and their evaluation methods are in-depth and highlighted from different literature from recent years.
- Secondly, this review paper focused on deep learning approaches for object detection, including two-stage and single-stage object detection algorithms and generative adversarial networks.
- The third part of this paper surveyed the deep learning-based object detection algorithm applications in multimedia, remote sensing, and agriculture. Finally draws a conclusion and some future works.

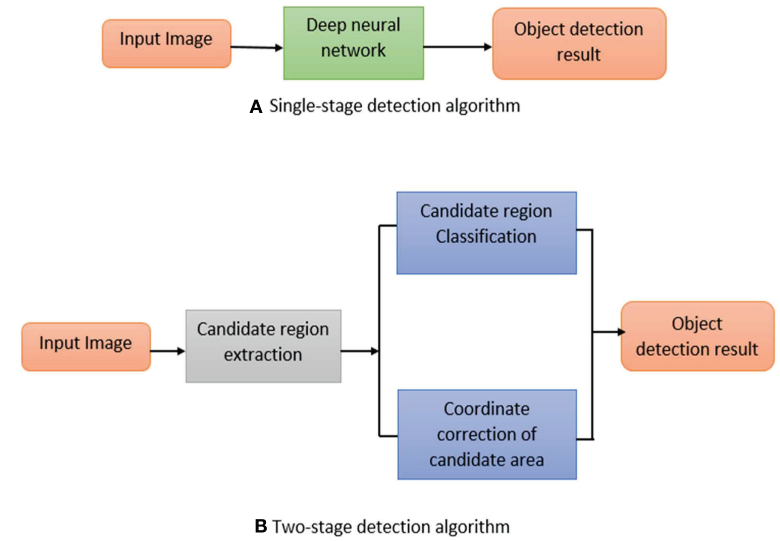

Input Image
Deep neural network
Object detection result
A Single-stage detection algorithm
Candidate region Classification
Input Image
Candidate region extraction
Object detection result
Coordinate correction of candidate area
B Two-stage detection algorithm

FIGURE 1
Object detection method based on deep learning **(A)** Single stage method **(B)** Two stage method.

2 Common data sets and evaluation indicators

This section highlights the datasets used for objects in remote sensing, agriculture, and multimedia applications.

2.1 Common datasets

In the task of object detection, a dataset with strong applicability can effectively test and assess the performance of the algorithm and promote the development of research in related fields. The most widely used datasets for deep learning-based object detection tasks are PASCAL VOC2007 (Ito et al., 2007), PASCAL VOC2012 (Marris et al., 2012), Microsoft COCO (Lin et al., 2014), ImageNet (Deng et al., 2009) and OICOD (Open Image Challenge Object Detection) (Krasin et al., 2017). Different features and quantities of images in datasets are listed in Table 1.

2.2 Evaluation indicators

The act of the object detection algorithm is mainly evaluated by the following parameters: intersection over union (IoU) (Rahman and Wang, 2016), frame per second (FPS), accuracy (A), recall (R), precision (P), average precision (AP), and mean average precision (mAP) (Tong et al., 2020). Where AP consists of the area enclosed by the P-R curve and the coordinates, and mAP is the mean of AP (Kang, 2019; Wang, 2021).

3 Deep learning approaches for object detection in multimedia

3.1 Two-stage object detection algorithm

In two-stage object detection, one branch of object detectors is based on multi-stage models. Deriving from the work of R-CNN, one model is used to extract regions of objects, and a second model is used to classify and further refine the localization of the object. To obtain test results, the two-stage object detection approach primarily uses algorithms such as Selective Search or Edge Boxes (Zitnick and Dollár, 2014) to choose the candidate region (Region Proposal) (Hu and Zhai, 2019) that may include the object detection for the input image, and then categorize and position the candidate region. The R-CNN (Girshick et al., 2014) series, R-FCN (Dai et al., 2016), Mask R-CNN (He et al., 2017), and other algorithms are examples.

3.1.1 OverFeat algorithm

The OverFeat algorithm was proposed by the author in Sermanet et al. (2013), who improved AlexNet. The approach combines AlexNet with multi-scale sliding windows (Naqvi et al., 2020) to achieve feature extraction, shares feature extraction layers and is applied to tasks including image classification, localization, and object identification. On the ILSVRC 2013 (Lin et al., 2018) dataset, the mAP is 24.3%, and the detection effect is much better than traditional approaches. The algorithm has heuristic relevance for deep learning's object

TABLE 1 Comparison of related data sets.

Dataset Name	Quantity	Type	Year	Features
CIFAR-10 (Krizhevsky and Hinton, 2009)	60000	10	2009	Color pictures of everyday things in daily life; take up little storage space; objects detection in images is large; this dataset is often used to measure the classification ability of the model
PASCAL VOC 2007 (Everingham et al., 2010) PASCAL VOC 2012 (Everingham et al., 2015)	9963 11530	20 20	2010 2015	Standardized datasets that can be used for image classification, object detection, and image segmentation; the standardized process makes most of the self-made datasets use this format; most of them are real-world data, which is difficult to detect; it has better image quality and complete Labels are mostly used to evaluate model performance; every image resembles to its annotation file one-to-one, which is easy to manage;
ImageNet (Russakovsky et al., 2015)	14.19 Million	21841	2015	Because this dataset has extremely rich variety information and can contain the underlying features of most detected objects, it is often used as a dataset for pre-training models, which also makes the model extremely challenging in both object detection and object classification.
Microsoft COCO (Lin et al., 2014)	328000	91	2014	The image environment is complex and diverse, which increases the difficulty of detection; in addition to the category and location information of the image, it also contains the scene description of the image; the number of categories is far from the ImageNet, Open Image, and SUN datasets, but this also makes each category more difficult to detect. The larger the number of images contained, the better the detection ability of the model during training.
Open Image (Kuznetsova et al., 2020)	1.9 Million	600	2020	The largest dataset with target location annotations currently available; the annotation information is manually reviewed to ensure accuracy and consistency; The majority of the photographs are complex settings with several objects
Places (Zhou et al., 2017)	2.5 Million	205	2017	The Places dataset is a scene-centric database, and the scene categories in the images represent the scene information of each image
SUN (Xiao et al., 2016)	130519	899	2016	Compared with the Places dataset, it has more scene category information, but the average category of the SUN dataset in each scene is about 80 times different from the Places dataset, resulting in a weaker scene classification ability learned by the model using the SUN dataset; In addition to scene recognition, object recognition under the scene can be performed.

detection algorithm; however, it is ineffective at detecting small objects and has a high mistake rate.

3.1.2 R-CNN algorithm

The convolutional neural network (CNN) to the job of object detection introduced the R-CNN Krizhevsky et al. (2012), a standard two-stage object detection approach. Three modules of deep feature extraction and classification and regression based on CNN:

1. Use a selective algorithm to extract about 2000 regional candidate frames that may contain target objects from the individual image;
2. Normalize the applicant areas scale to a static magnitude for feature mining;
3. Use AlexNet to input the candidate region features into SVM one by one for classification, using Bounding Box Regression and Non-Maximum Suppression (NMS).

The Hinge loss with the L_2 regularization term (Moore and DeNero, 2011) is the loss function of the SVM classification algorithm. The following is the definition of the function form:

$$L_{cls} = c\sum_i \max\left(0,\ 1 - p_i^* \ .\ p_i\right) + \frac{1}{2}w^2 \tag{1}$$

where the proper category of the item is represented by p_i^*, the possibility of the projected object class is represented by p_i, and the index of the mini-batch is denoted by i. To improve the prediction's resilience, the main premise is to penalize the distance variation among the predicted bounding-box and the ground truth. The following is the definition of the function:

$$t_x^* = (x^* - x)/w, \quad t_y^* = (y^* - y)/h$$

$$t_w^* = \log(w^*/w), t_h^* = (h^*/h) \tag{2}$$

$$L_{loc} = \sum_i \left(t_*^i - w_*^T \phi(t^i)\right)^2 \tag{3}$$

where, the true coordinate is $t^* = (x^*,y^*,w^*,h^*)$ the predicted coordinate is $t = (x,y,w,h)$, where (x, y) signifies the coordinate of the box center, (w, h) denotes the width and height of the box. w_*^T is the learned limit, and $\phi(t^i)$ is the feature vector. The regional scores are adjusted and filtered for location regression in a fully connected network (Girshick et al., 2014).

On the ILSVRC2013 dataset, the R-CNN algorithm improves the mAP to 31.4% and 58.5% on the VOC2007 dataset. The performance is better than the typical object detection algorithm. However, the following issues persist:

1. Because every stage must be qualified separately, training involves a multi-stage pipeline that is slow and difficult to optimize.
2. Because CNN features should be derived from each object proposal for each image, training of the SVM classifier and bounding box regressor is time and disk intensive. This is critical for large-scale detection.
3. The test speed is slow, because the CNN structures need to be mined in each test image object proposal, and there is no shared computation.

3.1.3 SPP-Net algorithm

He et al. (2015) presented the Spatial Pyramid Pooling Network (SPP-Net) in 2015 as a solution to the problem that R-CNN pulls features from all candidate regions separately, which takes a lot of time. Between the last convolutional layer and the fully connected layer, SPP-Net adds a spatial pyramid structure, segments the image using numerous standard scales fine-tuners, and fuses the quantized local features to form a mid-level representation. To avoid repetitive feature extraction and break the shackles of fixed-size input, a fixed-length feature vector is built on the feature map, and features are extracted all at once. On the PASCAL 2007 dataset, the SPP-Net algorithm is 24102 times faster than the R-CNN algorithm in detection, and the mAP is increased to 59.2%. However, the following issues want to be addressed:

1. A huge sum of features must be kept, which consumes a lot of space;
2. the SVM classifier is still utilized, which requires a lot of training steps and takes a long time.

3.1.4 Fast R-CNN algorithm

Girshick (2015) introduced the Fast R-CNN technique grounded on bounding box and multi-task loss classification to solve the difficulties of SPP-Net. The algorithm streamlines the SPP layer and creates a single-scale ROI Pooling layer assembly, in which the applicant region of the entire image is tested into a static size, a feature map is created for SVD decomposition, and the Softmax classification score and BoundingBox are obtained *via* the ROI Pooling layer. As follow;

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v) \tag{4}$$

where, $L_{cls}(p,u) = -\log p_u$ computes the log loss for ground truth class u, and p_u is determined from the separate chance dispersal $p = (p_0,\cdot\cdot,p_c)$ over the C+1 outputs from the last FC layer. $L_{loc}(t^u, v)$ is well-clear over the forecast offsets $t^u = (t_x^u, t_y^u, t_w^u, t_h^u)$ and ground-truth bounding-box regression objects $v = (v_x,v_y,v_w,v_h)$, where x, y, w, and h mean the two synchronizes of the box center, width, and height, respectively. To stipulate an object proposal with a log-space height/width change and scale-invariant conversion, each t^u uses the parameter settings (Zitnick and Dollár, 2014). To omit all backdrop RoIs, the Iverson bracket indicator function $[u \geq 1]$ is used. A smooth L_1 loss is used to fit bounding-box regressors in order to give additional robustness against outliers and remove sensitivity in exploding gradients:

$$L_{loc}(t^u, v) = \sum\nolimits_{i\in x,y,w,h} \mathrm{smoth}\ \ L_1(t_i^u - v_i) \tag{5}$$

And

$$\mathrm{smooth}L_1(x) = \begin{cases} 0.5x^2 & if|x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \tag{6}$$

3.1.5 Faster R-CNN algorithm

The employment of candidate region generating methods such as bounding boxes, selective search, and others stymies accuracy progress. Ren et al. (2015) presented Faster R-CNN in 2017 as a solution to this problem and introduced a Region Proposal Network (RPN) to replace the selective search algorithm. Comparing suggestions to reference boxes, regressions toward actual BBs can be accomplished (anchors). Anchors of three scales and three feature ratios are used in the Faster R-CNN. The loss function resembles that of (4);

$$L(p_i, t_i) = \frac{1}{N_{cls}}\sum\nolimits_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum\nolimits_i p_i^* L_{reg}(t_i, t_i^*) \tag{7}$$

where, p_i denotes the likelihood that the i^{th} anchor will be an object. If the anchor is positive, the ground truth label p_i^* is 1, otherwise, it is 0. t_i^* is related to the ground-truth box overlying with a positive anchor, while t_i contains four parameterized coordinates of the predicted bounding box. L_{cls} is a binary log loss, while L_{reg} is a smoothed L_1 loss, both of which are similar to (5). On the PASCAL VOC 2007 dataset, faster R-CNN achieves 73.2% mAP using the VGG-16 backbone network. However, there are still issues:

- The scale chosen by the selection box on the feature map when the anchor mechanism is employed is not adequate for all objects, notably for small object identification;
- Only the last layer of the VGG-16 network is used. The accumulation layer's output features are predicted. The network topographies lose conversion invariance and accuracy after the RoI Pooling layer;

3.1.6 R-FCN algorithm

The idea and performance of the R-CNN series of algorithms determine the milestones of object detection. This series of structures is essentially composed of two subnets (Faster R-CNN adds PRN, which is composed of three subnets), the former subnet is the spine network for feature withdrawal, and the latter subnet is used to complete the classification and localization of object detection. Between the two subnetworks, the RoI pooling layer turns the multi-scale feature map into a static-size feature map, but this step breaks the network's translation invariance and is not favorable to object classification. Using the ResNet -101 He et al. (2016) backbone network, Dai et al. (2016) developed a position-sensitive score map (Position-Sensitive Score Maps) containing object location info in the R-FCN (Region based Fully Convolutional Networks) algorithm.

3.1.7 Mask R-CNN algorithm

MaskR-CNN, proposed by He et al. (2017) is a Faster R-CNN extension that uses the ResNet-101-FPN backbone network. Multi-task loss is combined with segmentation branch loss, arrangement, and bounding box regression loss in Mask R-CNN. A Mask network branch for RoI calculation and division is added to the object classification and bounding box regression to enable real-time object identification and instance segmentation. Lin et al. (2017a) projected the RoIAlign layer to replace the RoI pooling layer and used bilinear difference to plug the pixels of non-integer situations to tackle the problem of rounding the feature map scale in the downsampling and RoI pooling layers. The COCO dataset's mAP has been increased to 39.8% with a detection speed of 5 frames per second. However, meeting real-time criteria for detection speed is still problematic, and the cost of instance segmentation and labeling is too high.

3.1.8 Comparison and analysis

On the COCO dataset, the two-stage object detection uses a cascade structure and has been successful in instance segmentation. Although detection accuracy has improved over time, detection speed has remained poor. On the VOC2007 test set, VOC 2012 test set, and COCO test set, Figure 2 reviews the spine network of the two-stage object detection method, as well as the detection accuracy (mAP) and detection speed. "—" signifies no relevant data. Performance comparison of two-stage object detection algorithms as shown in Figure 2.

The two-stage object detector, as shown in Figure 2, presents profound pillar networks such as ResNet (Allen-Zhu and Li, 2019) and ResNeXt (Hitawala, 2018), and the detection precision can reach 83.6%, but the expansion of the algorithm model causes an increase in the amount of calculation, and the detection speed is only 11% frame/s, which cannot meet the real-time requirements. Table 2 outlines the benefits, drawbacks, and contexts in which certain object detection techniques can be used.

It can be realized from Table 2, that the two-stage object detection algorithm has been making up for the faults of the preceding algorithm, but the problems such as large model scale and slow detection speed have not been solved. In this regard, some researchers put forward the idea of transforming Object detection into regression problems, simplifying the algorithm model, and improving the detection accuracy while improving the detection speed.

3.2 Single-stage object detection algorithm

The single-stage object detection technique, also known as the object detection algorithm based on regression analysis, is

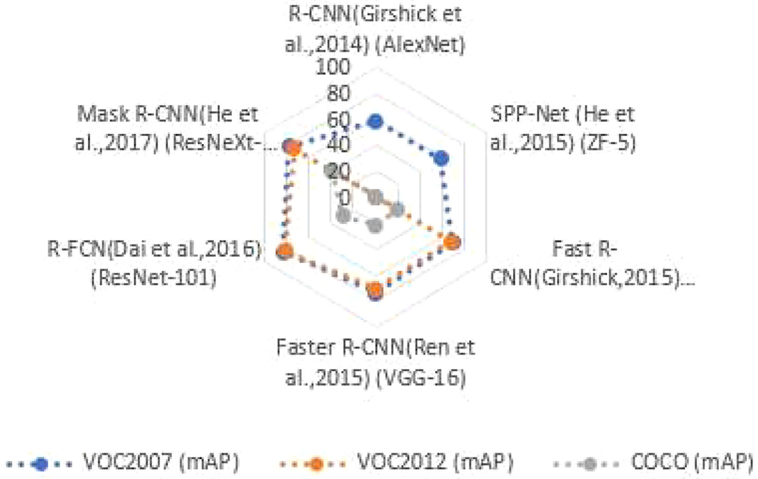


FIGURE 2
Performance comparison of two-stage object detection algorithms.

TABLE 2 Advantages, disadvantages, and applicable scenarios of two-stage Object detection algorithms.

Model	Advantage	Disadvantage	Applicable	References of Applications in Agriculture, Multimedia and Remote Sensing
OverFeat	Feature extraction using CNN	Using a sliding window, the time and space overhead is large	Object Detection	(Diwan et al., 2022; Li K. et al., 2020)
R-CNN	Combining CNN with the candidate box method	Feature extraction is complex, time-consuming, fixed image input size	Object Detection	(Yan et al., 2019; Jiao et al., 2020)
SPP-Net	Perform convolution operation on the entire image to realize multi-scale convolution calculation	High space cost	Object Detection	(Karim et al., 2020; Kumar and Kumar, 2022)
Fast R-CNN	Extract features with ROI Pooling layer, saving time and feature loading space	The selection of candidate regions is computationally complex	Object Detection	(Li M. et al., 2020; Yi et al., 2021)
Faster R-CNN	Replacing region proposals with RPN to speed up training and accuracy	The model is complex and the spatial quantification is rough	Object Detection	(Cynthia et al., 2019; Zhang et al., 2022)
R-FCN	Improved positioning accuracy	The model process is multifaceted and the amount of calculation is large	Object Detection	(Gera et al., 2022; Nguyen, 2022; Cai and Zhang, 2022)
Mask R-CNN	Solve the misalignment between the feature map and the original image, combining detection and segmentation	Instance segmentation is expensive	Object detection, instance segmentation	(Jian et al., 2022; Storey et al., 2022)

based on the principle of regression analysis. The single-stage object detector, which is generally represented by the YOLO and SSD series, skips the applicant area generation stage and obtains object classification and position information directly.

3.2.1 YOLO object detection algorithm

Redmon et al. (2016) proposed the YOLO (You Only Look Once) target detector in 2016. The YOLO architecture comprises of 24 convolutional layers and 2 FC layers, with the topmost feature map predicting bounding boxes and the P-Relu activation function explicitly evaluating the likelihood of each class. The following loss function is optimized during training:

$$\begin{aligned}&\lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{obj}[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2]\\&+\lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{obj}\left[\left(\sqrt{w_i}-\sqrt{\hat{x}_i}\right)^2+\left(\sqrt{h_i}-\sqrt{\hat{h}_i}\right)^2\right]\\&+\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{obj}(C_i-\hat{C}_i)^2+\lambda_{noobj}\sum_{i=0}^{S^2}\sum_{j=0}^{B}\mathbb{1}_{ij}^{noobj}(C_i-\hat{C}_i)^2\\&+\sum_{i=0}^{S^2}\mathbb{1}_{ij}^{noobj}\sum_{c\in classes}(p_i(c)-\hat{p}_i(c))^2\end{aligned}\tag{8}$$

where, n is a certain cell of i,(x_i,y_i) and denotes the center of the box relative to the grid cell limits, (w_i,h_i) are the standardized width and height relative to the image size. The confidence scores are represented by C_i, the existence of objects is indicated by $\mathbb{1}_{i}^{obj}$, and the prediction is made by the j^{th} bounding box predictor is indicated by $\mathbb{1}_{ij}^{obj}$.

The technique eliminates the stage of generating candidate regions and combines feature extraction, regression, and classification into a single volume. The YOLO detection speed in real-time is 45 frames per second, and the average detection accuracy mAP is 63.4%. YOLO's detection effect on small-scale objects, on the other hand, is poor, and it's simple to miss detection in environments where objects overlap and occlude.

Zhou et al. (2022) proposed YOLOv5 with total of four network models: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The detection speed of YOLOv5 is very fast, and the inference time of each picture reaches 0.007 s, which is 140 frame/s. The generalization process of the YOLO series is not good in dealing with uncommon scale objects, and multiple down sampling is required to obtain standard features. Moreover, due to the influence of space limitation in bounding box prediction, the detection effect of small object detection is not good.

3.2.2 SSD object detection algorithm

Liu et al. (2016) introduced the SSD (Single Shot multi-box Detector) algorithm to balance detection accuracy and detection speed by combining the advantages of Faster RCNN and YOLO. For feature extraction, SSD uses the VGG-16 backbone network. Convolutional layers take the place of FC6 and FC7 and add four different levels. SSD also employs a target prediction method to distinguish between target types and positions based on candidate frames collected by the anchor at various scales. The following are some of the benefits of this mechanism: (1) The convolutional layer predicts the target location and category, reducing the amount of computation; (2) the object detection process has no spatial limitations, allowing it to detect clusters of small target items effectively. The running speed of SSD on Nvidia Titan X is increased to 59 frame/s, which is significantly better than YOLO; the mAP on the VOC2007 dataset reaches 79.8%, which is 3 times that of Faster R-CNN.

3.2.3 RetinaNet algorithm

Lin et al. (2017b) borrowed the ideas of Faster R-CNN and multi-scale Object detection Erhan et al. (2014) to design and train a RetinaNet Object detector. The chief idea of this module is to explain the previous detection model by reshaping the Focal Loss Function. The problem of class imbalance of positive and negative samples in training samples during training. The ResNet backbone network and two task-specific FCN subnetworks make up the RetinaNet network, which is a single network. Convolutional features are computed over the entire image by the backbone network. On the output of the backbone network, the regression subnetworks conduct image classification tasks. Convolutional bounding box regression is handled by the network.

In one-stage detectors, the class imbalance of foreground and background is the main reason for the convergence of network training. During the training phase, Focal Loss avoids many simple negative examples and focuses on hard training samples. By training unbalanced positive and negative instances, the speed of single-stage detectors is inherited. The experimental results show that on the MS COCO test set, the AP of RetinaNet using the ResNet-101-FPN backbone network is increased by 6% compared with the DSSD513; using the ResNeXt-101-FPN, the AP of RetinaNet is increased by 9%.

3.2.4 Tiny RetinaNet algorithm

Cheng M. et al. (2020) planned Tiny RetinaNet, which customs MobileNetV2-FPN as the backbone network for feature extraction, primarily composed of Stem block backbone network and SEnet, as well as two task-specific subnets, to improve accuracy and reduce information. The mAPs for the PASCAL VOC2007 and PASCAL VOC2012 datasets are respectively 71.4% and 73.8%.

3.2.5 M2Det algorithm

Zhao et al. (2019) proposed M2Det based on Multi-Level Feature Pyramid Network (ML-FPN), which solved the problem of scale variation between target instances. The model achieves the final incremental feature pyramid through three steps: (1) extract multi-layer features from a huge number of layers in the backbone network and fuse them into basic features; (2) send the base layer features into TUM (Thinned U-shape Modules) In a block formed by connecting the module and the FFM (Feature Fusion Modules) module, the TUM decoding layer is obtained as the input of the next step; (3) The decoding layer of equivalent scale is integrated to construct a feature pyramid of multi-layer features. M2Det adopts the VGG backbone network and obtains 41.0% AP at a speed of 1.8 frame/s using the single-scale inference strategy on the MS COCO test dataset, and 44.2% AP using the multi-scale inference strategy.

3.2.6 Comparison of single-stage object detection algorithms

The single-stage object detection algorithm was developed later than the two-stage object detection algorithm, but it has piqued the interest of many academics due to its simplified structure and efficient calculation, as well as its rapid development. Single-stage object detection algorithms are frequently rapid, but their detection precision is much substandard to that of two-stage detection methods. With the rapid advancement of computer vision, the present single-stage object detection framework's speed and accuracy have substantially increased. Figure 3, reviews the backbone network of the single-stage detection algorithm and the detection accuracy (mAP) and detection speed on the PASCAL VOC2007 test set, PASCAL VOC2012 test set and COCO test set, as well as Table 3 recaps the advantages, disadvantages and applicable situations of the single-stage object detection algorithm. The Performance assessment of single-stage Object detection algorithms as shown in Figure 3.

Table 3 shows how the single-stage object detection algorithm improves object detection performance by employing pyramids to pact with pose changes and small object detection problems, novel training tactics, data augmentation, a mixture of changed backbone networks, multiple detection frameworks, and other techniques. The YOLO series is not practical for small-scale and dense object detection, and the SSD series has improved this to achieve high-precision, multi-scale detection.

3.3 Object detection algorithm based on Generative Adversarial Networks

Goodfellow et al. (2014) proposed Generative Adversarial Networks (GANs), which are unsupervised generative models that work based on the maximum likelihood principle and use adversarial training. The objective behind adversarial learning is to train the detection network by using an adversarial network to generate occlusion and deformed image samples, and it is one of the most used generative model methods for generating data distribution. GAN is more than just an image generator; it also uses training data to perform object detection, segmentation, and classification tasks across various domains.

3.3.1 A-Fast-RCNN algorithm

Wang et al. (2017) introduced the idea of adversarial networks and proposed the A-Fast-RCNN algorithm that uses adversarial networks to generate complex positive samples. Different from the traditional method of directly generating sample images, this method adopts some transformations on the feature map: (1) In the Adversarial Spatial Dropout Network (ASDN) dealing with occlusion, a

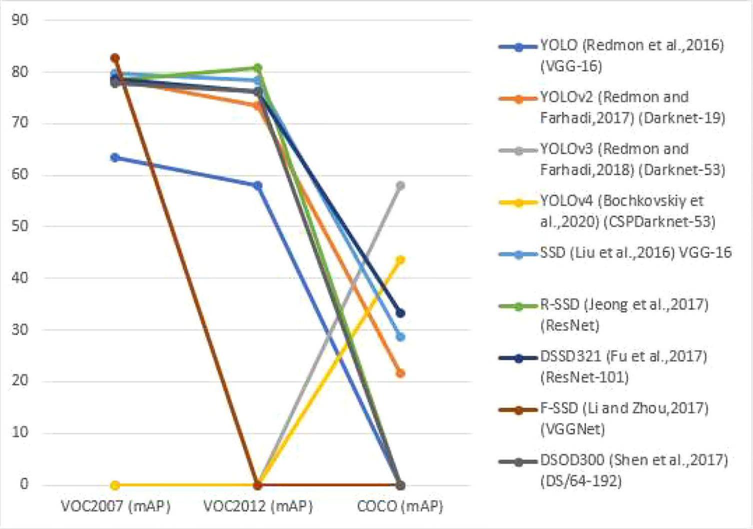


FIGURE 3
Performance assessment of single-stage Object detection algorithms in different datasets.

Mask layer is added to realize the part of the feature Occlusion, select Mask according to loss; (2) In the Adversarial Spatial Transformer Network (ASTN) that deals with deformation, partial deformation of features is achieved by manipulating the corresponding features. ASDN and ASTN provide two different variants, and by combining these two variants (ASDN output as ASTN input), the detector can be trained more robustly. In comparison with the OHEM (Online Hard Example Mining) method, on the VOC 2007 dataset, the method is slightly better (71.4% vs. 69.9%), while on the VOC 2012 dataset, OHEM is better (69.0% vs. 69.8%). The introduction of adversarial network into object detection is indeed a precedent. In terms of improvement effect, it is not as good as OHEM, and some occlusion samples may lead to misclassification. Table 4 shown the data Augmentation-

TABLE 3 Advantages, disadvantages, and applicable situations of single-stage Object detection algorithms.

Model	Advantage	Disadvantage	Applicable
YOLO	Divide the image into grid cells for fast detection	Not good for dense and small object detection	Object Detection
YOLOv2	Use clustering to make anchor boxes to improve classification precision	Using pre-training, difficult to transfer	Object Detection
YOLOv3	Using the residual learning idea to realize multi-scale detection	The model is complex, and the detection effect of medium and large-scale objects is poor	Multi-scale object detection
YOLOv4	Excellent trade-off of detection accuracy and detection speed	Detection precision needs to be better	High-precision real-time object detection
YOLOv5	Small model size, lower deployment costs, high flexibility, and high detection speed	Performance needs to be improved	Object Detection
SSD	Multi-scale anchor box discretization of boundary space	The accuracy rate is low, the model is difficult to converge, and the detection effect of small targets is not improved.	Multi-scale object detection
DSSD	Use ResNet-101 as the backbone network to improve the detection consequence of small objects	Slow detection speed compared to SSD	Object Detection
R-SSD	Improved feature fusion method to improve detection accuracy	The model calculation is complex, and the detection speed is average	Object Detection
F-SSD	Reconstruct the pyramid feature map to fuse features of different scales, which is beneficial to small object detection	Slow detection speed compared to SSD	Multi-scale object detection
DSOD	No pretraining required	Normal detection speed	Object Detection
RetinaNet	Optimize the ratio of positive and negative samples through Focal Loss	When training with dense samples, it will cause sample imbalance	Lightweight, multi-scale object detection

based object detection in Multimedia, Agriculture and Remote sensing.

3.3.2 SOD-MTGAN algorithm

Bai et al. (2018) developed an end-to-end multi-task generative adversarial network (Small Item Detection *via* Multi-Task Generative Adversarial Network, SOD-MTGAN) technique in 2018 to increase small object detection accuracy. It uses a super-resolution network to up trial small muddled photos to fine images and recover comprehensive information for more accurate detection. Furthermore, during the training phase, the discriminator's classification and regression losses are back-propagated into the generator to provide more specific information for detection. Extensive trials on the COCO dataset demonstration that the method is operative in recovering clear super-resolved images from blurred small images, and that it outperforms the state-of-the-art in terms of detecting performance (particularly for small items).

3.3.3 SAGAN algorithm

Traditional Convolutional Generative Adversarial Networks (CGANs) only generate functions of spatially local points on low-resolution feature maps, thereby generating high-resolution details. The Self-Attention Generative Adversarial Network (SA-GAN) proposed by Zhang et al. (2019) allows attention-driven and long-term dependency modeling for image generation tasks. It can generate details from cues at all feature locations, and also applies spectral normalization to improve the dynamics of training with remarkable results.

3.3.4 Your local GAN algorithm

Daras et al. (2020) proposed a two-dimensional local attention mechanism for generative models (2DLAMGM), and introduced a new local sparse attention layer that preserves 2D geometry and locality. It replaces the dense attention layer of SAGAN (Self-Attention Generative Adversarial Networks), and on ImageNet, the FID score is optimized from 18.65 to 15.94. The sparse attention pattern of the new layers proposed in this method is designed using the new information-theoretic criterion of the information flow graph, and a new method for reversing the attention of adversarial generative networks is also proposed.

3.3.5 MSG-GAN stabilized image synthesis algorithm

GANs although partially successful in image synthesis tasks, were unable to adapt to different datasets, in part due to unpredictability during training and sensitivity to hyperparameters. One cause for this instability is that when the supports of the real and virtual distributions do not overlap enough, the gradients passed from the discriminator to the generator will become underinformed. In response to the above problems, Karnewar and Wang (2019) planned a Multi-Scale Gradient Generative Adversarial Network (MSG-GAN), which consents gradients to flow from the discriminator to the generator at multiple scales for high resolution Rate image synthesis provides a stable method. MSG-GAN converges stably on datasets of different sizes, resolutions, and domains, as well as on different loss functions and architectures.

4 Deep learning-based object detection algorithm improvement

The rapid development of deep learning has increased the feasibility of improving various classical object detection algorithms in many ways. This section summarizes the main popular improvement methods from the aspects of data processing, model construction, prediction object and loss calculation, and discusses their characteristics, so that different algorithms can express different problems for different problems. The improved scheme corresponding to the algorithm detection process is shown in Figure 4.

TABLE 4 Data Augmentation-based object detection in Multimedia, Agriculture and Remote sensing.

Reference (Multimedia, Agriculture and Remote sensing)	Method description
(Haruna et al., 2022)	To improve the accuracy of deep learning models for identifying rice leaf disease, we built a GAN-based data augmentation pipeline with the state-of-the-art StyleGAN2-ADA and the variance of Laplace filter to generate high-quality synthetic rice leaf disease images.
(Bhakta et al., 2022)	Using state-of-the-art Generative Adversarial Network (GAN) technology, we can simulate thermal images of a rice plant with bacterial leaf blight.
(Liu W et al., 2021)	A multiscale attention module that boosts the Cycle-Consistent Adversarial Network (CycleGAN) in both spatial and channel dimensions to boost the quality of synthetic images.
(Yan et al., 2019)	The dataset trained a faster region-based convolutional neural network (Faster R-CNN) built on Res101netwok, which was then used to classify both synthetic and real images.
(Bosquet et al., 2022)	Synthetic data of superior quality achieved by combining a GAN with image inpainting and mixing. DS-GAN can create believable miniature things.

4.1 Data processing

4.1.1 Data augmentation

In the object detection algorithm based on deep learning, data augmentation techniques are divided into two types: supervised and unsupervised. Supervised data augmentation methods can be separated into three classes: geometric changes, color transformations, and hybrid transformations; unsupervised data augmentation methods can be divided into two sorts: generating new data and learning new augmentation strategies.

Currently, the research on supervised data augmentation strategies has tended to be perfect, and it has become the main requirement to combine multiple data augmentation techniques to improve model performance. The main reasons are as follows:

1. The widespread use of supervised data enhancement methods makes unsupervised data enhancement methods less valued to a certain extent;
2. The Object detection algorithm is gradually developing towards an end-to-end network, integrating data enhancement methods. It has become a requirement in the algorithm, but the unsupervised data enhancement method has certain difficulties in integration due to its complexity and large amount of calculation, and its application scope is limited;
3. The generative adversarial network or reinforcement learning-related technologies required for unsupervised data augmentation methods are complex and diverse, which hinders researchers' exploration.

4.2 Model construction

4.2.1 Improve the network structure

In 2015, the ResNet network first proposed the residual block (Residual block), which made the convolutional network deeper and less prone to degradation. As an improvement of the ResNet network, the DenseNet network Huang G. et al. (2017) achieves feature reuse by establishing dense connections among all former layers and the current layer, which can achieve well performance than the ResNet network with fewer parameters and less computational cost. The core part of the GoogLeNet network is the Inception module, which extracts the feature information of the image through different convolution kernels, and uses a 1×1 convolution kernel for dimensionality reduction, which significantly reduces the amount of computation. Feature Pyramid Networks Lin et al. (2017) (Feature Pyramid Networks, FPN) have made outstanding contributions to identifying small objects. As an improvement of the FPN network, the PANet network Liu et al. (2018) adds a bottom-up information transfer path based on the FPN to make up for the insufficient utilization of the underlying features. The structure is shown in Figure 5.

The existence of the fully connected layer leads to the fact that the size of the input image must be uniform, and the proposal of SPP-Net He et al. (2015) solves this problem, so that the size of the input image is not limited. Efficient-Net Tan and Le (2019) does not pursue an increase in one dimension (depth, width, image resolution) to improve the overall precision of the model but instead explores the best combination of these three dimensions. Based on EfficientNet, Tan et al. (2020) suggested a set of Object detection frameworks, EfficientDet, which can achieve good performance for different levels of resource constraints. The comparison of the above networks is shown in Table 5.

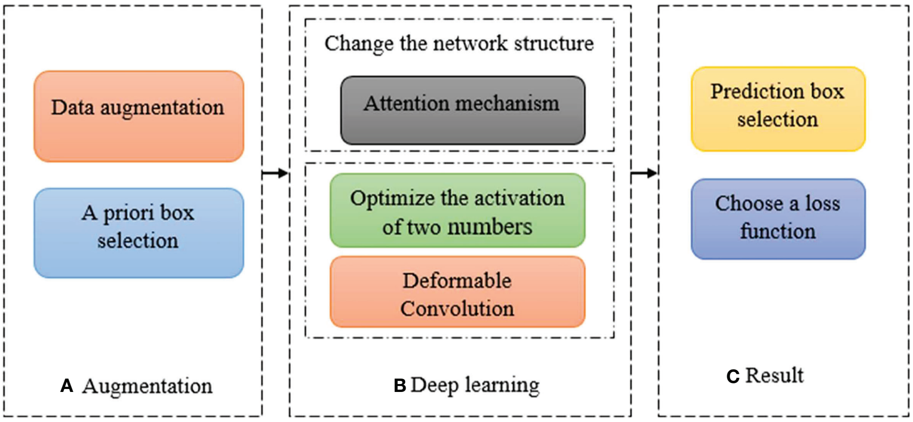


FIGURE 4
The corresponding improvement scheme of algorithm detection flow **(A)** Augmentation **(B)** Deep Learning **(C)** Results.

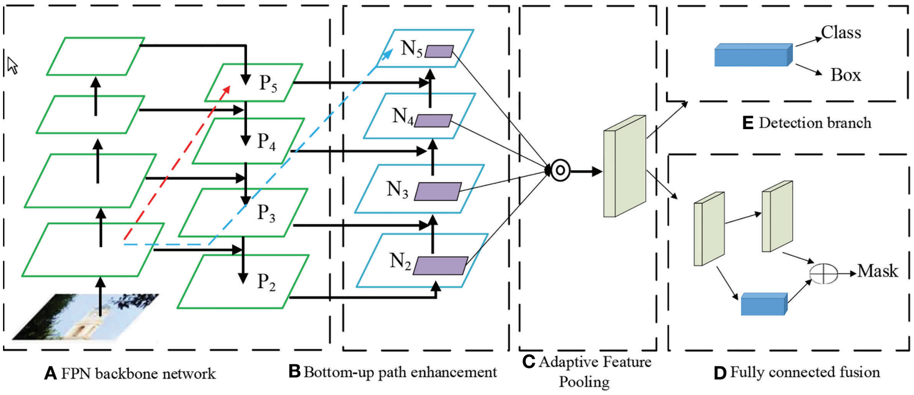


FIGURE 5
PANet model steps **(A)** FPN Backbone Network **(B)** Bottom Up Path Enhancement **(C)** Adaptive feature pooling **(D)** Fully Connected fusion.

Some scholars have introduced the above optimization scheme in the improvement of the network structure of related models to make the detection results more ideal. The related literature of the GoogLeNet network is a typical optimization method of the Inception module (Shi et al., 2017) and the optimization process is shown in Figure 6.

In order to better improve the model detection accuracy, today's network structure is gradually increasing the depth (residual module), width (Inception module) and context feature extraction capabilities of the network model (Li et al., 2016; Ghiasi et al., 2019; Cao et al., 2020b), etc. However, the resulting model is complicated and redundant, making the improved algorithm more difficult to apply in real life scenarios.

4.3 Other improved algorithms

At present, researchers have done a lot of study on the two-stage object detection algorithm and the single-stage object detection algorithm, so that they have a certain theoretical basis. The two-stage object detection algorithm has an advantage in detection accuracy, and needs to be continuously improved to enhance the detection speed; the single-stage object detection algorithm has an advantage in detection speed, and the model needs to be continuously improved to increase the detection accuracy, so some researchers put the two types of algorithm models such as detection accuracy and detection speed, as shown in Figure 7.

In 2017, the RON (Reverse connection with Objectness prior Networks) Kong et al. (2017) algorithm is an efficient and efficient algorithm based on the two-stage detection framework represented by Faster R-CNN and the single-stage detection framework signified by YOLO and SSD. Under the fully convolutional network, similar to SSD, RON uses VGG-16 as the backbone network, the difference is that RON changes the 14th and 15th fully connected layers of the VGG-16 network into a kernel size of 2 × 2. In tests, RON achieves state-of-the-art object detection performance, with input 384×384 size images, the mAP reaches 81.3% on the PASCAL VOC2007 dataset, and the mAP improves to 80.7% on the PASCAL VOC 2012 dataset. Zhang et al. (2018) designed the RefineDet algorithm, which inherited the advantages of single-stage detectors and two-stage detectors. RefineDet uses VGG-16 or ResNet-101 as the backbone network for feature extraction, and integrates the neck structure (feature pyramid and feature fusion) into the head structure.

5 Object detection and recognition applications in agriculture using AI

The use of computer vision technology to inspect agricultural products has the advantages of real-time, objective, and no damage, so it is favored by people. Saldaña et al. (2013) discussed the method of applying computer vision technology to detect mango weight and fruit surface damage, analyzed the algorithm to determine the required image area, and established the correlation between mango weight and its projected image. Experiments show that the accuracy rate of fruit surface damage classification is 76% and 80%, respectively. Slaughter and Harrell (1989) and others first studied using the chromaticity and brightness information of images taken under natural light conditions to guide the citrus harvesting manipulator, and established a classification model for identifying citrus from trees using color information in color images. The classifier was 75 percent accurate in identifying oranges from the orchard's natural environment.

Huang X. et al. (2017) realized the detection and localization of apples through pattern recognition, mainly using an algorithm to realize the identification of apples,

TABLE 5 Comparison of advantages and disadvantages of related networks.

Network name	Advantage	Disadvantage	References of applications in Multimedia, Agriculture and Remote Sensing
SPP-Net	Facilitate multi-scale training	Requires huge storage space for feature extraction and SVM classification tasks	(Ding et al., 2018; Gao et al., 2019; Hespeler et al., 2021)
GoogLeNet	Use a 1×1 convolution kernel to reduce the amount of computation; increase the width of the single-layer convolution to improve the network's ability to extract features	There is still 5×5 convolution kernels to increase the network operation; including more complex hyperparameters, each transformation needs to specify the size and number of convolution kernels	(Ding et al., 2019; Eser, 2021; Diwan et al., 2022)
ResNet	The residual module adopts skip connection, which alleviates the problem of gradient disappearance and degradation caused by the network being too deep.	The number of limits is large, and the hardware requirements are slightly higher; when the number of network layers is too deep, the mitigation effect of problems such as gradient disappearance will be greatly reduced	(Zhong et al., 2018; Pan et al., 2021; Storey et al., 2022)
DenseNet	Compared with ResNet, the amount of parameters and computation is greatly reduced, and the accuracy is improved; it effectively solves the problem of overfitting caused by too few data sets; dense connections are used to strengthen feature propagation	During training, since the splicing operation will re-open a new memory storage space to save the spliced feature information, it consumes a lot of memory.	(Zhu et al., 2019; Dubey et al., 2023; Huang X. et al., 2017)
FPN	Multi-scale feature fusion to improve the accuracy of small Object detection	Top-down structure, the underlying features are not fully utilized	(Hu et al., 2022; Gunturu et al., 2022;Liu N. et al., 2021)
PANet	Make full use of high-level semantic information and low-level location information	In addition to the top-down structure, a bottom-up structure is also constructed, which requires a lot of additional computational overhead	(Cheng G. et al., 2020;Chen et al., 2021; Piao et al., 2021)
ResNeXt	The multi-branch network structure is simplified by grouping convolution; the overall performance is better than ResNet when the parameter quantity remains basically unchanged; the modular structure is easy to transplant;	Compared with the overall operation, grouped convolution is less efficient in hardware execution.	(Lin et al., 2020; Savarimuthu, 2021; Shi et al., 2021)
EfficientNet	The three dimensions of network depth, width and image resolution are well balanced; in the case of reducing the amount of parameters, the detection accuracy has been qualitatively improved	There are too many network layers, and the intermediate results of all layers need to be saved during gradient calculation, which requires high hardware and occupies a large amount of video memory; when the image size is too large, the training speed will be slowed down	(Alhichri et al., 2021; Nguyen et al., 2021; Chatterjee et al., 2022)
EfficientDet	The Bidirectional Feature Pyramid Network (BiFPN) proposed on the basis of PANet has the characteristics of cross-scale connection and weighted feature fusion, which is more efficient for feature detection; compound scaling is performed on multiple aspects at the same time to find the depth, width, and resolution. The best combination results in more accurate and objective results; it is ahead of common target detection models in terms of accuracy and computational complexity, such as: Yolo v3, Mask-RCNN, etc.	In view of its characteristics of using neural network to search for the optimal architecture, the time and hardware cost required for training the model will be extremely high; the target detection framework has poor modular structure, which is not conducive to integration	(Wei et al., 2021; Chatterjee et al., 2022; Basavegowda et al., 2022)

filtering and boundary extraction of the original image of the apple tree, and calculating Determines the outline of the apple relative to the shape of the image. Wang and Cheng (2004) studied the identification method of apple fruit stem and fruit body and the search method of fruit surface defect. According to the characteristics of apple fruit stalk, it is proposed to use block scanning to judge whether the fruit stalk exists; the different reflection characteristics of the damaged surface and the non-damaged surface of the apple, as well as the statistical characteristics of the pixel points of different gray values, are analyzed to find out the damaged surface. The damaged area was separated from the fruit pedicel and the fruit calyx. The judging accuracy rate of 15 images without fruit stems was 100%, and the accuracy rate of 90 pictures with intact fruit

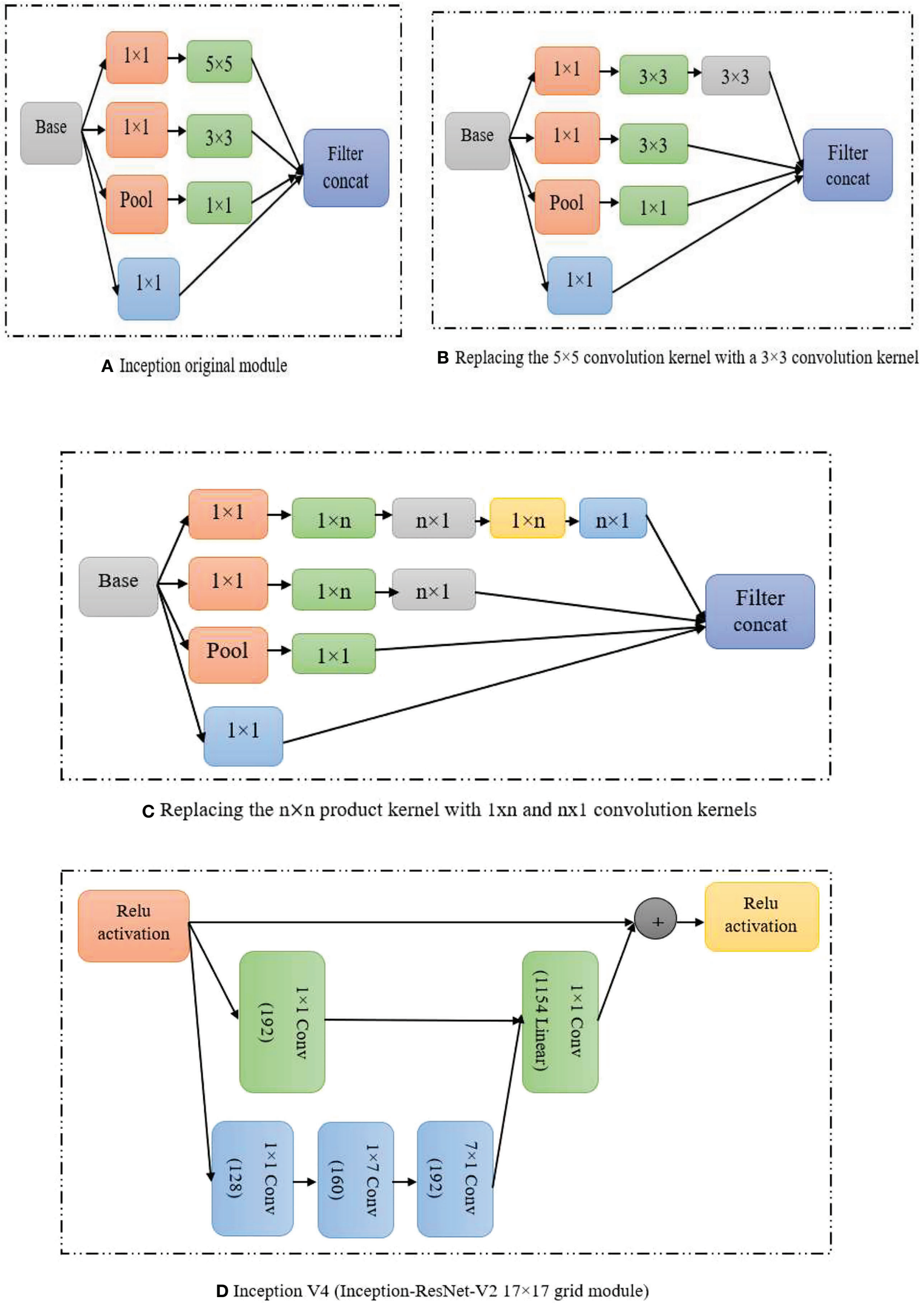


FIGURE 6
Inception modules **(A)** Inception original module **(B)** Replacing the 5*5 convolution kernel with a 3*3 convolutional kernal **(C)** Single * n kernel **(D)** Inception V4.

stems was 88%. Mahanti et al. (2021) used line scanning and analog cameras to detect apple damage, respectively, and showed that using digital image processing technology to detect apple damage can at least reach the accuracy of manual classification.

Ying et al. (2000) used computer vision for a new method of huanghua pear fruit stalk recognition. The computer vision system was used to capture images of huanghua pear, and image processing technology was used to complete the segmentation of the image and the background. The stem

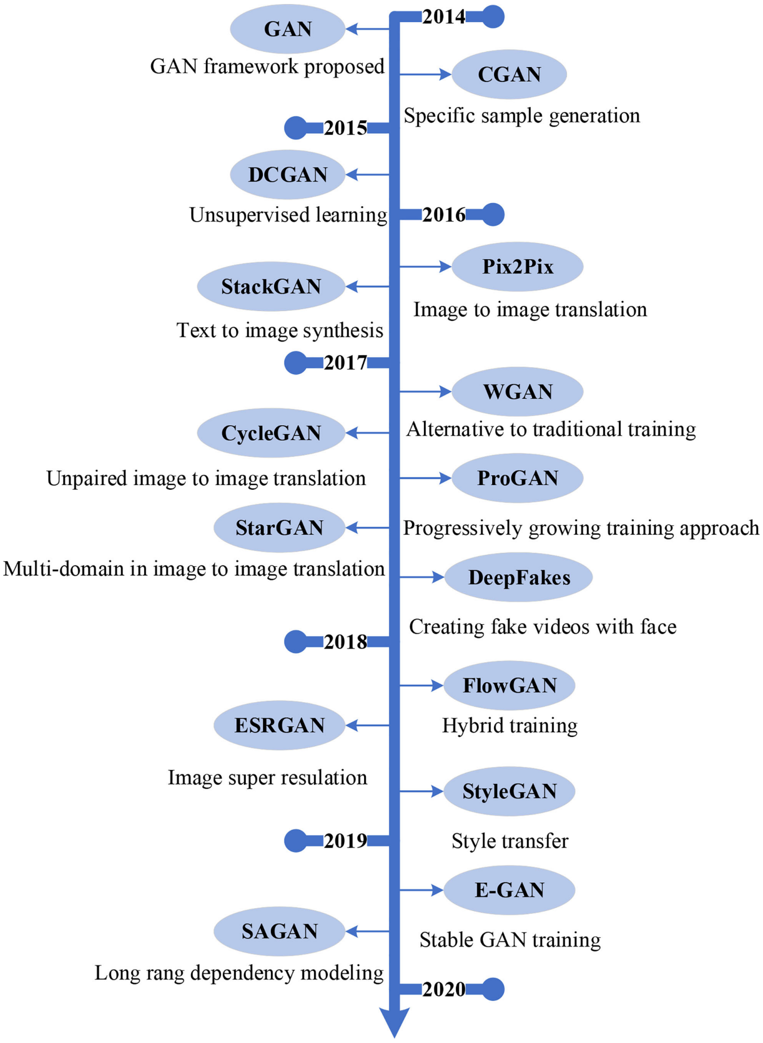

2014
GAN
GAN framework proposed
CGAN
Specific sample generation
2015
DCGAN
Unsupervised learning
2016
Pix2Pix
StackGAN
Image to image translation
Text to image synthesis
2017
WGAN
CycleGAN
Alternative to traditional training
Unpaired image to image translation
ProGAN
StarGAN
Progressively growing training approach
Multi-domain in image to image translation
DeepFakes
Creating fake videos with face
2018
FlowGAN
ESRGAN
Hybrid training
Image super resulation
StyleGAN
2019
Style transfer
E-GAN
SAGAN
Stable GAN training
Long rang dependency modeling
2020

FIGURE 7
The Evolution of mainstream GAN.

speed is slow, so a fast algorithm is proposed. This method uses the small diameter of the stem of the pear, selects templates of different sizes, determines whether there is a stem in the image, and obtains the coordinates of the intersection of the head of the stem and the bottom of the pear. The tangent slope information is used to judge the integrity of the fruit stalk. The test results show that the algorithm can 100% judge whether the fruit stalk exists, and the correct rate of judging whether the fruit stalk is intact is more than 90%. Li et al. (2018) applied computer vision technology to detect the bruising injury of pears, and proposed to distinguish multiple bruising injuries by regional marking technology. In order to improve the measurement accuracy of the bruising area, a mathematical model for measuring the bruising area was established according to the shape of the pear and the characteristics of the bruising. This method can accurately detect multiple crush injuries of pears, and the relative error of most measurements can be controlled within 10%. Patel et al. (2012) conducted an experimental study on Huanghua pear's machine vision technology to detect the external dimension and performance status. By determining the image processing window, using the Sobel operator and Hilditch to refine the edge, and determining the centroid point to find the representative fruit diameter, the test results show that the correlation coefficient between the predicted fruit diameter and the actual size can reach 0.96. For the detection of fruit surface defects, it is proposed to use the mutation of red (R) and green (G) color components at the junction of damaged and non-damaged to obtain suspicious points, and then to obtain the entire damaged surface through regional growth. Chang (2022) developed a machine vision system for the quality inspection of

Huanghuali, taking Huanghuali as the research object, and compared the influence of different intensity light sources and different backgrounds on the collected images, and developed a system suitable for Huanghuali and different backgrounds. Machine vision systems for other fruit quality inspections. Cubero et al. (2011) developed a machine vision system suitable for the quality inspection of Huanghuali by studying the spectroscopic reflection characteristics of Huanghuali. In order to adapt to the randomness of fruit orientation and the irregularity of fruit shape in actual production According to the requirements of the fruit size detection method, the method of fruit size detection has better adaptability. A method of using the minimum circumscribed rectangle (MER) method of fruit to find the maximum transverse diameter is designed, and the experimental verification is carried out, and the actual maximum transverse diameter is obtained. The regression equation of the relationship between the diameter and the predicted transverse diameter, the relationship between the two The coefficient is 0.996 2. The variation characteristics of the gray levels of R, G, and B components in the defect area of Huanghuali were analyzed, and finally the maximum combined set of defect pixels and all defect areas were found.

Li et al. (2022) put forward a method for identifying germ and endosperm with saturation S as a characteristic parameter by analyzing the color characteristics of germ rice and color images, in order to realize the automatic computer vision of rice germ retention rate detection. Experiments are carried out with the established identification indicators and methods, and the results show that the coincidence rate between the identification results of the computer vision system and the manual detection is over 88%.

6 Object detection and recognition applications in agriculture using AI

The detection and recognition of objects based on remote sensing images is a current research focus in the field of target detection. AI brings much improvement in different applications of computer vision and a lot of latest progress in all applications improve it methods (Nawaz et al., 2020; Nawaz et al., 2021). The detection and recognition methods used can be divided into two types: target detection algorithms based on traditional methods and target detection algorithms based on deep learning. Commonly used target detection algorithms based on traditional methods include HOG feature algorithm combined with SVM algorithm, Deformable Parts Model (DPM), etc.; target detection and recognition algorithms based on deep learning can be roughly summarized into two categories, namely R-CNN series algorithm based on two stage method and YOLO series algorithm based on one stage method (Han et al., 2022), SSD (Single Shot Multibox Detector) series algorithm (Arora et al., 2019).

Initially, the detection of remote sensing images to obtain information is mainly through manual visual analysis, and the amount of information obtained in this way completely depends on the professional ability of technicians. After more than ten years of development, a new technology has appeared to be applied to the reading of remote sensing image information. This new method detects and recognizes targets through statistical models. For example, Peng et al. (2018) is in order to achieve higher classification accuracy using the maximum likelihood method for remote sensing image classification, etc. Kassim et al. (2021) proposed a multi-degree learning method, which first combined feature extraction with active learning methods, and then added a K-means classification algorithm to improve the performance of the algorithm. Du et al. (2012) proposed the adaptive binary tree SVM classifier, which has further improved the classification accuracy of hyperspectral images. Luo et al. (2016) studied an algorithm called small random forest, the purpose is to solve the problem of low accuracy and overfitting of decision trees. In addition, due to the problems of low detection accuracy and long time consumption, the traditional target detection method cannot meet the real-time requirements of the algorithm in practical applications.

In 2006, Geoffrey Hinton and his students published a paper related to deep learning (Hinton and Salakhutdinov, 2006), which opened the door to object detection and recognition using deep learning. In recent years, with the breakthrough of deep learning theory, the detection accuracy and detection speed of target detection algorithms have been effectively improved, so that the feature information in images can be extracted by deep learning, which gradually replaces the information based on manual methods and traditional methods. Extraction has become the main direction of object detection research.

In the 2017 ImageNet competition, trained and learned a million image datasets through the design of a multi-layer convolutional neural network structure. The classification error rate obtained in the final experiment was only 15%, and the second place in the competition. That's nearly 11% higher. In addition, many researchers have used deep learning to detect and recognize remote sensing image targets, and have achieved good results and achieved many breakthroughs (Krizhevsky et al., 2017). Mnih and Hinton (2010) used two datasets of remote sensing images to conduct research on deep learning technology. They extracted road features from images for training and achieved good experimental results. This is the first time that deep learning is used. applied to remote sensing technology. Zou et al. (2015) developed a new algorithm for extracting features in images. The algorithm designed a deep belief network structure and conducted experiments on feature extraction, and finally achieved an accuracy of 77%. Ienco et al.

(2019) used a combination of deep learning and a patch classification system to detect ground cover, and achieved good detection results. Wei et al. (2017) developed a more accurate convolutional neural network for road structure feature extraction, and this algorithm has a remarkable effect on road extraction from aerial images. Cheng et al. (2018) proposed a rotation-invariant CNN (RICNN) model, which effectively addresses the technical difficulties of object detection in high-resolution remote sensing images. From the object detection experiment of remote sensing images using deep learning, it can be concluded that the extraction of target features by constructing a deep model structure can effectively improve the detection effect. (Bhatti et al., 2021) used edge detection for identification of objects in remote sensing images by using geometric algebra methods.

7 Challenges for object detection in agriculture

7.1 Insufficient individual feature layers

Deep CNN plannings generate hierarchy feature maps due to pooling and subsampling operations, resulting in changed layers of feature maps with differing 3D resolutions. As is generally known, the feature maps of the early-layer feature maps have a higher resolution and signify smaller response fields. They also lack high-level semantic information, which is necessary for object detection. The latter-layer feature maps, on the other hand, contain additional semantic information that is required for detecting and classifying things like distinct object placements and illuminations. Higher-level feature maps are valuable for classifying large objects, but they may not be enough to recognize small ones.

7.2 Limited context information

Small items usually have low resolutions, which makes it difficult to distinguish them. Contextual information is crucial in small item detection because small objects themselves carry limited information. From a "global" picture level to a "local" image level, contextual information has been utilized in object recognition. A global image level takes into account image statistics from the entire image, whereas a local image level takes into account contextual information from the objects' surrounding areas. Contextual characteristics can be divided into three categories such as local pixel context, semantic context, and spatial context.

7.3 Class imbalance

The term "class imbalance" refers to the unequal distribution of data between classes. There are two different sorts of class disparities. One issue is a disparity between foreground and background instances. By densely scanning the entire image, region proposal networks are utilized in object detection to create possible regions containing objects. The anchors are rectangular boxes that have been extensively tiled throughout the full input image. Anchor scales and ratios are pre-determined based on the sizes of target items in the training dataset. When detecting little items, the number of anchors generated per image is higher than when recognizing large things. Positive instances are only those anchors that have a high IoU with the ground truth bounding boxes. Anchors are considered bad examples since they have little or no overlap with the ground truth bounding boxes. The sparseness of ground-truth bounding boxes and IoU matching procedures between ground-truth and anchors are both drawbacks of the anchor-based object identification methodology, and the dense sliding window strategy has a high temporal complexity, making training time consuming.

7.4 Insufficient positive examples

Most object detection deep neural network models were proficient with objects of varying sizes. They usually work well with huge objects but not so well with small ones. A lack of small-scale anchor boxes produced to match the small objects, as well as an inadequate number of examples to be properly matched to the ground truth, could be the cause. The anchors are feature mappings from certain intermediate layers in a deep neural network that are projected back to the original image. Anchors for little objects are difficult to come by. In addition, the anchors must match the ground truth bounding boxes. The following is an example of a widely used matching method. A positive example is one that has a high IoU score in relation to a ground truth bounding box, such as more than 0.9. Furthermore, the anchor with the highest IoU score for each ground truth box is designated as a positive example. As a result, small objects usually have a limited number of anchors that match the ground truth bonding boxes.

8 Conclusion

Deep learning-based object detection techniques have become a trendy research area due to their powerful learning capabilities and superiority in handling occlusion, scale variation, and background exchange. In this paper, we introduce the development of object detection algorithms based on deep learning and summarize two types of object detectors such as single and two-stage. In-depth analysis of the network structure, advantages, disadvantages, and applicable scenarios of various algorithms, we compare the analysis of standard data sets and experimental results of different related algorithms on mainstream data sets. Finally, this study summarizes some application areas of object detection to comprehensively understand and analyze its future development trend.

Future work

Based on the analysis and summary of the above knowledge, we propose the following directions for future research.

- Video object detection has problems such as uneven moving targets, tiny targets, truncation, and occlusion, and it isn't easy to achieve high precision and high efficiency. Therefore, studying multi-faceted data sources such as motion-based objects and video sequences will be one of the most promising future research areas.
- Weakly supervised object detection models aim to detect many non-annotated corresponding objects using a small set of fully annotated images. Therefore, using many annotated and labeled pictures with target objects and bounding boxes to train the network to achieve high effectiveness efficiently is an essential issue for future research.
- Region-specific detectors tend to perform better, achieving higher detection accuracy on predefined datasets. Therefore, developing a general object detector that can detect multi-domain objects without prior knowledge is a fundamental research direction in the future.
- Remote sensing photos are frequently employed in military and agricultural industries and are detected in real-time. The rapid development of these fields will be aided by automatic model detection and integrated hardware components.

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

Funding acquisition: JL; Project administration: MS, SN, JL, UB, and RA; Writing – original draft: SN. All authors contributed to the article and approved the submitted version.

Conflict of interest

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

Publisher's note

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

References

Alhichri, H., Alswayed, A. S., Bazi, Y., Ammour, N., and Alajlan, N. A. (2021). Classification of remote sensing images using EfficientNet-B3 CNN model with attention. *IEEE Access* 9, 14078–14094. doi: 10.1109/ACCESS.2021.3051085

Allen-Zhu, Z., and Li, Y. (2019). What can resnet learn efficiently, going beyond kernels? *Adv. Neural Inf. Process. Syst.* 32. doi: 10.48550/arXiv.1905.10337

Arora, A., Grover, A., Chugh, R., and Reka, S. S. (2019). Real time multi object detection for blind using single shot multibox detector. *Wireless. Pers. Commun.* 107 (1), 651–661. doi: 10.1007/s11277-019-06294-1

Ashritha, P., Banusri, M., Namitha, R., and Duela, ,. J. S. (2021). "Effective fault detection approach for cloud computing," in *Journal of physics: Conference series*, vol. 1979. (Sidney, Australia: IOP Publishing), 012061.

Bai, Y., Zhang, Y., Ding, M., and Ghanem, B. (2018). "Sod-mtgan: Small object detection via multi-task generative adversarial network," in *Proceedings of the European conference on computer vision (ECCV)* (Munich, Germany: Springer), 206–221.

Basavegowda, D. H., Mosebach, P., Schleip, I., and Weltzien, C. (2022). *Indicator plant species detection in grassland using EfficientDet object detector* (Bonn, Germany: GIL-Jahrestagung, Künstliche Intelligenz in der Agrar-und Ernährungswirtschaft), 42.

Bhakta, I., Phadikar, S., and Majumder, K. (2022). "Thermal image augmentation with generative adversarial network for agricultural disease prediction," in *International conference on computational intelligence in pattern recognition* (Singapore: Springer), 345–354.

Bhatti, U. A., Huang, M., Wu, D., Zhang, Y., Mehmood, A., and Han, H. (2019). Recommendation system using feature extraction and pattern recognition in clinical care systems. *Enterprise. Inf. Syst.* 13 (3), 329–351. doi: 10.1080/17517575.2018.1557256

Bhatti, U. A., Ming-Quan, Z., Qing-Song, H., Ali, S., Hussain, A., Yuhuan, Y., et al. (2021). Advanced color edge detection using Clifford algebra in satellite images. *IEEE Photonics. J.* 13 (2), 1–20. doi: 10.1109/JPHOT.2021.3059703

Bingtao, G., Xiaorui, W., Yujiao, C., Zhaohui, L., and Jianlei, Z. (2015). A high-accuracy infrared simulation model based on establishing the linear relationship between the outputs of different infrared imaging systems. *Infrared. Phys. Technol.* 69, 155–163. doi: 10.1016/j.infrared.2015.01.010

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

Bosquet, B., Cores, D., Seidenari, L., Brea, V. M., Mucientes, M., and Del Bimbo, A. (2022). A full data augmentation pipeline for small object detection based on generative adversarial networks. *Pattern Recogn.* 133, 108998. doi: 10.1016/j.patcog.2022.108998

Cai, D., and Zhang, P. (2022). "Rotating target detection for remote sensing images based on dense attention," in *International conference on computing, control and industrial engineering* (Singapore: Springer), 50–63.

Cao, J., Chen, Q., Guo, J., and Shi, R. (2020b). Attention-guided context feature pyramid network for object detection. *arXiv. preprint. arXiv.* 2005, 11475. doi: 10.48550/arXiv.2005.11475

Cao, J., Kong, Y., Zhang, X., Li, Y., and Xie, ,. X. (2020a). "Target detection algorithm based on improved multi-scale SSD," in *Journal of physics: Conference series*, vol. 1570. (Zhangjiajie, China: IOP Publishing), 012014.

Chang, X. (2022). "Application of computer vision technology in post-harvest processing of fruits and vegetables: Starting from shape recognition algorithm," in *2022 international conference on applied artificial intelligence and computing (ICAAIC)* (Salem, India: IEEE), 934–937.

Chatterjee, R., Chatterjee, A., Islam, S. K., and Khan, M. K. (2022). *An object detection-based few-shot learning approach for multimedia quality assessment, Multimedia Systems* (Springer), 1–14.

Cheng, M., Bai, J., Li, L., Chen, Q., Zhou, X., Zhang, H., et al. (2020). "Tiny-RetinaNet: a one-stage detector for real-time object detection," in *Eleventh international conference on graphics and image processing (ICGIP 2019)*, vol. 11373. (Hangzhou, China: International Society for Optics and Photonics), 113730R.

Cheng, G., Han, J., Zhou, P., and Xu, D. (2018). Learning rotation-invariant and fisher discriminative convolutional neural networks for object detection. *IEEE Trans. Image. Process.* 28 (1), 265–278. doi: 10.1109/tip.2018.2867198

Cheng, G., Si, Y., Hong, H., Yao, X., and Guo, L. (2020). Cross-scale feature fusion for object detection in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* 18 (3), 431–435. doi: 10.1109/lgrs.2020.2975541

Chen, J. W., Lin, W. J., Cheng, H. J., Hung, C. L., Lin, C. Y., and Chen, S. P. (2021). A smartphone-based application for scale pest detection using multiple-object detection methods. *Electronics* 10 (4), 372. doi: 10.3390/electronics10040372

Chen, S., and Wang, ,. H. (2014). "SAR target recognition based on deep learning," in *2014 international conference on data science and advanced analytics (DSAA)* (Shanghai, China: IEEE), 541–547.

Cubero, S., Aleixos, N., Moltó, E., Gómez-Sanchis, J., and Blasco, J. (2011). Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. *Food Bioprocess. Technol.* 4 (4), 487–504. doi: 10.1007/s11947-010-0411-8

Cynthia, S. T., Hossain, K. M. S., Hasan, M. N., Asaduzzaman, M., and Das, ,. A. K. (2019). "Automated detection of plant diseases using image processing and faster r-CNN algorithm," in *2019 international conference on sustainable technologies for industry 4.0 (STI)* (Dhaka, Bangladesh: IEEE), 1–5.

Dai, J., Li, Y., He, K., and Sun, J. (2016). R-fcn: Object detection *via* region-based fully convolutional networks. *Adv. Neural Inf. Process. Syst.* 29.

Daras, G., Odena, A., Zhang, H., and Dimakis, A. G. (2020). "Your local GAN: Designing two dimensional local attention mechanisms for generative models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (Seattle, USA: IEEE/CVF), 14531–14539.

Degang, X., Lu, W., and Fan, L. (2021). *A review of typical target detection algorithms for deep learning [J/OL]* (Beijing, China: Computer engineering and application), 1–21.

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* (Miami, Florida: Ieee), 248–255.

Ding, P., Zhang, Y., Deng, W. J., Jia, P., and Kuijper, A. (2018). A light and faster regional convolutional neural network for object detection in optical remote sensing images. *ISPRS. J. Photogrammet. Remote Sens.* 141, 208–218. doi: 10.1016/j.isprsjprs.2018.05.005

Ding, P., Zhang, Y., Jia, P., and Chang, X. L. (2019). A comparison: different DCNN models for intelligent object detection in remote sensing images. *Neural Process. Lett.* 49 (3), 1369–1379. doi: 10.1007/s11063-018-9878-5

Diwan, T., Anirudh, G., and Tembhurne, J. V. (2022). Object detection using YOLO: challenges, architectural successors, datasets and applications. *Multimedia. Tools Appl.*, 1–33. doi: 10.1007/s11042-022-13644-y

Dollár, P., Wojek, C., Schiele, B., and Perona, ,. P. (2009). "Pedestrian detection: A benchmark," in *2009 IEEE conference on computer vision and pattern recognition* (Miami, Florida: IEEE), 304–311.

Dubey, N., Bhagat, E., Rana, S., and Pathak, K. (2023). "A novel approach to detect plant disease using DenseNet-121 neural network," in *Smart trends in computing and communications* (Singapore: Springer), 63–74.

Du, P., Tan, K., and Xing, X. (2012). A novel binary tree support vector machine for hyperspectral remote sensing image classification. *Optics. Commun.* 285 (13-14), 3054–3060. doi: 10.1016/j.optcom.2012.02.092

Erhan, D., Szegedy, C., Toshev, A., and Anguelov, D. (2014). "Scalable object detection using deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Columbus, Ohio: IEEE), 2147–2154.

Eser, S. E. R. T. (2021). A deep learning based approach for the detection of diseases in pepper and potato leaves. *Anadolu. Tarım. Bilimleri. Dergisi.* 36 (2), 167–178. doi: 10.7161/omuanajas.805152

Everingham, M., Eslami, S. M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vision* 111 (1), 98–136. doi: 10.1007/s11263-014-0733-5

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision* 88 (2), 303–338. doi: 10.1007/s11263-009-0275-4

Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., and Berg, A. C. (2017). Dssd: Deconvolutional single shot detector. *arXiv. arXiv. preprint. arXiv.*, 1701.06659. doi: 10.48550/arXiv.1701.06659

Gao, M., Du, Y., Yang, Y., and Zhang, J. (2019). Adaptive anchor box mechanism to improve the accuracy in the object detection system. *Multimedia. Tools Appl.* 78 (19), 27383–27402. doi: 10.1007/s11042-019-07858-w

Gera, U. K., Siddarth, D., and Singh, P. (2022). "Smart farming: Industry 4.0 in agriculture using artificial intelligence," in *Artificial intelligence for societal development and global well-being* (India: IGI Global), 211–221.

Ghiasi, G., Lin, T. Y., and Le, Q. V. (2019). "Nas-fpn: Learning scalable feature pyramid architecture for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Long Beach, California: IEEE), 7036–7045.

Girshick, R. (2015). "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision* (Washington, DC. United States: IEEE Computer Society), 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (Columbus, Ohio: IEEE), 580–587.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial networks. *Adv. Neural Inf. Process. Syst.* 63(11), 139–144. doi: 10.1145/3422622

Gunturu, S., Munir, A., Ullah, H., Welch, S., and Flippo, D. (2022). A spatial AI-based agricultural robotic platform for wheat detection and collision avoidance. *AI* 3 (3), 719–738. doi: 10.3390/ai3030042

Han, C., Zhao, Q., Zhang, S., Chen, Y., Zhang, Z., and Yuan, J. (2022). YOLOPv2: Better, faster, stronger for panoptic driving perception. *arXiv. preprint. arXiv.*, 2208.11434. doi: 10.48550/arXiv.2208.11434

Haruna, Y., Qin, S., and Mbyamm Kiki, M. J. (2022). *An improved approach to detection of rice leaf disease with GAN-based data augmentation pipeline*, (USA: SSRN) SSRN 4135061.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*. (Venice, Italy: IEEE), 2961–2969.

Hespeler, S. C., Nemati, H., and Dehghan-Niri, E. (2021). Non-destructive thermal imaging for object detection *via* advanced deep learning for robotic inspection and harvesting of chili peppers. *Artif. Intell. Agric.* 5, 102–117. doi: 10.1016/j.aiia.2021.05.003

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), 1904–1916. doi: 10.1109/TPAMI.2015.2389824

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, USA: IEEE), 770–778.

Hinton, G. E., and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science* 313 (5786), 504–507. doi: 10.1126/science.1127647

Hitawala, S. (2018). Evaluating resnext model architecture for image classification. *arXiv. preprint. arXiv.*, 1805.08700.

Huang, X., Bi, J., Zhang, N., Ding, X., Li, F., and Hou, F. (2017). Application of computer vision technology in agriculture. *Agric. Sci. Technol.* 18 (11), 2158–2162.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Honolulu, Hawaii: IEEE), 4700–4708.

Hu, Y., Dai, Y., and Wang, ,. Z. (2022). "Real-time detection of tiny objects based on a weighted bi-directional FPN," in *International conference on multimedia modeling* (Cham: Springer), 3–14.

Hu, Q., and Zhai, L. (2019). RGB-D image multi-target detection method based on 3D DSF r-CNN. *Int. J. Pattern Recogn. Artif. Intell.* 33 (08), 1954026. doi: 10.1142/S0218001419540260

Ienco, D., Interdonato, R., Gaetano, R., and Minh, D. H. T. (2019). Combining sentinel-1 and sentinel-2 satellite image time series for land cover mapping *via* a multi-source deep learning architecture. *ISPRS. J. Photogrammet. Remote Sens.* 158, 11–22. doi: 10.1016/j.isprsjprs.2019.09.016

Ito, S., Chen, P., Comte, P., Nazeeruddin, M. K., Liska, P., Péchy, P., et al. (2007). Fabrication of screen-printing pastes from TiO2 powders for dye-sensitised solar cells. *Prog. Photovoltaics.: Res. Appl.* 15 (7), 603–612. doi: 10.1002/pip.768

Jeong, J., Park, H., and Kwak, N. (2017). Enhancement of SSD by concatenating feature maps for object detection. *arXiv. preprint. arXiv.*, 1705.09587. doi: 10.5244/C.31.76

Jian, L., Pu, Z., Zhu, L., Yao, T., and Liang, X. (2022). SS R-CNN: Self-supervised learning improving mask r-CNN for ship detection in remote sensing images. *Remote Sens.* 14 (17), 4383. doi: 10.3390/rs14174383

Jiao, L., Dong, S., Zhang, S., Xie, C., and Wang, H. (2020). AF-RCNN: An anchor-free convolutional neural network for multi-categories agricultural pest detection. *Comput. Electron. Agric.* 174, 105522. doi: 10.1016/j.compag.2020.105522

Kang, H. J. (2019). "Real-time object detection on 640x480 image with vgg16+ ssd," in *2019 international conference on field-programmable technology (ICFPT)* (Tianjin, China: IEEE), 419–422.

Karim, S., Zhang, Y., Yin, S., Bibi, I., and Brohi, A. A. (2020). A brief review and challenges of object detection in optical remote sensing imagery. *Multiagent. Grid. Syst.* 16 (3), 227–243. doi: 10.3233/MGS-200330

Karnewar, A., and Wang, O. (2019). *MSG-GAN: multi-scale gradient GAN for stable image synthesis*. (Long Beach, California: CVF).

Kassim, T., Mohan, B. S., and Muneer, ,. K. A. (2021). "Modified ML-kNN and rank SVM for multi-label pattern classification," in *Journal of physics: Conference series*, vol. 1921. (Goa, India: IOP Publishing), 012027.

Kong, T., Sun, F., Yao, A., Liu, H., Lu, M., and Chen, Y. (2017). "Ron: Reverse connection with objectness prior networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (Honolulu, Hawaii: IEEE), 5936–5944.

Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., et al. (2017) *Openimages: A public dataset for large-scale multi-label and multi-class image classification*. Available at: https://github.com/openimages.

Krizhevsky, A., and Hinton, G. (2009). Learning multiple layers of features from tiny images. *utoronto, Dissertation*, 1–60

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 25, 1097–1105. doi: 10.1145/3065386

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Commun. ACM* 60 (6), 84–90. doi: 10.1145/3065386

Kumar, R., and Kumar, D. (2022). Comparative analysis of validating parameters in the deep learning models for remotely sensed images. *J. Discrete. Math. Sci. Cryptograp.* 25 (4), 913–920. doi: 10.1080/09720529.2022.2068602

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., et al. (2020). The open images dataset v4. *Int. J. Comput. Vision* 128 (7), 1956–1981. doi: 10.1007/s11263-020-01316-z

Li, J., Chen, L., and Huang, W. (2018). Detection of early bruises on peaches (Amygdalus persica l.) using hyperspectral imaging coupled with improved watershed segmentation algorithm. *Postharvest. Biol. Technol.* 135, 104–113. doi: 10.1016/j.postharvbio.2017.09.007

Lienhart, R., and Maydt, ,. J. (2002). "An extended set of haar-like features for rapid object detection," in *Proceedings. international conference on image processing*, vol. 1. (New York, USA: IEEE), I–I.

Li, B., Liu, B., Li, S., and Liu, H. (2022). An improved EfficientNet for rice germ integrity classification and recognition. *Agriculture* 12 (6), 863. doi: 10.3390/agriculture12060863

Lin, L., Chen, H., Zhang, H., Liang, J., Li, Y., Shan, Y., et al. (2020). "Dual semantic fusion network for video object detection," in *Proceedings of the 28th ACM international conference on multimedia*. (Seattle, WA (USA): ACM), 1855–1863.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Honolulu, Hawaii: IEEE), 2117–2125.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, (Venice, Italy: IEEE) 2980–2988.

Lin, S., Ji, R., Chen, C., Tao, D., and Luo, J. (2018). Holistic cnn compression *via* low-rank decomposition with knowledge transfer. *IEEE Trans. Pattern Anal. Mach. Intell.* 41 (12), 2889–2905. doi: doi.org/10.1109/tpami.2018.2873305

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "Microsoft Coco: Common objects in context," in *European Conference on computer vision* (Cham: Springer), 740–755.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., et al. (2016). "Ssd: Single shot multibox detector," in *European Conference on computer vision* (Cham: Springer), 21–37.

Liu, N., Celik, T., and Li, H. C. (2021). Gated ladder-shaped feature pyramid network for object detection in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/lgrs.2020.3046137

Liu, W., Luo, B., and Liu, J. (2021). Synthetic data augmentation using multiscale attention CycleGAN for aircraft detection in remote sensing images. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/lgrs.2021.3052017

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (Salt Lake City, UT, USA: IEEE) 8759–8768.

Li, K., Wan, G., Cheng, G., Meng, L., and Han, J. (2020). Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS. J. Photogrammet. Remote Sens.* 159, 296–307. doi: 10.1016/j.isprsjprs.2019.11.023

Li, J., Wei, Y., Liang, X., Dong, J., Xu, T., Feng, J., et al. (2016). Attentive contexts for object detection. *IEEE Trans. Multimedia.* 19 (5), 944–954. doi: 10.1109/tmm.2016.2642789

Li, M., Zhang, Z., Lei, L., Wang, X., and Guo, X. (2020). Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster r-CNN, YOLO v3 and SSD. *Sensors* 20 (17), 4938. doi: 10.3390/s20174938

Li, Z., and Zhou, F. (2017). FSSD: feature fusion single shot multibox detector. *arXiv. preprint. arXiv.*, 1712.00960. doi: 10.48550/arXiv.1712.00960

Luo, Y. M., Huang, D. T., Liu, P. Z., and Feng, H. M. (2016). An novel random forests and its application to the classification of mangroves remote sensing image. *Multimedia. Tools Appl.* 75 (16), 9707–9722. doi: 10.1007/s11042-015-2906-9

Mahanti, N. K., Pandiselvam, R., Kothakota, A., Ishwarya, P., Chakraborty, S. K., Kumar, M., et al. (2021). Emerging non-destructive imaging techniques for fruit damage detection: Image processing and analysis. *Trends Food Sci. Technol* 120, 418–438. doi: 10.1016/j.tifs.2021.12.021

Marris, H., Deboudt, K., Augustin, P., Flament, P., Blond, F., Fiani, E., et al. (2012). Fast changes in chemical composition and size distribution of fine particles during the near-field transport of industrial plumes. *Sci. Total. Environ.* 427, 126–138. doi: 10.1016/j.scitotenv.2012.03.068

Mnih, V., and Hinton, ,. G. E. (2010). "Learning to detect roads in high-resolution aerial images," in *European Conference on computer vision* (Berlin, Heidelberg: Springer), 210–223.

Moore, R. C., and DeNero, J. (2011). *L1 and L2 regularization for multiclass hinge loss models*.

Naqvi, S. F., Ali, S. S. A., Yahya, N., Yasin, M. A., Hafeez, Y., Subhani, A. R., et al. (2020). Real-time stress assessment using sliding window based convolutional neural network. *Sensors* 20 (16), 4400. doi: 10.3390/s20164400

Nawaz, S. A., Li, J., Bhatti, U. A., Bazai, S. U., Zafar, A., Bhatti, M. A., et al. (2021). A hybrid approach to forecast the COVID-19 epidemic trend. *PloS One* 16 (10), e0256971. doi: 10.1371/journal.pone.0256971

Nawaz, S. A., Li, J., Bhatti, U. A., Mehmood, A., Ahmed, R., and Ul Ain, Q. (2020). A novel hybrid discrete cosine transform speeded up robust feature-based secure medical image watermarking algorithm. *J. Med. Imaging Health Inf.* 10 (11), 2588–2599. doi: 10.1166/jmihi.2020.3220

Nguyen, H. (2022). An efficient license plate detection approach using lightweight deep convolutional neural networks. *Adv. Multimedia.* 2022, 1–10 doi: 10.1155/2022/8852142

Nguyen, T. T., Vien, Q. T., and Sellahewa, H. (2021). An efficient pest classification in smart agriculture using transfer learning. *EAI. Endorsed. Trans. Ind. Networks Intelligent. Syst.* 8 (26), 1–8. doi: 10.4108/eai.26-1-2021.168227

Pan, T. S., Huang, H. C., Lee, J. C., and Chen, C. H. (2021). Multi-scale ResNet for real-time underwater object detection. *Signal. Image. Video. Process.* 15 (5), 941–949. doi: 10.1007/s11760-020-01818-w

Patel, K. K., Kar, A., Jha, S. N., and Khan, M. A. (2012). Machine vision system: a tool for quality inspection of food and agricultural products. *J. Food Sci. Technol.* 49 (2), 123–141. doi: 10.1007/s13197-011-0321-4

Peng, J., Li, L., and Tang, Y. Y. (2018). Maximum likelihood estimation-based joint sparse representation for the classification of hyperspectral remote sensing images. *IEEE Trans. Neural Networks Learn. Syst.* 30 (6), 1790–1802. doi: 10.1109/tnnls.2018.2874432

Piao, Y., Jiang, Y., Zhang, M., Wang, J., and Lu, H. (2021). *PANet: Patch-aware network for light field salient object detection* (USA: IEEE Transactions on Cybernetics).

Rahman, M. A., and Wang, ,. Y. (2016). "Optimizing intersection-over-union in deep neural networks for image segmentation," in *International symposium on visual computing* (Cham: Springer), 234–244.

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*, (Las Vegas, NV, USA: IEEE) 779–788.

Redmon, J., and Farhadi, A. (2017). "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (Honolulu, Hawaii: IEEE Computer Society) 7263–7271.

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

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* 28. doi: 10.1109/tpami.2016.2577031

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision* 115 (3), 211–252. doi: 10.1007/s11263-015-0816-y

Saldaña, E., Siche, R., Luján, M., and Quevedo, R. (2013). Computer vision applied to the inspection and quality control of fruits and vegetables. *Braz. J. Food Technol.* 16, 254–272. doi: 10.1590/S1981-67232013005000031

Savarimuthu, N. (2021). "Investigation on object detection models for plant disease detection framework," in *2021 IEEE 6th international conference on computing, communication and automation (ICCCA)* (New Delhi, India: IEEE), 214–218.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv. preprint. arXiv.*, 1–16. doi: 10.48550/arXiv.1312.6229

Shen, Z., Liu, Z., Li, J., Jiang, Y. G., Chen, Y., and Xue, X. (2017). "Dsod: Learning deeply supervised object detectors from scratch," in *Proceedings of the IEEE international conference on computer vision*, (Venice, Italy: IEEE) 1919–1927.

Shi, W., Jiang, F., and Zhao, ,. D. (2017). "Single image super-resolution with dilated convolution based multi-scale information learning inception module," in *2017 IEEE international conference on image processing (ICIP)* (Beijing, China: IEEE), 977–981.

Shi, L., Zhang, F., Xia, J., Xie, J., Zhang, Z., Du, Z., et al. (2021). Identifying damaged buildings in aerial images using the object detection method. *Remote Sens.* 13 (21), 4213. doi: 10.3390/rs13214213

Shu, Q., Lai, H., Wang, L., and Jia, Z. (2021). Multi-feature fusion target re-location tracking based on correlation filters. *IEEE Access* 9, 28954–28964. doi: 10.1109/ACCESS.2021.3059642

Slaughter, D. C., and Harrell, R. C. (1989). Discriminating fruit for robotic harvest using color in natural outdoor scenes. *Trans. ASAE.* 32 (2), 757–0763. doi: 10.13031/2013.31066

Storey, G., Meng, Q., and Li, B. (2022). Leaf disease segmentation and detection in apple orchards for precise smart spraying in sustainable agriculture. *Sustainability* 14 (3), 1458. doi: 10.3390/su14031458

Tan, M., and Le, ,. Q. (2019). "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning* (Long Beach, California: PMLR), 6105–6114.

Tan, M., Pang, R., and Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. *In. Proc. IEEE/CVF. Conf. Comput. Vision Pattern Recogn.*, 10781–10790). doi: 10.1109/CVPR42600.2020.01079

Tong, K., Wu, Y., and Zhou, F. (2020). Recent advances in small object detection based on deep learning: A review. *Image. Vision Computing.* 97, 103910. doi: 10.1016/j.imavis.2020.103910

Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W. (2013). Selective search for object recognition. *Int. J. Comput. Vision* 104 (2), 154–171. doi: 10.1007/s11263-013-0620-5

Vedaldi, A., Gulshan, V., Varma, M., and Zisserman, A. (2009). "Multiple kernels for object detection," in *2009 IEEE 12th international conference on computer vision* (Kyoto, Japan: IEEE), 606–613.

Viola, P., and Jones, ,. M. (2001). "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition*, vol. 1. (Kauai, HawaiiIeee), I–I.

Wang, S. (2021). "Research towards yolo-series algorithms: Comparison and analysis of object detection models for real-time UAV applications," in *Journal of physics: Conference series*, vol. 1948. (Lisbon, Portugal: IOP Publishing), 012021.

Wang, M. F., and Cheng, L. (2004). Exposure of the shaded side of apple fruit to full sun leads to up-regulation of both the xanthophyll cycle and the ascorbate-glutathione cycle. *HortScience* 39 (4), 887A–8887. doi: 10.21273/hortsci.39.4.887a

Wang, X., Shrivastava, A., and Gupta, A. (2017). "A-fast-rcnn: Hard positive generation via adversary for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2606–2615. (Honolulu, Hawaii: IEEE)

Wei, S., Chen, Z., Wang, J., Zheng, X., Xiang, D., and Dong, ,. Z. (2021). "Object detection with noisy annotations in high-resolution remote sensing images using robust EfficientDet," in *Image and signal processing for remote sensing XXVII*, vol. 11862. (SPIE), 66–75.

Wei, Y., Wang, Z., and Xu, M. (2017). Road structure refined CNN for road extraction in aerial image. *IEEE Geosci. Remote Sens. Lett.* 14 (5), 709–713. doi: 10.1109/LGRS.2017.2672734

Wu, Q., Feng, D., Cao, C., Zeng, X., Feng, Z., Wu, J., et al. (2021). Improved mask r-CNN for aircraft detection in remote sensing images. *Sensors* 21 (8), 2618. doi: 10.3390/s21082618

Xiao, J., Ehinger, K. A., Hays, J., Torralba, A., and Oliva, A. (2016). Sun database: Exploring a large collection of scene categories. *Int. J. Comput. Vision* 119 (1), 3–22. doi: 10.1007/s11263-014-0748-y

Xu, D., Wang, L., and Li, F. (2021). Review of typical object detection algorithms for deep learning. *Comput. Eng. Appl.* 57 (8), 10–25.

Yan, Y., Tan, Z., and Su, N. (2019). A data augmentation strategy based on simulated samples for ship detection in RGB remote sensing images. *ISPRS. Int. J. Geo-Inform.* 8 (6), 276. doi: 10.3390/ijgi8060276

Yi, D., Su, J., and Chen, W. H. (2021). Probabilistic faster R-CNN with stochastic region proposing: Towards object detection and recognition in remote sensing imagery. *Neurocomputing* 459, 290–301.

Ying, Y., Jing, H., Tao, Y., Jin, J., Ibarra, J. G., and Chen, ,. Z. (2000). "Application of machine vision in inspecting stem and shape of fruits," in *Biological quality and precision agriculture II*, vol. 4203. (SPIE), 122–130.

Yu, Y., Zhang, J., Huang, Y., Zheng, S., Ren, W., Wang, C., et al. (2010). "Object detection by context and boosted HOG-LBP," in *ECCV workshop on PASCAL VOC.* (PASCAL)

Zhang, H., Goodfellow, I., Metaxas, D., and Odena, ,. A. (2019). "Self-attention generative adversarial networks," in *International conference on machine learning* (Long Beach, California: PMLR), 7354–7363.

Zhang, Q., Liu, Y., Gong, C., Chen, Y., and Yu, H. (2020). Applications of deep learning for dense scenes analysis in agriculture: A review. *Sensors* 20 (5), 1520. doi: 10.3390/s20051520

Zhang, L., Ma, Z., and Peng, X. (2022). "A remote sensing object detection algorithm based on the attention mechanism and faster r-CNN," in *Artificial intelligence in China* (Singapore: Springer), 336–344.

Zhang, S., Wen, L., Bian, X., Lei, Z., and Li, S. Z. (2018). "Single-shot refinement neural network for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (USA: IEEE), 4203–4212.

Zhao, Q., Sheng, T., Wang, Y., Tang, Z., Chen, Y., Cai, L., et al. (2019). M2det: A single-shot object detector based on multi-level feature pyramid network. *Proc. AAAI. Conf. Artif. Intell.* 33, 9259–9266. doi: 10.1609/aaai.v33i01.33019259

Zhong, Y., Han, X., and Zhang, L. (2018). Multi-class geospatial object detection based on a position-sensitive balancing framework for high spatial resolution remote sensing imagery. *ISPRS. J. Photogrammet. Remote Sens.* 138, 281–294. doi: 10.1016/j.isprsjprs.2018.02.014

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. (2017). Places: A 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (6), 1452–1464. doi: 10.1109/tpami.2017.2723009

Zhou, T., Zheng, L., Peng, Y., and Jiang, ,. R. (2022). "A survey of research on crowd abnormal behavior detection algorithm based on YOLO network," in *2022 2nd international conference on consumer electronics and computer engineering (ICCECE)* (Guangzhou, China: IEEE), 783–786.

Zhu, H., Zhang, P., Wang, L., Zhang, X., and Jiao, L. (2019). A multiscale object detection approach for remote sensing images based on MSE-DenseNet and the dynamic anchor assignment. *Remote Sens. Lett.* 10 (10), 959–967. doi: 10.1080/2150704X.2019.1633486

Zitnick, C. L., and Dollár, ,. P. (2014). "Edge boxes: Locating object proposals from edges," in *European Conference on computer vision* (Cham: Springer), 391–405.

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 04 January 2023
DOI 10.3389/fpls.2022.1089454

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University
(PSU), United States

REVIEWED BY
Uzair Aslam Bhatti,
Hainan University, China
Sijia Yu,
Rutgers, The State University of New
Jersey - Busch Campus, United States

*CORRESPONDENCE
Jinlin Xue
✉ xuejinlin@njau.edu.cn

SPECIALTY SECTION
This article was submitted to
Technical Advances in Plant Science,
a section of the journal
Frontiers in Plant Science

RECEIVED 04 November 2022
ACCEPTED 05 December 2022
PUBLISHED 04 January 2023

CITATION
Sun H, Wang B and Xue J (2023)
YOLO-P: An efficient method for pear
fast detection in complex orchard
picking environment.
Front. Plant Sci. 13:1089454.
doi: 10.3389/fpls.2022.1089454

YOLO-P: An efficient method for pear fast detection in complex orchard picking environment

Han Sun[1], Bingqing Wang[2] and Jinlin Xue[1]*

[1]College of Engineering, Nanjing Agricultural University, Nanjing, China, [2]Agricultural Machinery Information Center, Department of Agriculture and Rural Affairs of Jiangsu Province, Nanjing, China

Introduction: Fruit detection is one of the key functions of an automatic picking robot, but fruit detection accuracy is seriously decreased when fruits are against a disordered background and in the shade of other objects, as is commmon in a complex orchard environment.

Methods: Here, an effective mode based on YOLOv5, namely YOLO-P, was proposed to detect pears quickly and accurately. Shuffle block was used to replace the Conv, Batch Norm, SiLU (CBS) structure of the second and third stages in the YOLOv5 backbone, while the inverted shuffle block was designed to replace the fourth stage's CBS structure. The new backbone could extract features of pears from a long distance more efficiently. A convolutional block attention module (CBAM) was inserted into the reconstructed backbone to improve the robot's ability to capture pears' key features. Hard-Swish was used to replace the activation functions in other CBS structures in the whole YOLOv5 network. A weighted confidence loss function was designed to enhance the detection effect of small targets.

Result: At last, model comparison experiments, ablation experiments, and daytime and nighttime pear detection experiments were carried out. In the model comparison experiments, the detection effect of YOLO-P was better than other lightweight networks. The results showed that the module's average precision (AP) was 97.6%, which was 1.8% higher than the precision of the original YOLOv5s. The model volume had been compressed by 39.4%, from 13.7MB to only 8.3MB. Ablation experiments verified the effectiveness of the proposed method. In the daytime and nighttime pear detection experiments, an embedded industrial computer was used to test the performance of YOLO-P against backgrounds of different complexities and when fruits are in different degrees of shade.

Discussion: The results showed that YOLO-P achieved the highest F1 score (96.1%) and frames per second (FPS) (32 FPS). It was sufficient for the picking robot to quickly and accurately detect pears in orchards. The proposed method can quickly and accurately detect pears in unstructured environments. YOLO-P provides support for automated pear picking and can be a reference for other types of fruit detection in similar environments.

KEYWORDS

deep learning, pear, fruit detection, YOLOv5, convolutional neural network

1 Introduction

Pears are a common fruit which have rich nutrition and good taste. China grows the most pear trees, with a pear tree planting area that accounts for 67.30% of the global total pear tree planting area (Food and Agriculture Organization of the United Nations, 2022). However, the continuous loss of agricultural labor in recent years has led to a substantial increase in the cost of manual picking. The problem became more prominent after the COVID-19 pandemic (Nawaz et al., 2021). Therefore, efficient picking machines are a current research focus and an area of importance in orchard intelligence. Automated picking can increase the income of fruit farmers and promote economic development (Galvan et al., 2022).

Fruit detection is one of the most important steps for orchard picking robots working autonomously. At present, some scholars have used machine learning methods, especially based on color features, to detect fruits which are significantly different from the background color. For example, Si et al. (2010) proposed a method based on the red–green differential separation which used the contour formed by the shape of fruit to segment the red apple and green background. But this method is no longer effective when the target is similar to the background color, because some fruits (like some varieties of apples and mangoes) are green even when they are ripe. Xiang et al. (2012) used the curvature of overlapping tomato boundary lines to detect shaded tomatoes, but the accuracy for large shaded areas was only 76.9%. Compared with the deep learning technology that has developed rapidly in recent years, traditional machine learning methods exposed more limitations, such as low speed, low detection accuracy, and poor universality. Also, the designed algorithm can detect only a single target. As far as computers are concerned, the low-level features that machine learning uses are difficult to extract deep semantic information (Arrieta et al., 2020), making it unsuitable for online equipment and fruit detection in the complex and changeable environment of orchards.

Deep learning technology has been widely used in target detection in orchards. Object detection based on deep learning is mainly divided into a two-stage algorithm and a one-stage algorithm. Two-stage algorithms have been extensively studied due to high accuracy in the field of agriculture. Zhang et al. (2020) developed a detection system for apples and branches based on VGG-19 and Faster R-CNN for the vibration harvest. The mean average precision (mAP) for detecting apples was 82.4% and the fitting degree to the branches and trunks was over 90%. Tu et al. (2020) used a red, green, blue plus depth (RGB-D) camera to obtain the red, green, blue (RGB) image and depth information of passion fruit and combine them. A multi-scale-based Faster Region-based Convolutional Neural Network (R-CNN) network (MS-FRCNN) was proposed, which achieved an F1 score of 90.9%. Yan et al. (2019) improved the Region of interest (ROI) pooling layer of Faster R-CNN and combined VGG16 to detect 11 types of *Rosa roxbunghii* with different shapes; an average precision of 92.01% was obtained. The accuracy of two-stage detection is high. However, the huge number of parameters leads to increased computation costs and decreased detection speeds, which make it difficult to apply to online detection tasks.

The one-stage detection algorithm can greatly improve detection speed while maintaining detection accuracy because there is no process of generating candidate regions. Peng et al. (2018) used ResNet-101 to improve Single shot detector (SSD) for four kinds of fruit detection: citrus, apple, orange, and lychee. Compared with the original SSD, the average accuracy increased by 3.15%, and performance improved in shaded conditions. The "You Only Look Once" (Redmon et al., 2016; Redmon and Farhadi, 2017; Redmon and Farhadi, 2018; and Bochkovskiy et al., 2020) series of algorithms was born in 2015. This series has reached its fifth iteration and shows the trend and potential of continuous updating and strengthening. Due to the continuous integration of the latest network optimization tricks, both speed and accuracy can be maintained at a high level. The YOLO algorithm is considered to be one of the most successful one-stage detection networks. Bresilla et al., 2019 established an apple detection model based on YOLOv2. By adding computer-drawn images to assist training, the author found that synthesized images can reduce the position loss of the network and better locate the target. Pear detection was performed by transfer learning and the model achieved an F1 score of 0.87%. Liu et al. (2022) improved YOLOv3 to detect pineapples and calculated the 3D coordinates based on binocular vision cameras. The average precision (AP) value of fruit detection was 97.55% and the average relative error of binocular camera positioning was 24.4 mm. Xu et al. (2020) improved the backbone of YOLOv3, modified the batch normalization layer to group normalization, and used Soft-NMS to replace the original network management system (NMS) bounding box filter. The author proposed an image enhancement method to improve backlit images. The model finally got an F1 score of 97.7%. Parico and Ahamed (2021) improved YOLOv4, realizing fruit counting through a unique identity document (ID) method, which could meet the requirements of online operation. Zheng et al. (2022) used the improved YOLOv4 to detect tomatoes in a natural environment, and accuracy was improved by 1.52% compared with the original model. Jiang et al. (2022) integrated a non-local attention module and a convolutional block attention module (CBAM) into YOLOv4 to detect growing apples. Improved extraction ability of advanced features and perception of regions of interest. The test achieved an AP of 97.2%. Lu et al. (2022) used the improved YOLOv4 to calculate the number and the size of fruits on the whole apple tree. The network had the highest detection rate during fruit picking. This research enhanced the management ability of fruit trees. Zhang et al. (2022) proposed real-time strawberry detection network

(RTSD-Net) by improving YOLOv4-tiny's cross stage partial network (CSPNet). The detection of strawberries with the embedded system Jetson Nano had a detection speed of 25.2 FPS; hence, the real-time performance of the network was good. Chen et al. (2022) used YOLOv5 to detect citrus fruits and proposed a citrus ripeness detection algorithm that combined visual saliency with residual network (RESNet)-34. The accuracy of the model could reach 95.07%. Yan et al. (2021) used an improved YOLOv5 to detect apples and judge whether the fruit could be grasped by the picking machine. The model obtained a mAP of 86.75% and an F1 score of 87.49%. Yao et al. (2021) improved YOLOv5 by adding a small object detection layer, inserting a squeeze and excitation (SE) layer, and using a complete intersection over union (CIoU) loss function. The model achieved a mAP of 94.7% in an experiment detecting kiwifruit defects. Sozzi et al. (2022) utilized multiple networks to detect white grapes under different lighting conditions, against different backgrounds, and at different growth stages. The F1 score of YOLOv5x in the experiment was 0.76% and the detection speed was 31 FPS. Summarizing the above studies, using a one-stage algorithm such as YOLOv5 has become the most common method of fruit detection. However, the detection speed and accuracy of the network is still one of the problems to be solved urgently, and the existing research rarely considers the complex natural environment of the orchard.

YOLOv5 can achieve good results in datasets such as PASCAL VOC (Everingham et al., 2015) and COCO (Lin et al., 2014). However, for detection tasks in agriculture, the complete YOLOv5 network produces more performance redundancy. Even the light version of YOLOv5s struggles to achieve satisfactory results in orchards. At the same time, the background in orchards can be complex and fruits are easily shaded by other objects. The nighttime environment also has a significant impact on the effectiveness of detection. The existing YOLOv5 algorithm is facing great challenges, especially in low-performance devices, such as industrial computers, in online detection. Therefore, the purpose of this research was to design the YOLO-P network for fast and efficient detection of pears against complex backgrounds, in shade and during night picking. This method was based on YOLOv5. We designed a new module, named an inverted shuffle block, which can be applied in deeper layers to solve the problem of small targets missing in detection. We replaced some of the CBS structure in the YOLOv5 backbone with a shuffle block and an inverted shuffle block to form a new backbone. A CBAM was inserted into the new backbone to improve the ability to capture key features of pears. In addition, the activation functions in the remaining CBS of the entire network were replaced by Hard-Swish to improve the running speed. The detection effect of this method had been verified under different degrees of shade and background complexity during daytime and nighttime. YOLO-P can be used for fast and accurate detection of pears in orchards and can a references for other types of fruit detection in similar environments.

2 Pear detection framework

As one of the most mature, stable, and effective target detection algorithms currently available, YOLOv5 consists of three main parts: a backbone network, neck network, and classifier. The backbone is cross stage partial (CSP)-DarkNet53, which is used to extract different scale feature information from images. The neck network is path aggregation network (PANet) (Liu et al., 2018) with feature pyramid network (FPN), which is used to fuse feature information. The classifier outputs bounding boxes of large, medium, and small scales to complete the target detection. The

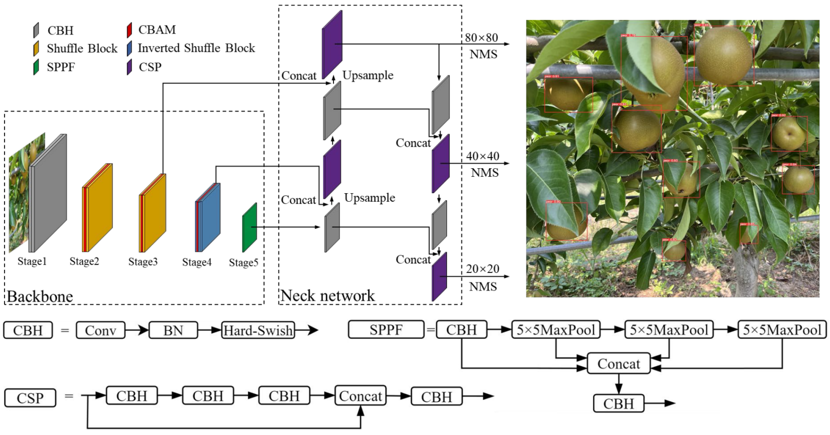


FIGURE 1
The network structure of YOLO-P.

YOLO-P method proposed in this paper is based on YOLOv5 and the structure is shown in Figure 1. The CBS structure in the second and third stages of the YOLOv5 backbone were replaced with a shuffle block. An inverted shuffle block was designed and used to replace the CBS structure of the fourth stage. This new backbone could extract features of distant pears in images more efficiently. CBAM was inserted in the new backbone to improve the important information perception capability of pears. The sigmoid linear unit (SiLU) activation function in the rest of the CBS structure was replaced with Hard-Swish to improve the running speed of the network. A weighted confidence loss function was designed to strengthen the detection effect of small targets. The details of the improvements are described below.

2.1 Backbone network

Ma et al. (2018) proposed that making the input and output feature maps equal, reducing convolution and element-wise operations, and integrating the network structure would help improve the inference speed of the network. Tan and Le (2020) suggested that increasing the depth of the network could result in richer features but may cause gradients to disappear. Increasing the width of the network results in finer-grained features, but it may fail to learn deep features. Therefore, it is necessary to balance the depth and width of the network to achieve the best results. Figure 2 shows the backbone of YOLO-P, built following the above lightweight network design principles, and lists the size of the output feature map (C×H×W). The input image size of the network is 3 × 640 × 640. The first stage is downsampling through two convolutional layers to obtain a feature map with a size of 64 × 160 × 160. The second and third stages use the shuffle block to extract features in the middle and shallow layers and downsample twice to obtain a feature map with a size of 256 × 40 × 40. The fourth stage uses the inverted shuffle block to extract features in deeper layers of the network and downsamples to obtain a 512 × 20 ×20 feature map. The fifth stage uses the improved spatial pyramid polling (SPPF) module in the deepest layer of the network to fuse the receptive field information of different scales. Finally, the SPPF output of the fifth stage and the output after the third and fourth stages' CBAMs are sent to the neck network of YOLO-P.

2.1.1 Feature extraction

The CSP-DarkNet53 of the YOLOv5 backbone uses a large number of CBS (Conv, Batch Norm, SiLU) structures which are suitable for target detection of complex features. However, this combination occupies a large amount of computation, and it is difficult for the application to run online in embedded devices. Therefore, this part needed to be optimized first. Xie et al. (2017) proposed the concept of group convolution in ResNeXt, which can effectively reduce the computational load of the network, as shown in Figure 3A. But there was no information exchange between groups and reduced the feature extraction ability. Based on the idea of group convolution, Ma et al. (2018) proposed a lightweight neural network ShuffleNetv2 that added channel shuffle in shuffle block. Figure 3B shows the group convolution process with channel shuffle. The channels between groups are shuffled before output. The resulting information exchange enables feature extraction to be done more efficiently.

2.1.1.1 Shuffle block

The shuffle block includes two cases where the stride is 1 and 2, respectively, as shown in Figure 4. First, the input feature matrix channels was divided into two groups by channel split and pass through two branches. If stride was 1, a residual structure containing 1×1Conv, 3×3DwConv and 1×1Conv in one branch was performed. If stride was 2 (downsampling), an additional 3×3DwConv and a 1×1Conv on the other branch was performed. The two branches were concatenated and the feature map was outputted through channel shuffle.

2.1.1.2 Inverted shuffle block

The residual structure in CSP-DarkNet53 is shown in Figure 5A. First, increases the dimension of the feature map increased and the dimension was reduced to extract features. However, there could be more zeros in the convolution kernel's

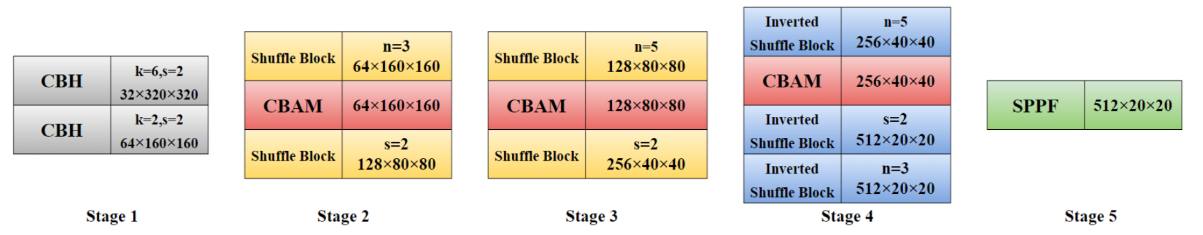


FIGURE 2
YOLO-P's backbone. k is convolutional kernel size, s is stride, and n is the number of module's repetitions. Unspecified k is 3, s is 1, and n is 1.

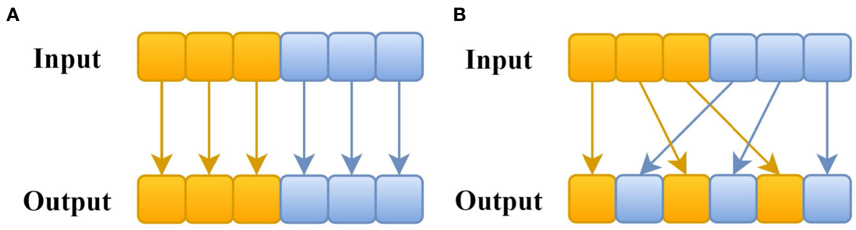


FIGURE 3
(A) Group Convolution; **(B)** Group Convolution with Channel Shuffle.

parameter of deeper layers. Directly increasing dimension brings difficulties to deep layers' feature extraction. In MobileNet (Howard et al., 2017), an inverted residual structure that first reduced the dimension of the feature map and then increased the dimension was proposed to extract more information, as shown in Figure 5B. Inspired by lightweight networks such as ShuffleNet and MobileNet, this study designed the inverted shuffle block used in deeper layers of network (the fourth stage of backbone), as shown in Figures 5C, D. The reversed structure made it easier to extract features from small objects. It was similar to shuffle block, but the residual structure of the branch was changed to an inverted residual structure. Similarly, if the stride was 2 (downsampling), an additional 3×3DwConv and a PwConv on the branch of the inverted residual structure was performed. The two branches were concatenated together and output the feature map was outputted through channel shuffle.

2.1.2 Attention module

Attention mechanism is a way to reinforce important information and suppress secondary information in a neural network. Application in the field of image object detection had proved attention mechanism's effectiveness. The CBAM is a lightweight soft attention module that is divided into channel and spatial parts (Woo et al., 2018). The channel attention module (CAM) when the inputs were C × H × W is shown in Figure 6A. We then performed global average pooling (GAP) and global maximum pooling (GMP) to the feature map in order to obtain two C × 1 × 1 feature matrices and send them to a multi-layer perceptron which has two layers. This was then summed and activated to get the channel attention vector. CAM focuses on what is in the feature map. The Spatial Attention Module (SAM) is shown in Figure 6B; we then performed GAP and GMP on the channel dimensions of the feature map to obtain a 2 × H × W feature matrix, then a 7 × 7 convolutional

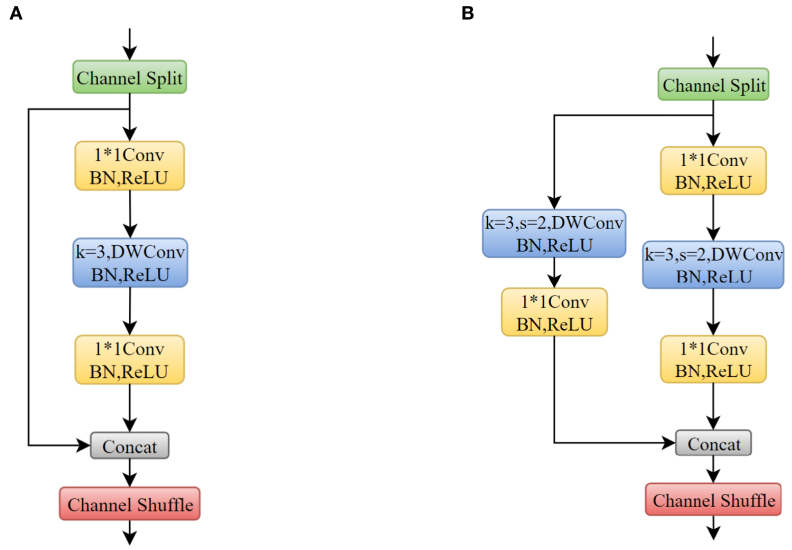


FIGURE 4
(A) Shuffle Block (s=1); **(B)** Shuffle Block (s=2). a * b means the width and height of the convolution kernel.

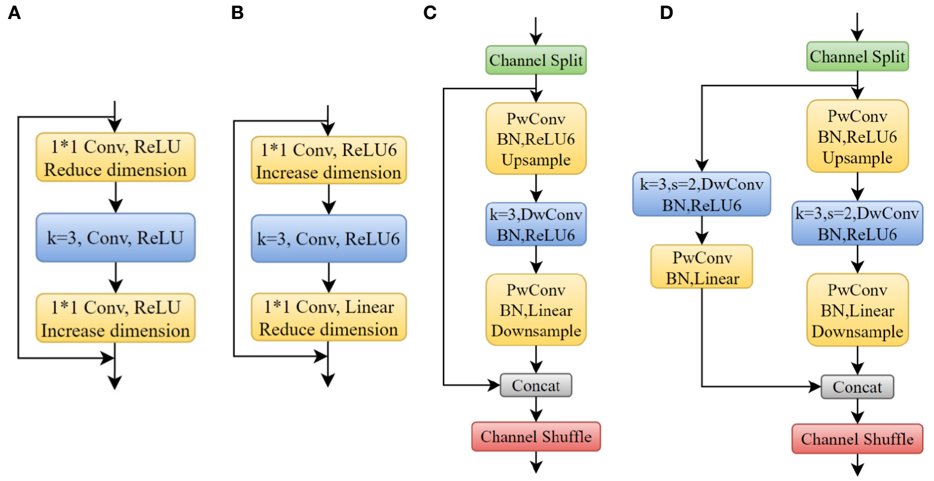


FIGURE 5
(A) Residual Block; **(B)** Inverted Residual Block; **(C)** Inverted Shuffle Block (s=1); **(D)** Inverted Shuffle Block (s=2). a * b means the width and height of the convolution kernel.

layer and activation to get a 1 × H × W spatial attention vector. The purpose of SAM is to more prominently express the characteristics of key regions. Each pixel of the feature map generates a weighted mask and outputs it, which reinforces where the key target is. Figure 6C shows CBAM. The channel attention vector obtained by CAM was first multiplied with input feature map. Then the resulting feature map was multiplied by spatial attention matrix obtained by SAM. Finally, the output of CBAM is obtained through the residual structure. The sequence of using CAM and then SAM to correct the feature maps was based on the characteristics of the human cerebral cortex, Woo et al. (2018) experiments also verified this. We applied CBAM to the second, third, and fourth stages of YOLO-P's backbone. Following experiments by Park et al.

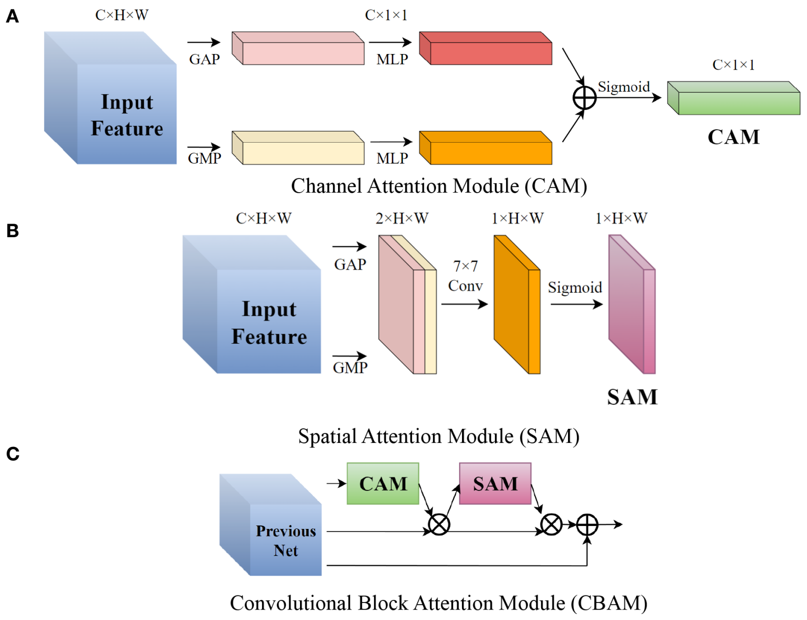


FIGURE 6
Schematic diagram of the CBAM structure in YOLO-P. **(A)** Channel Attention Module (CAM) **(B)** Spatial Attention Module (SAM) **(C)** Convolutional Block Attention Module (CBAM).

(2018), we inserted the attention module at the bottleneck of the network, i.e., before the downsampling layer. We then connected the output of CBAM to the neck network of YOLO-P for better feature fusion.

2.2 Activation function

The activation function of the network was mainly improved in two aspects. First was to replace the SiLU activation function for all CBS structures in YOLOv5 with Hard-Swish, and the second was to use the linear activation function for the last convolution layer in the inverted shuffle block.

First, all CBS structures in YOLOv5 used SiLU as an activation function. For the network applied to embedded devices, obviously the linear activation function could make the network faster. Hard-Swish (Howard et al., 2019) activation function was bounded up and down. The non-monotonic and piecewise linear characteristics reduced the amount of calculation. It was beneficial to eliminate saturation and make the feature expression ability better. All Conv, Batch Norm, Hard-swish (CBH) structures in YOLO-P's backbone and neck network used Hard-Swish as an activation function. Equation (1) is the Hard-Swish expression where x_{in} represents the input of the activation function. Second, ReLU was used as an activation function after most convolutional layers in the original shuffle block. However, due to the inverted residual structure of the inverted shuffle block, first an increase in dimension and then a reduction in dimension made the final output a low-dimensional feature vector. Although ReLU can better express high-dimensional features, it has serious loss of low-dimensional feature information (Sandler et al., 2018). In order to ensure the feature information was not lost and to better match the complete output of the inverted residual, each branch of the last convolutional layer of inverted shuffle block's used a linear activation function.

$$\text{Hard-Swish}(x_{in}) = x_{in}\frac{\text{ReLU6}(x_{in}+3)}{6} \qquad (1)$$

$$\text{ReLU6}(x_{in}) = \min(\max(x_{in},0),6) \qquad (2)$$

2.3 Loss function

Since the detection target type of the model was only pear, we did not set the class loss. The loss function of YOLO-P consists of confidence loss and location loss. Equation 3 shows confidence loss which was used to measure the probability that the predicted bounding box contained the real target. It was calculated by using binary cross entropy (BCE). In In Equations 3 and 4, I is the intersection area of the ground-truth box and predicted bounding box, U is the area of the union, C_i is the prediction confidence, N is the total number of samples, and spl represents all samples. According to the structure of the YOLO-P predictor, different weights K_1, K_2, and K_3 are adopted on the three prediction layers of small, medium, and large to strengthen the targets' detection ability of different scales. The confidence loss is shown in Equation 5. Since pears with a greater distance (small objects on the image) are more difficult to detect, we took K_1, $K_{2,}$ and K_3 as 6.0, 1.0, and 0.5 in YOLO-P, respectively.

$$L'_{conf} = -\frac{\sum_{i\in spl}\left(\frac{I}{U}\ln(C'_i) + (1-\frac{I}{U})\ln(1-C'_i)\right)}{N} \qquad (3)$$

$$C'_i = \text{sigmoid}(C_i) \qquad (4)$$

$$L_{conf} = 6.0\cdot L^{small}_{conf} + 1.0\cdot L^{medium}_{conf} + 0.5\cdot L^{large}_{conf} \qquad (5)$$

The location loss measures the location error between predicted bounding box and ground-truth box. Zheng et al. (2020) pointed out that the regression loss of bounding box should take the overlapping area, the distance between center points of the box, and the aspect ratio into account. In this study, we used CIoU loss as the location loss of YOLO-P, as shown in Equations 6–8, where w_{gt} and b_{gt} are the length and width of ground-truth box, w_p and b_p are the length and width of the predicted bounding box, d is the Euclidean distance between the predicted box and the ground-truth box, and c is the diagonal distance of the union of the predicted box and the ground-truth box. The CIoU loss can directly minimize the distance between two boxes (Zheng et al., 2020), so it has a faster convergence rate.

$$L_{loc} = 1 - \left(\frac{I}{U} - \left(\frac{d^2}{c^2} + \alpha v\right)\right) \qquad (6)$$

$$\alpha = \frac{v}{(1-\frac{I}{U}) + v} \qquad (7)$$

$$v = \frac{4}{\pi}\left(\arctan\frac{w_{gt}}{b_{gt}} - arctan\frac{w_p}{b_p}\right)^2 \qquad (8)$$

Combined with confidence loss and location loss, the loss function of YOLO-P is shown in Equation 9.

$$Loss = L_{conf} + L_{loc} \qquad (9)$$

3 Experiments

3.1 Dataset

Images required for the experiment were collected at a pear planting base located in Gaochun District, Nanjing City, Jiangsu Province, China. In this research, Akidzuki pears were used as detection targets. In August 2022, images were captured using a

Sony FDR-AX60 4K camera with a sensor type of 1/2.5 stacked complementary metal-oxide-semiconductor (CMOS), and a total of 533 images containing Akidzuki pears were captured as training samples while 118 images different from the training samples were taken for model testing. In addition to normal daytime lighting, the dataset also contained samples at night. The images at night were taken with the aid of a 1000 lm light source. Images contained shaded pears and complex backgrounds. We used ImageLabel to annotate images and perform data augmentation by randomly selecting three of the following augmentation strategies: (1) 50% probability of horizontal mirror flip, (2) 50% probability of vertical mirror flip, (3) random scaling 80–95%, (4) random brightness adjustment to 35–150%, (5) randomly added Gaussian blur, or (6) randomly added Gaussian noise. The images that could not be used for training were eliminated, and the training dataset was finally expanded to 5257 images. The expanded image inherited the previous annotations with 55496 labels in total. According to the ratio of 8 : 2, the dataset was divided into a training set and a validation set, which had 4206 and 1051 images, respectively. All images were stored in JPG format. The details of the dataset are shown in Table 1.

The difference in the distance between the camera and the pear will result in different scales of the collected images. The further the distance, the smaller the target. At this time, most areas of the image will be covered by useless background and increase the image's background complexity. The disordered background in the orchard makes it more challenging for the model to detect objects. Also, the number of smaller objects will increase significantly. According to the distance between the camera and the fruit, we divided the background of the image into three cases: uncomplicated, moderately complicated, and extremely complicated. Among them, the distance of 0.3–0.5 m was set for uncomplicated, while 0.5–1 m for moderately complicated, and farther than 1m for extremely complicated.

The pears on the fruit trees photographed by camera were sometimes shaded by leaves or other objects, and there were also cases where the pears might be shaded by each other. The shaded target would bring difficulties to detection. In order to specifically verify the reliability of YOLO-P in detecting such targets, we proposed a method for calculating the pears' shaded degree. K_s was used to evaluate the degree of shade, which was the ratio of the shaded area to the total area of the pear in images. According to our previous experiments, it was extremely difficult to detect when Ks was higher than 0.6, so only the case of $K_s< 0.6$ was considered in this study, as shown in Table 2.

3.2 Experimental environment and parameters

Training of YOLO-P was carried out in a Windows 10 environment. The graphics processing unit (GPU) was Nvidia GeForce RTX 3060, the central processing unit (CPU) was AMD Ryzen 7 5800, and the memory was 32 GB. We used the Pytorch1.8.1 framework, CUDA 11.1 computing platform and CUDNN 8.1 deep neural network acceleration library.

The momentum decay and weight decay of all models during training were designed to be 0.9 and 0.0005, respectively, and the initial learning rate was 0.01. At the same time, the cosine annealing algorithm was used to optimize the learning rate. We used three rounds of epoch to warmup in order to stabilize the early training model. The warmup momentum was 0.8 and the batch size was set to 32. We used Adam as the optimizer with 500 training epochs. To prevent overfitting, the model would automatically stop training if there was no accuracy improvement in the last 50 training epochs.

3.3 Evaluation indicators

A variety of indicators could be used to evaluate the quality of the model in different experimental contexts, such as precision (P), recall (R), F1 score, AP, mAP, FPS, FLOPs, model volume, etc. The higher the P, R, F1 score, and AP, the more reliable the model would be. Their computation consists of true positives (TP), false positives (FP), and false negatives (FN), as shown in Equations 10-13 respectively. The intersection over union (IoU) threshold in AP took 0.5 (AP@0.5). It is worth mentioning that there was only one category of pears in this study, so AP and mAP were equal.

$$P = \frac{TP}{TP + FP} \tag{10}$$

$$R = \frac{TP}{TP + FN} \tag{11}$$

$$\mathrm{F1} = \frac{2PR}{P + R} \tag{12}$$

$$AP = \int_0^1 P(R)\mathrm{d}R \tag{13}$$

TABLE 1 Details of the pear image dataset.

	Uncomplicated background	Moderately complex background	Extremely complex background	Daytime	Nighttime	Total images
Number of images	1209	1630	2418	3680	1577	5257

TABLE 2 Index of shaded pear's degree in the dataset.

	Evaluation indicators
Not shaded or slightly shaded	$0 \leq K_s \leq 0.2$
Medium shaded	$0.2 < K_s \leq 0.4$
Serious shaded	$0.4 < K_s \leq 0.6$

Model volume refers to the size of weight file obtained after training. FPS refers to the number of images the model can process per second. FLOPs is the total floating-point operations of the model, as shown in Equation (14), where N represents all convolutional layers, L_i and C_i are the output feature layer size and number of channels of the current layer, respectively, K_i is the number of convolution kernels of the current layer, and $C_{i\text{-}1}$ is the number of input channels of the current layer. Like the model volume, the higher the FLOPs and the more complex the model, the slower the operation speed and the lower the FPS.

$$\text{FLOPs} = \sum_{i \in [1,N]} L_i^2 \times K_i^2 \times C_i \times C_{i-1} \tag{14}$$

3.4 Experiments results

3.4.1 Model comparison experiments

Since YOLO-P is a one-stage model, the purpose is to run at high speed on low-performance devices, so it is not meaningful to compare with the two-stage model. We selected several mainstream lightweight networks including RegNet, MobileNetv3, and EfficientNetv2 to compare with YOLO-P. RegNet (Radosavovic et al., 2020) optimized design space of the network to obtain optimal solution. MobileNetv3 (Howard et al., 2019) added squeeze excitation attention to the inverted residual module, and reduced the amount of computation without losing accuracy by improving the structure of the last stage. EfficientNetv2 (Tan and Le, 2021) improved feature extraction efficiency by introducing Fused-MBConv. In order to make the model volume more similar to YOLO-P, we replaced the backbone of YOLOv5s with the above three networks. At the same time, the classic YOLOv5s model was used for comparison. In the model comparison experiments of this section, we selected P, AP@0.5, FLOPs, and module volume as evaluation indicators. The test results are shown in Table 3.

From the data in Table 3, it can be seen that YOLO-P achieved the best AP in section's experiments, which was 97.6% and it was 1.8% higher than its original network. RegNet-YOLO had the lowest AP. Although the FLOPs of YOLO-P was not the lowest, we got the smallest model volume which was only 8.3 MB. Compared with YOLOv5s, it was 39.4% smaller. MobileNet-YOLO had the lowest FLOPs of only 7.3 G, which is related to the reduction of last stage in this network. Model comparison experiments showed that the combination of shuffle block and inverted shuffle block was reliable. The proposed YOLO-P model could detect pears in orchards with a smaller model volume and high accuracy.

3.4.2 Ablation experiments

We conducted ablation experiments on YOLO-P and discussed the performance improvement of YOLOv5s with new modules and new structures. New operations included shuffle block, inverted shuffle block, Hard-Swish activation function used in CBH, and inserted CBAM. We designed four sets of experiments in this section. In the T1 experiment, the four CBS groups and their corresponding downsampling modules in the YOLOv5s backbone network were replaced with shuffle blocks. In the T2 experiment, the four CBS groups and their corresponding downsampling modules in the YOLOv5s backbone network were replaced with an inverted shuffle block. The number of module repetitions in both T1 and T2 was the same as YOLO-P. In the T3 experiment, all four CBS groups were replaced with the same shuffle block and inverted shuffle block as YOLO-P. The T4 experiment used Hard-Swish on the basis of the T3. Finally, full YOLO-P network was CBAM's insertion. In the model ablation experiments of this section, we selected precision, AP0.5and FLOPs as evaluation indicators: the test results are shown in Table 4.

It can be seen from Table 4 that only using a shuffle block or an inverted shuffle block in the backbone was not as good as the AP obtained by YOLOv5s, because the inverted structure is not suitable for shallow networks. Also, the use of upsampling in deep networks reduced the ability to detect small objects. We

TABLE 3 Results of model comparison experiments.

	Precision (%)	AP@0.5 (%)	FLOPs (G)	Model Volume (MB)
RegNet-YOLO	92.8	90.3	13.4	14.6
MobileNet-YOLO	95.4	95.2	**7.3**	9.2
EffiecientNet-YOLO	95.6	95.0	14.4	17.8
YOLOv5s	96.0	95.8	15.9	13.7
YOLO-P	**98.1**	**97.6**	10.1	**8.3**

Bold means the best score achieved in that category.

TABLE 4 Results of ablation experiments.

	Shuffle Block	Inverted Shuffle Block	Hard-Swish	CBAM	Precision (%)	AP@0.5 (%)	FLOPs (G)
YOLOv5s					96.0	95.8	15.9
T1	√				94.3	93.9	10.6
T2		√			94.8	94.7	**9.3**
T3	√	√			96.2	95.9	10.0
T4	√	√	√		96.9	96.5	10.0
YOLO-P	√	√	√	√	**98.1**	**97.6**	10.1

Bold means the best score achieved in that category.

used different structures in shallow and deep layers of the network to deal with different sized targets. It would be easier to detect targets with inconspicuous feature expressions by combining the characteristics and advantages of the two modules. The AP obtained by the T3 experiment was similar to original network, which was only 0.1% higher than YOLOv5s. However, due to the influence of the channel shuffle, the calculation amount of model was reduced which made the FLOPs reduce, and the detection speed was also be improved. The model's AP was improved by 0.6% after optimizing the SiLU activation function to Hard-Swish. On this basis, the feature extraction ability was further strengthened by inserting CBAM, which made AP increase by 1.1%, reaching 97.6%. The comparison of four sets of experiments above proved that the proposed improved application is feasible in the pear detection network.

3.4.3 Pear detection experiments

Pear detection experiments were carried out on an industrial computer with limited computing resources in order to verify the feasibility of YOLO-P online work. We chose the embedded industrial computer of model DTB-3049-H310 produced by Dongtintech. The operating environment was Ubuntu 18.04, CPU was i7 9700 with 16 GB memory and it was without GPU. Detection experiments considered many situations of an intelligent picking robot in orchard. Different types of picking machinery working at different distances resulted in different degrees of background complexity. Dense foliage made pears shaded. For efficiency purposes, picking should be done not only during the daytime, but also at night. The experiment used 59 daytime and 57 nighttime pear images that different from the training samples, with a total of 649 labels. Three models (YOLOv5s, MobileNet-YOLO, YOLO-P) were selected in this section's experiments. The models' detection abilities under different background complexities and different degrees of shaded were respectively studied. We set the confidence threshold of the detection model to 0.4, i.e., confidence below 0.4 was not annotated in the image. The P, R, and F1 score were calculated by counting TP, FP and FN. FPS of the model operation were recorded. The overall test results are shown in Table 5. Pears that were detected by YOLO-P are shown in Figure 7.

3.4.3.1 Experiments during daytime

There was sufficient sunlight during the daytime: pears were easily detected when the background was not complicated (the target was obvious) and the degree of shade was low. However, the shade led to reduction of features or the image taken from a long distance led to fewer pixels on the target which would weaken the feature representation of pears. In this section, detection experiments were carried out on pears in different situations according to the proposed method of calculating background complexity and shaded degree under sufficient light during daytime.

First, experiments of different background complexities were carried out. We measured the background complexity by the distance between camera and pears. The F1 score obtained in this section is shown in Table 6. The experiments images are shown in Figure 8. Figures 8A–C are images of pears in uncomplicated backgrounds. YOLO-P detected all objects accurately. There were two false detections in YOLOv5s. MobileNet-YOLO did not detect a pear that had been shaded below. Figures 8D–F are images of pears in moderately complex backgrounds. All three networks detected all targets, but both YOLOv5s and MobileNet-YOLO mistakenly marked a dead leaf as a pear. Figures 8G–I are images of pears in extremely complex backgrounds. The environment of these images was relatively harsh. There were 15 valid targets in the image and many pears were seriously shaded. MobileNet-YOLO missed four targets.

TABLE 5 Result of Akidzuki pear detection experiments.

	Precision (%)	F1 (%)	FPS
MobileNet-YOLO	90.1	89.6	28
YOLOv5s	94.8	92.8	19
YOLO-P	**97.3**	**96.1**	**32**

Bold means the best score achieved in that category.

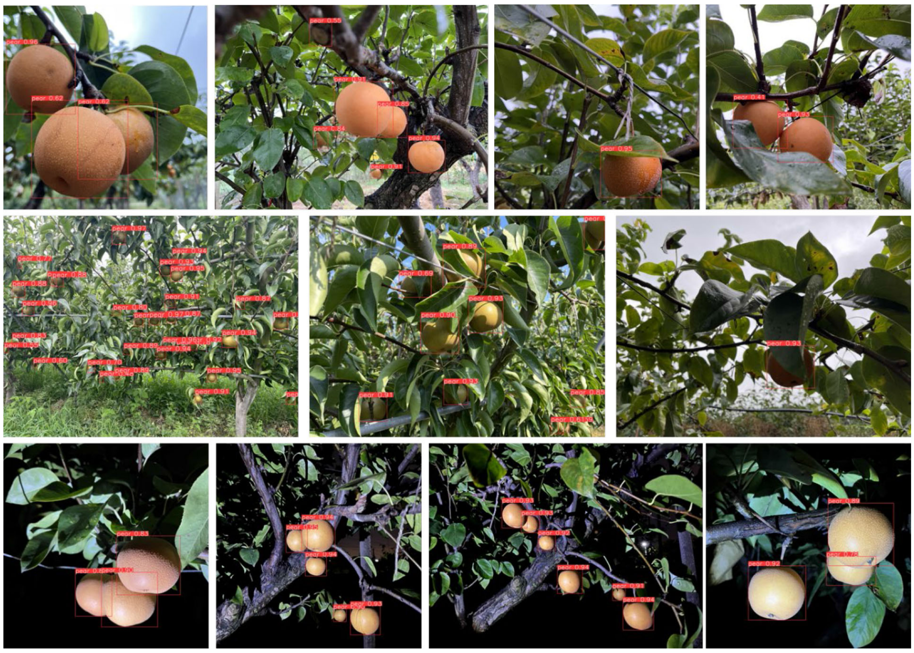


FIGURE 7
The detecting effect of Akidzuki pear in complex environment.

YOLOv5s and YOLO-P both missed two targets, but YOLOv5s had two false detections. It can be seen from the experiment in this section that YOLO-P had strong anti-interference ability. Although YOLOv5s could also detect targets accurately, it often misidentified other objects such as dead leaves as pears due to similar features. Even in the case of extremely complex backgrounds and few pixels, YOLO-P hardly had false detections and missed detections.

In the experiment of different degrees of shade, the degree was measured by the shaded area of pears. The more severely shaded, the more difficult feature expression of pears in the image, and the more difficult to it was detect accurately. The F1 score obtained in this section is shown in Table 7. The experimental images are shown in Figure 9. Figures 9A–C are not shaded or slightly shaded pear images and Figures 9D–F are medium-shaded pear images. As can be seen from the figure, all three networks could detect the shaded pears, but YOLO-P always had the highest confidence in detecting shaded targets. Figures 9G–I are serious-shaded pear images. Only MobileNet-YOLO failed to detect serious shaded objects. YOLO-P was more stable against shade problems during the day due to its higher confidence.

3.4.3.2 Experiments during nighttime

The problem of nighttime detection is the presence of shadows. Shadows are very similar in color to the background, so shadows can also be considered as a form of detection. Shadows may have pixel values very similar to the external environment due to the uncertain lighting direction. The boundaries between the outline of pear and the environment become blurred. Therefore, detecting pears at night will be more difficult than during the day. In this section, detecting experiments were carried out under the illumination of an auxiliary light source at night.

TABLE 6 F1 score (%) in different background complexities experiments during daytime.

	Uncomplicated background	Moderately complex background	Extremely complex background	Average
YOLOv5s	95.5	95.1	93.2	94.6
MobileNet-YOLO	92.5	91.8	89.5	91.3
YOLO-P	**96.9**	**96.6**	**95.5**	**96.3**

Bold means the best score achieved in that category.

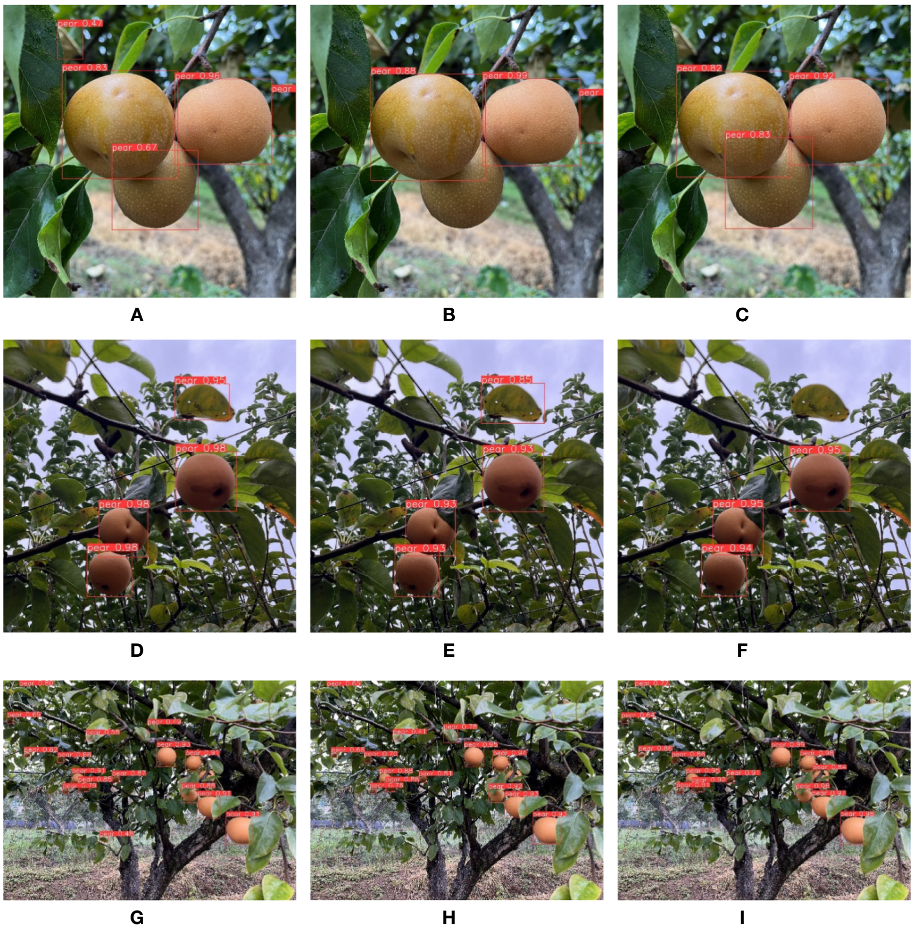


FIGURE 8
From left to right are the detection effects of YOLOv5s, MobileNet-YOLO and YOLO-P. **(A–C)** Uncomplicated background; **(D–F)** Moderately complex background; **(G–I)** Extremely complex background.

The F1 scores obtained by the experiments of different background complexity at night are shown in Table 8. The experiment images are shown in Figure 10. Figures 10A–C are images of pears in an uncomplicated background. It can be seen from the figure that MobileNet-YOLO missed a target. Both YOLOv5s and YOLO-P detected each objects successfully. But YOLOv5s had lower confidence and the location of the bounding box was not accurate. Figures 10D–F are images of pears in moderately complex backgrounds. The situation was similar to the previous group; although both YOLOv5s and YOLO-P detected all targets, YOLO-P had significantly higher confidence. Figures 10G–I are images of pears in extremely complex background. Both YOLOv5s and YOLO-P had a false detection, but they all detected a target in the middle of the image which was interfered with by a more complex shadow, while MobileNet-YOLO did not detect this target. The unclear edge of pears caused by nighttime illumination is one of the important reasons that affect the stability of the model. It can be concluded from the

TABLE 7 F1 score (%) in different shaded degrees experiments during daytime.

	Not shaded or slightly shaded	Medium shaded	Serious shaded	Average
YOLOv5s	94.8	94.3	94.2	94.4
MobileNet-YOLO	94.5	93.4	90.7	92.9
YOLO-P	**97.2**	**96.6**	**96.4**	**96.7**

Bold means the best score achieved in that category.

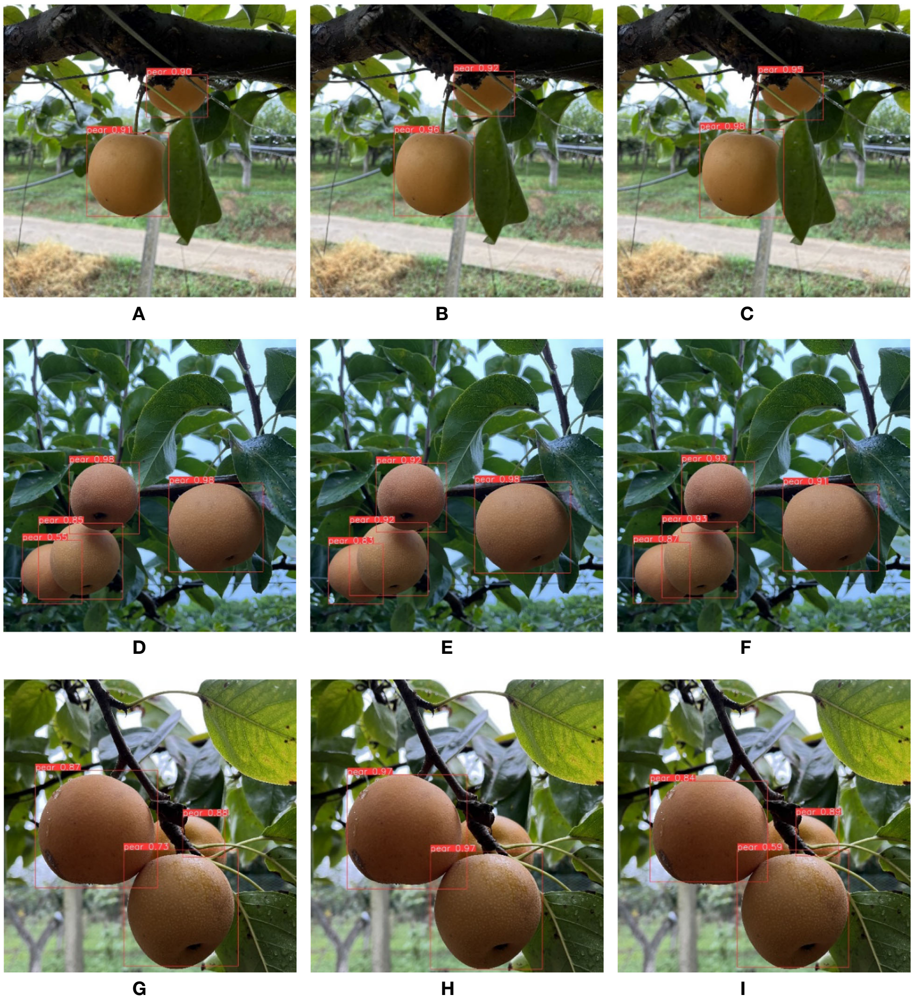


FIGURE 9
From left to right are the detection effects of YOLOv5s, MobileNet-YOLO and YOLO-P. **(A–C)** No shaded or slightly shaded; **(D–F)** Medium shaded; **(G–I)** Serious shaded.

experiments that the performance of YOLO-P is better than other models in the complex background situation at night.

The F1 scores obtained by the experiments of different shade degrees at night are shown in Table 9. The experiment images of at night are shown in Figure 11. Figures 11A–C are not shaded or slightly shaded pear images. All three networks detected the target accurately. Figures 11D–F are medium-shaded pear images. YOLOv5s and YOLO-P detected all targets. Neither of the two shaded fruits was successfully detected by MobileNet-YOLO. Figures 11G–I are serious-shaded pear images. YOLOv5s

TABLE 8 F1 score (%) in different background complexities experiments during nighttime.

	Uncomplicated background	Moderately complex background	Extremely complex background	Average
YOLOv5s	92.8	92.5	88.9	91.4
MobileNet-YOLO	87.3	86.8	86.4	86.8
YOLO-P	**97.8**	**95.6**	**93.9**	**95.8**

Bold means the best score achieved in that category.

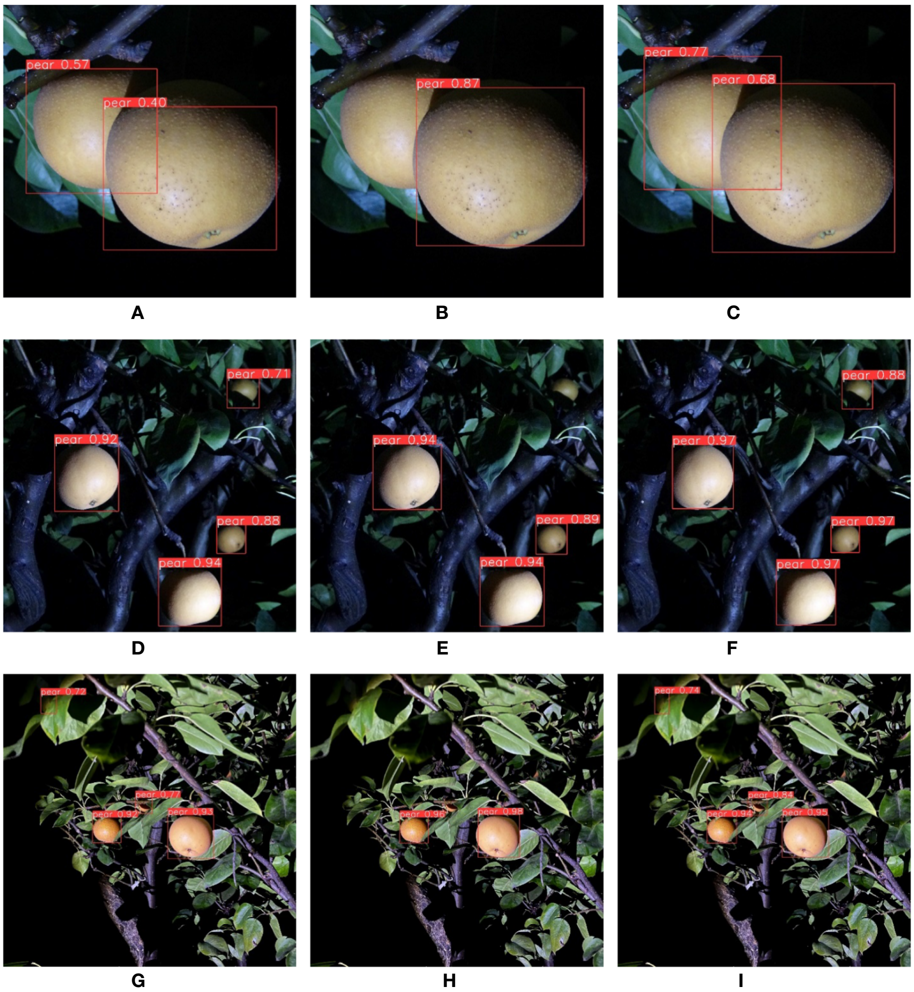


FIGURE 10
From left to right are the detection effects of YOLOv5s, MobileNet-YOLO and YOLO-P. **(A–C)** Uncomplicated background; **(D–F)** Moderately complex background; **(G–I)** Extremely complex background.

and YOLO-P detected all pears. But MobileNet-YOLO only detected one of the two targets. Likewise, YOLO-P had the highest confidence in this section's experiment.

It can be seen that YOLO-P could accurately detect pears in various situations according to the above experiments. Although YOLOv5s could also accurately detect most targets, there were many false detections and lower confidence. Another weakness is that YOLOv5s needs more computing resources. MobileNet-YOLO was difficult to extract high-semantic features due to the insufficient feature extraction ability. Therefore, there was a high degree of missed detection which is especially evident in the case of high complexity and seriously shaded. In summary, YOLO-P

TABLE 9 F1 score (%) in different shaded degrees experiments during nighttime.

	Not shaded or slightly shaded	Medium shaded	Serious shaded	Average
YOLOv5s	91.5	90.6	90.2	90.8
MobileNet-YOLO	89.2	86.9	85.7	87.3
YOLO-P	**95.7**	**95.6**	**95.1**	**95.5**

Bold means the best score achieved in that category.

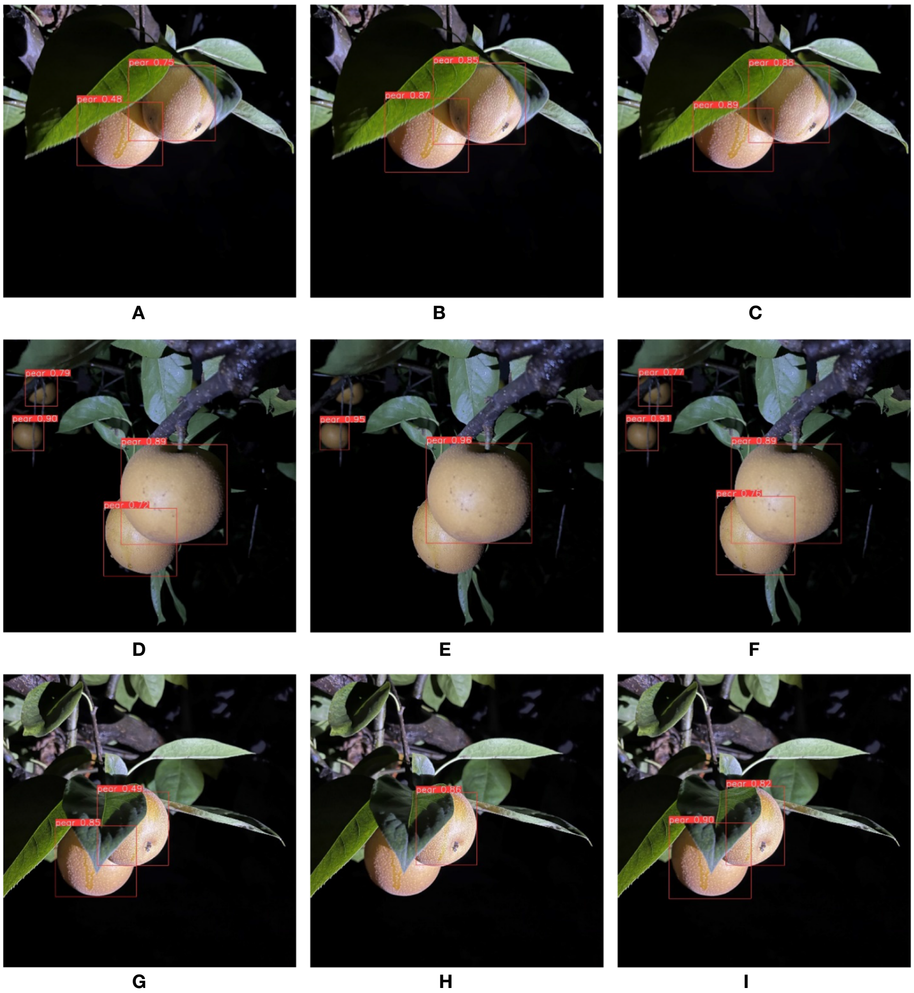


FIGURE 11
From left to right are the detection effects of YOLOv5s, MobileNet-YOLO and YOLO-P. **(A–C)** No shaded or slightly shaded; **(D–F)** Medium shaded; **(G–I)** Serious shaded.

had the best reliability in detecting pears in complex environments. YOLO-P had the best reliability in detecting pears under complex environments.

4 Discussion

Extensive research work has proved that building more complex datasets is the key to further improving the accuracy and robustness of deep learning models. For the automatic picking work in orchards, there are different shade patterns and backgrounds for each step the robot moves. Therefore, the scene it sees is far more complex than the images used for training. Although we collected as many complex images as possible, the variety of shaded fruits is too numerous. If a similar pattern of shaded fruits is not trained, the model will most likely be unable to recognize this object (although it looks remarkably easy to recognize). In this study, only the case where the fruit was shaded below 60% was considered. More diverse image data should be obtained in future work to deal with the more severely shaded fruit detection.

In experiments at night, we found that pixels in shadow-covered locations might be very similar to the outside environment, especially when the angle of the light source to the target was uncertain. This is one of the most important barriers to detecting pears at night. At present, some studies (Xu et al., 2020; Wang et al., 2022) have proved that the use of image enhancement can improve the accuracy of deep learning in harsh environments, especially in low light. If the models use some kind of machine learning method to preprocess the image and enhance the target boundary then input to neural network for recognition, the night detection ability of the model could be further improved.

Furthermore, only the detection of fully ripe pears was investigated in this study. In practice, picking in orchards should be done in batches. There may be cases that some pears are mature and some are not. Therefore, the intelligent detection of fruit ripeness is also one of the main research directions. Fruit ripeness can be judged by directly detecting the appearance characteristics (Chen et al., 2022). In addition, remote sensing can also be used for detection. From a macro perspective, the leaves of pear trees will become darker during the ripening season, and the fruits on pear trees may also have different characteristics. Remote sensing detection combined with deep learning may better judge fruit ripeness, thereby helping intelligent picking in orchards.

5 Conclusions

The cost of manual picking has gradually increased with the continuous loss of agricultural labor. In order to improve the economic benefits of fruit farmers and the automation degree of orchards, it is imperative to study the intelligent picking technology. Accurate and fast fruit detection is one of the most critical steps for orchard robot automatic picking. The robustness of fruit detection in complex backgrounds and shaded environments is a key factor affecting the work of automated picking robots. This study aimed to improve the accuracy and speed of fruit detection by improving the existing methods. The results will improve the reliability of pear detection in unstructured environments and enable it to be applied to online detection tasks in an industrial computer.

Based on YOLOv5, we proposed a deep learning model YOLO-P for detecting pears in complex orchard environments. The research carried out the following design and improvements. A new module named inverted shuffle block was designed. The inverted shuffle block was used in deeper networks. Combined with the shuffle block used in the shallow networks, the backbone of YOLOv5 was reconstructed. The new backbone had a good ability to detect small targets. The activation function was replaced with Hard-Swish to reduce the computational load of the network. CBAM was inserted to improve the capture of key information. Finally, a weighted loss function was designed to further improve the feature extraction ability of small targets.

We used the Akidzuki pears as detection object of the model. We compared YOLO-P with some mainstream lightweight models. The detection effect of YOLO-P was significantly better than others. Compared with the original YOLOv5s, AP increased from 1.8% to 97.6%, and the volume of the model was compressed by 39.4% to only 8.3MB. Ablation experiments on YOLO-P demonstrated the effectiveness of these improvements. In daytime and nighttime Akidzuki pear detection experiments, we used an embedded industrial computer to test the performance of the model under different background complexities and different shade degrees. The experimental results showed that YOLO-P achieved the highest F1 score and FPS of 96.1% and 32, respectively which were 3.3% and 68.4% higher than YOLOv5s, respectively. The YOLO-P developed in this paper can provide technical support for intelligent picking in pear orchards, and can also provide a reference for other types of fruit detection in complex environments.

In this research, we only considered the situation that the degree of shade is less than 60%. In the real orchard environment, there may be fruits that are more seriously shaded and difficult to be detected. Efficiently obtain high-quality and more abundant data to train models will be our next research goal. In detection at night, border of the fruit may be similar to the environment due to the lack of light. This is one of the reasons why the accuracy at night is lower than that during the day. In follow-up research, we will consider using image enhancement algorithms to further improve the reliability of the model.

Data availability statement

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

Author contributions

HS designed the model, obtained the pear images, designed the experiments, and carried it out. BW guided the research of this paper, processed the required images, and optimized the experiment scheme. JX also guided the research, determined basic framework of the research, revised the manuscript several times, and provided the final version. All authors contributed to the article and approved the submitted version.

Funding

This research was supported by the earmarked fund for Jiangsu Agricultural Industry Technology System (Grant No. JATS-2022-483) and Project of modern agricultural machinery equipment and technology demonstration and promotion of Jiangsu province (Grant No. NJ2021-38).

Acknowledgments

We would like to express our gratitude to Wen Ma and Fan Liu of Nanjing Huashan Modern Horticulture Laboratory for their invaluable assistance in collecting the image data required for the experiments.

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

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., et al. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. fusion* 58, 82–115. doi: 10.1016/j.inffus.2019.12.012

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

Bresilla, K., Perulli, G. D., Boini, A., Morandi, B., Corelli Grappadelli, L., and Manfrini, L. (2019). Single-shot convolution neural networks for real-time fruit detection within the tree. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.00611

Chen, S., Xiong, J., Jiao, J., Xie, Z., Huo, Z., and Hu, W. (2022). Citrus fruits maturity detection in natural environments based on convolutional neural networks and visual saliency map. *Precis. Agric.* 23, 1515–1531. doi: 10.1007/S11119-022-09895-2

Everingham, M., Eslami, S. M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vision* 111 (1), 98–136. doi: 10.1007/s11263-014-0733-5

Food and Agriculture Organization of the United Nations (2022) *E. coli*. Available at: https://www.fao.org/statistics/en/ (Accessed October 23, 2022).

Galvan, L. P. C., Bhatti, U. A., Campo, C. C., and Trujillo, R. A. S. (2022). The nexus between CO2 emission, economic growth, trade openness: Evidences from middle-income trap countries. *Front. Environ. Sci.* 10 (7). doi: 10.3389/fenvs.2022.938776

Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., et al. (2019). Searching for mobilenetv3. *arXiv.* doi: 10.48550/arXiv.1905.02244

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand,, et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv.* doi: 10.48550/arXiv.1704.04861

Jiang, M., Song, L., Wang, Y., Li, Z., and Song, H. (2022). Fusion of the YOLOv4 network model and visual attention mechanism to detect low-quality young apples in a complex environment. *Precis. Agric.* 23 (2), 559–577. doi: 10.1007/S11119-021-09849-0

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "Microsoft Coco: Common objects in context," in *European Conference on computer vision* (Cham: Springer), 740–755. doi: 10.1007/978-3-319-10602-1_48

Liu, T. H., Nie, X. N., Wu, J. M., Zhang, D., Liu, W., Cheng, Y. F., et al. (2022). Pineapple (Ananas comosus) fruit detection and localization in natural environment based on binocular stereo vision and improved YOLOv3 model. *Precis. Agric.* 23, 1–22. doi: 10.1007/s11119-022-09935-x

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). Path aggregation network for instance segmentation. *arXiv.* doi: 10.48550/arXiv.1803.01534

Lu, S., Chen, W., Zhang, X., and Karkee, M. (2022). Canopy-attention-YOLOv4-based immature/mature apple fruit detection on dense-foliage tree architectures for early crop load estimation. *Comput. Electron. Agric.* 193, 106696. doi: 10.1016/J.COMPAG.2022.106696

Ma, N., Zhang, X., Zheng, H. T., and Sun, J. (2018). Shufflenet v2: Practical guidelines for efficient cnn architecture design. *arXiv.* doi: 10.48550/arXiv.1807.11164

Nawaz, S. A., Li, J., Bhatti, U. A., Bazai, S. U., Zafar, A., Bhatti, M. A., et al. (2021). A hybrid approach to forecast the COVID-19 epidemic trend. *PloS One* 16 (10), e0256971. doi: 10.1371/journal.pone.0256971

Parico, A. I. B., and Ahamed, T. (2021). Real time pear fruit detection and counting using YOLOv4 models and deep SORT. *Sensors* 21 (14), 4803. doi: 10.3390/S21144803

Park, J., Woo, S., Lee, J. Y., and Kweon, I. S. (2018). Bam: Bottleneck attention module. *arXiv.* doi: 10.48550/arXiv.1807.06514

Peng, H., Huang, B., Shao, Y., Li, Z., Zhang, C., Chen, Y., et al. (2018). General improved SSD model for picking object recognition of multiple fruits in natural environment. *Trans. Chin. Soc. Agric. Eng.* 34 (16), 155–162. doi: 10.11975/j.issn.1002-6819.2018.16.020

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., and Dollár, P. (2020). Designing network design spaces. *arXiv.* doi: 10.48550/arXiv.2003.13678

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*. Las Vegas, NV, USA: IEEE, 779–788. doi: 10.1109/CVPR.2016.91

Redmon, J., and Farhadi, A. (2017). "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Honolulu, HI, USA: IEEE, 7263–7271. doi: 10.1109/CVPR.2017.690

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

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. *arXiv.* doi: 10.48550/arXiv.1801.04381

Si, Y., Qiao, J., Liu, G., Gao, R., and He, B. (2010). Recognition and location of fruits for apple harvesting robot. *Trans. Chin. Soc. Agric. Machinery* 41 (9), 148–153. doi: 10.3969/j.issn.1000-1298.2010.09.030

Sozzi, M., Cantalamessa, S., Cogato, A., Kayad, A., and Marinello, F. (2022). Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms. *Agronomy* 12 (2), 319. doi: 10.3390/agronomy12020319

Tan, M., and Le, Q. (2020). Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv.* doi: 10.48550/arXiv.1905.11946

Tan, M., and Le, Q. (2021). Efficientnetv2: Smaller models and faster training. *arXiv.* doi: 10.48550/arXiv.2104.00298

Tu, S., Pang, J., Liu, H., Zhuang, N., Chen, Y., Zheng, C., et al. (2020). Passion fruit detection and counting based on multiple scale faster r-CNN using RGB-d images. *Precis. Agric.* 21 (5), 1072–1091. doi: 10.1007/s11119-020-09709-3

Wang, Y., Xie, W., and Liu, H. (2022). Low-light image enhancement based on deep learning: a survey. *Optical Eng.* 61 (4), 40901. doi: 10.1117/1.OE.61.4.040901

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). Cbam: Convolutional block attention module. *arXiv.* doi: 10.48550/arXiv.1807.06521

Xiang, R., Ying, Y., Jiang, H., Rao, X., and Peng, Y. (2012). Recognition of overlapping tomatoes based on edge curvature analysis. *Trans. Chin. Soc. Agric. Machinery* 43 (3), 157–162. doi: 10.6041/j.issn.1000-1298.2012.03.029

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). Aggregated residual transformations for deep neural networks. *arXiv.* doi: 10.48550/arXiv.1611.05431

Xu, Z. F., Jia, R. S., Sun, H. M., Liu, Q. M., and Cui, Z. (2020). Light-YOLOv3: fast method for detecting green mangoes in complex scenes using picking robots. *Appl. Intell.* 50 (12), 4670–4687. doi: 10.1007/s10489-020-01818-w

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 (9), 1619. doi: 10.3390/rs13091619

Yan, J., Zhao, Y., Zhang, L., Su, X., Liu, H., Zhang, F., et al. (2019). Recognition of rosa roxbunghii in natural environment based on improved faster RCNN. *Trans. Chin. Soc. Agric. Eng.* 35 (18), 143–150. doi: 10.11975/j.issn.1002-6819.2019.18.018

Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., and Li, X. (2021). A real-time detection algorithm for kiwifruit defects based on YOLOv5. *Electronics* 10 (14), 1711. doi: 10.3390/electronics10141711

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

Zhang, Y., Yu, J., Chen, Y., Yang, W., Zhang, W., and He, Y. (2022). Real-time strawberry detection using deep neural networks on embedded system (rtsd-net): An edge AI application. *Comput. Electron. Agric.* 192, 106586. doi: 10.1016/J.COMPAG.2021.106586

Zheng, T., Jiang, M., Li, Y., and Feng, M. (2022). Research on tomato detection in natural environment based on RC-YOLOv4. *Comput. Electron. Agric.* 198, 107029. doi: 10.1016/J.COMPAG.2022.107029

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D. (2020). “Distance-IoU loss: Faster and better learning for bounding box regression,” in *Proceedings of the AAAI conference on artificial intelligence*. 34 (07), 12993–13000. doi: 10.1609/aaai.v34i07.6999

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 15 February 2023
DOI 10.3389/fpls.2023.1070699

Tree-level almond yield estimation from high resolution aerial imagery with convolutional neural network

Minmeng Tang[1], Dennis Lee Sadowski[2], Chen Peng[2], Stavros G. Vougioukas[2], Brandon Klever[1], Sat Darshan S. Khalsa[3], Patrick H. Brown[3] and Yufang Jin[1]*

[1]Department of Land, Air, and Water Resources, University of California, Davis, Davis, CA, United States, [2]Department of Biological and Agricultural Engineering, University of California, Davis, Davis, CA, United States, [3]Department of Plant Sciences, University of California, Davis, Davis, CA, United States

Introduction: Estimating and understanding the yield variability within an individual field is critical for precision agriculture resource management of high value tree crops. Recent advancements in sensor technologies and machine learning make it possible to monitor orchards at very high spatial resolution and estimate yield at individual tree level.

Methods: This study evaluates the potential of utilizing deep learning methods to predict tree-level almond yield with multi-spectral imagery. We focused on an almond orchard with the 'Independence' cultivar in California, where individual tree harvesting and yield monitoring was conducted for ~2,000 trees and summer aerial imagery at 30cm was acquired for four spectral bands in 2021. We developed a Convolutional Neural Network (CNN) model with a spatial attention module to take the multi-spectral reflectance imagery directly for almond fresh weight estimation at the tree level.

Results: The deep learning model was shown to predict the tree level yield very well, with a R2 of 0.96 (±0.002) and Normalized Root Mean Square Error (NRMSE) of 6.6% (±0.2%), based on 5-fold cross validation. The CNN estimation captured well the patterns of yield variation between orchard rows, along the transects, and from tree to tree, when compared to the harvest data. The reflectance at the red edge band was found to play the most important role in the CNN yield estimation.

Discussion: This study demonstrates the significant improvement of deep learning over traditional linear regression and machine learning methods for accurate and robust tree level yield estimation, highlighting the potential for data-driven site-specific resource management to ensure agriculture sustainability.

KEYWORDS

CNN, deep learning, yield prediction, multispectral imagery, almond, UAV/drone

EDITED BY
Long He,
The Pennsylvania State University (PSU), United States

REVIEWED BY
Md Sultan Mahmud,
Tennessee State University, United States
Jing Zhang,
Capital University of Economics and Business, China

*CORRESPONDENCE
Yufang Jin
✉ yujin@ucdavis.edu

SPECIALTY SECTION
This article was submitted to Technical Advances in Plant Science, a section of the journal Frontiers in Plant Science

RECEIVED 15 October 2022
ACCEPTED 12 January 2023
PUBLISHED 15 February 2023

CITATION
Tang M, Sadowski DL, Peng C, Vougioukas SG, Klever B, Khalsa SDS, Brown PH and Jin Y (2023) Tree-level almond yield estimation from high resolution aerial imagery with convolutional neural network. *Front. Plant Sci.* 14:1070699. doi: 10.3389/fpls.2023.1070699

1 Introduction

Over 2.2 million ha of land produces about 4.1 million metric tons of almonds in 2020 globally, with United States (US) as the largest producer (FAO, 2022). About 80 percent of the world's almonds are produced in California's irrigated land, generating about $5bn "farm gate value" and an additional $3 billion of indirect and induced values (CDFA, 2022). In the last two decades, the total acreage of almond orchards in California doubled and became the state's second largest agricultural commodity. The continued expansion of water and fertilizer-intensive tree crops, coupled with climate change, poses a threat to the long-term sustainability of almond industry, despite ongoing research and outreach efforts focused on tree crops (Khalsa et al., 2022). Excessive groundwater pumping especially during drought years, for example, has caused a significant drop of aquifer's water depths in Central Valley (Fulton et al., 2019). Groundwater has also been degraded due to nitrogen leaching from agricultural fields (Harter, 2009). One out of ten public water supply wells in California have nitrate levels exceeding the maximum contamination level (Harter, 2009).

In response to these challenges, various regulatory programs have been implemented in California over the past decade, requiring growers to increase the efficiency of irrigation and nitrogen use (Rudnick et al., 2021). Meeting these regulations will require more precise and adaptive irrigation and nitrogen management strategies. In particular, a change from whole-field management to zonal and even tree-specific precision agricultural practices is critical for maximizing 'crop per drop or lb of N', considering large yield variability within an individual almond orchard (Jin et al., 2020). Accurate yield estimation and prediction is a missing link in current nitrogen management tool, although the guidance is available on N fertilization given the expected almond yield for a particular orchard. An improved understanding of within-field yield variability is also needed for adaptive on-farm management to close the yield gap (Jin et al., 2020). Reliable yield estimation can also help with insurance and market decisions, which rely on the understanding of mean and variability of yields at the field scale (Lobell et al., 2015).

Both mechanistic simulation models and statistical approaches have been used for yield estimation (Hodges et al., 1987; Dzotsi et al., 2013; Burke and Lobell, 2017; Kang and Özdoğan, 2019; Sidike et al., 2019). The process models simulate crop growth, nutrient cycling, soil-plant dynamics, and energy and water balance under various climate and management scenarios (Zhang et al., 2019; Archontoulis et al., 2020), such as the Agricultural Production Systems Simulator (APSIM) model (Keating et al., 2003). Although powerful, it is challenging to calibrate these models across different sites, because of the complexity of the biological processes (Jagtap and Jones, 2002). These models often require extensive biotic and abiotic data as input, such as soil properties, which may not be available at the field or finer scale (Sakamoto et al., 2013; Zhang et al., 2019). Moreover, the majority of crop models focus on row crops such as corn, soybean, barley, and etc., while the simulation of tree crops with complicated physiological processes is very limited (Keating et al., 2003).

Statistical models, on the other hand, are based on the empirical relationships learned from the observed yield data and the factors affecting production, instead of simulating complex biophysical processes (Medar and Rajpurohit, 2014). Regression models, for example, have been developed to quantify the impact of climate on agriculture production at county and state level (Lobell et al., 2007; Lobell and Field, 2011; Mourtzinis et al., 2015; Xu et al., 2016). Studies have shown that the recent climatic trends have mixed effects on tree crop yields in California (Lobell et al., 2007; Lobell and Field, 2011). Across the US, it has been estimated that warming will lead to reduction in soybean and maize production in the Midwest (Mourtzinis et al., 2015; Xu et al., 2016). All these statistical studies provide guidance for county, state or nation-wide climate mitigation and adaptation strategies. However, the utility of these coarse scale empirical models is limited in terms of informing growers for their on-farm resource management for individual fields or trees.

Recent advancement of remote sensing technologies enables plant monitoring across a range of spatial and temporal resolutions, opening doors for data-driven yield estimation at the field scale (Shahhosseini et al., 2020; van Klompenburg et al., 2020; Rashid et al., 2021; Muruganantham et al., 2022). Both traditional and machine learning methods have been developed to relate field surveyed yield data with remote sensing metrics and other environmental drivers (Burke and Lobell, 2017; Lambert et al., 2018; Hunt et al., 2019; Zhang et al., 2019). Burke and Lobell (2017) found that the linear regression model, driven by vegetation indexes (VIs) derived from high resolution multi-spectral images from Terra Bella satellite at 1m, predicted well the yield for maize fields in west Kenya. Machine learning models such as random forest and gradient boosting trees have also been developed to predict yield for individual fields over almond tree crops by integrating Landsat VIs and weather data in California (Zhang et al., 2019), over wheat in United Kingdom using Sentinel-2 VIs (Hunt et al., 2019), and over cotton, maize, millet and sorghum in Mali using Sentinel-2 VIs (Lambert et al., 2018).

Most recently more complex deep learning models such as Deep Neural Network, Convolutional Neural Network (CNN), and Recurrent Neural Network have been introduced to improve yield estimation with large remote sensing datasets, due to their improved performance over traditional statistical approaches (Ball et al., 2017; You et al., 2017; Cai et al., 2018; Kang and Özdoğan, 2019; Khaki and Wang, 2019; Sidike et al., 2019; Kang et al., 2020; Khaki et al., 2020; Ma et al., 2021). The Bayesian neural network model, for example, has been shown to predict county-level corn yield well in twelve Midwestern states of US ($R^2 = 0.77$), using VI time series from MODIS imagery, climate variables, soil properties, and historical average yield (Ma et al., 2021). A limited studies applied recurrent neural network framework such as Long Short Term Memory models to take into account of sequential imagery and weather for county-level corn yield in combination with CNN; their models outperform the traditional regression and machine learning models (You et al., 2017; Khaki et al., 2020). Shahhosseini et al. (2021) also explored a hybrid approach to integrate features from crop modeling to machine learning models and found the importance of hydrological inputs for yield estimation in the US corn belt. At field scales, data assimilation technique has been explored to incorporate the remote sensing observations of canopy development into the Decision Support System for Agrotechnology Transfer (DSSAT) crop model for corn yield mapping over the US corn belt (Kang and Özdoğan, 2019). However, most of the studies still use human-engineered index-based feature extraction method, such as some widely used vegetation index and contextual information derived from imagery, to predict yield and do not explore the potential of learning-based feature extraction with deep learning models that directly use multi-spectral imagery as input.

In order to capture variations of crop yield among individual plants for precision management, higher spatial resolution observations of canopy structure and conditions are required, such as those from very high-resolution commercial satellite and aerial imagery (Sidike et al., 2019; Maimaitijiang et al., 2020). Recent advances in computer vision and deep learning technology further unlock the power of centimeter imagery for fine scale yield estimation at individual plant or sub-field level. Chen et al. (2019) developed a region-based CNN model to detect and count the number of flowers and strawberries at plant level from the RGB drone imagery and found an overall counting accuracy of 84.1%. Another study integrated multi-spectral and thermal drone imagery with machine learning and deep neural network models to estimate the sub-field soybean yield in US (Maimaitijiang et al., 2020). However, the study on plant-level yield variation is still very limited and the majority focuses on row crops, mostly due to the lack of field-based yield database for individual plants, especially for tree crops.

We here took advantage of a unique individual tree harvesting data and aerial imagery of multiple spectral bands at 30cm spatial resolution over an almond orchard in California's central valley, to explore the potential of deep learning for tree level almond yield estimation. Specifically, we aimed to address the following questions: (i) how CNN model can be used to estimate almond yield for each individual tree, based on very high resolution multi-spectral imagery; and (ii) what is the capability of the trained CNN models in capturing the within-field almond yield variation; and (iii) what is the relative importance or added value of the observations in the red edge part of the spectrum, a spectral band increasingly available in recent imaging systems, with regard to almond yield estimation.

2 Materials

2.1 Study orchard and Individual tree harvest data

This study was conducted over an almond orchard with a size of 2 squared kilometers in Vacaville, California, USA (Figure 1). Under a typical Mediterranean climate, the area experiences hot dry summer with average daily max temperature in July of 34 °C and cool winter with average daily minimum temperature in January of 3.7 °C. Mean annual precipitation is 63 (± 21) cm and the majority rainfall occurs from November to March (BestPlaces, 2022; Cedar Lake Ventures, 2022; WRCC, 2022). For almond tree, the water usage increases gradually from March to July, and decreases from July to October (Athwal, 2021). The hot and dry summer requires large amount of irrigation water usage to support crop growing, which mainly comes from groundwater and surface water including Lake Berryessa and Putah Creek (SID, 2012; BoR, 2022).

The orchard was planted with a self-fertile productive almond cultivar, 'Independence', between 2015 and 2017. Within the orchard, rows are oriented northeast to southwest in parallel with prevailing winds, and the average row spacing is about 6 m and the average spacing between trees along the same row is about 4.5 m. Almond trees bloom between late February and early March, followed by leaf out, fruit set and rapid growth, reaches full canopy typically in June or early-July, and fruit maturity progresses through summer. Almonds are typically harvested from August to October, and trees become dormancy during the winter season.

We designed an automatic weighing system attached to the commercial almond harvester to measure the almond yield of an individual tree (Figure S1). The yield (including wet hulls and shells) measurements were made for each individual tree every seven rows in the north-west portion of the orchard between August 23 and August 27 in 2021 (Figure 1). A total number of 1,893 trees were individually harvested, with an average fresh weight yield of 53.1 ± 17.6 kg per tree. The location of each sampled trees was also recorded. Large yield variation was found among individual trees with a coefficient of variation of 33.1% and interquartile range of 24.3 kg per tree.

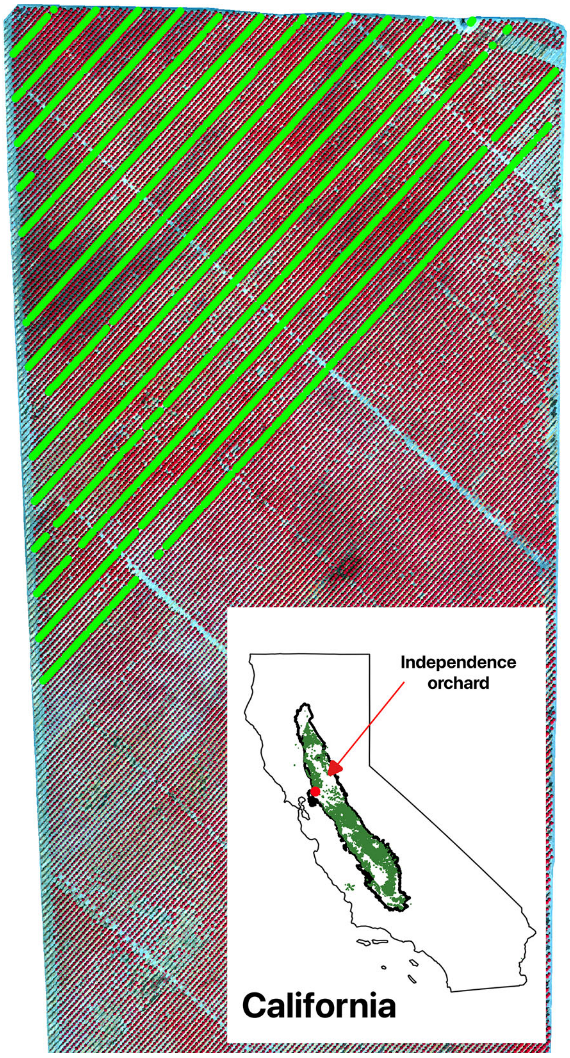


FIGURE 1
Study orchard as shown by the color infrared composite of CERES aerial imagery acquired on July 29, 2021. Individual trees with yield measurements were shown as green dots. The inset shows the location of the study orchard among all almond orchard fields (green) in California's Central Valley (black polygon).

2.2 Aerial imagery acquisition and processing

Multi-spectral aerial imagery was acquired on July 29, 2021, about one month ahead of the harvest, by CERES Imaging (Oakland, USA.) A multi-spectral imaging camera was integrated with a crop duster plane flying at 6,000 ft above the ground, resulting in images with a 0.3-meter spatial resolution. Four spectral bands are centered around 800 nm (near infrared), 717 nm (red edge), 671 nm (red), and 550 nm (green), with a spectral resolution of 10 nm (the full width at half maximum). The image was acquired near local solar noon to minimize the shadow effects.

2.3 Tree identification and location extraction from imagery

For each individual tree, extracting its center location from CERES imagery is needed in order to match the tree yield record from the harvester and to clip the corresponding image block as CNN input. We developed a multi-stage segmentation method to identify all individual crowns with varying canopy sizes, especially over a mature orchard. First, Normalized Difference Vegetation Index (NDVI) was calculated for each pixel from the red and near infrared bands of the CERES aerial imagery (Figure 2A). Second, NDVI imagery was segmented based on the NDVI threshold to identify potential tree crowns automatically (Figure 2B). Lower NDVI threshold tended to be more inclusive in identifying canopy pixels and resulted in a tree crown boundary with multiple inter-connected trees in it; whereas higher NDVI threshold separated individual tree crowns better but may miss smaller trees (Figure 2B). We therefore applied seven NDVI thresholds ranging from 0.60 to 0.83 (Table S1), producing seven layers of potential tree crown polygon maps. Third, for each layer, those polygons that actually had multiple trees were removed, based on the comparison of the polygon major axis length and the orchard tree spacing (Figure 2C). The assumption is that one single tree crown diameter can't exceed the spacing between adjacent trees. Finally, by taking advantage of higher threshold's capability of separating individual trees and lower NDVI threshold's capability of identifying small trees, we combined those seven potential single tree crown polygons iteratively, based on their spatial relationships, into one final tree crown boundary optimal for tree center extraction. The goal was to remove the redundancy among those layers yet maintain the largest crown size. Starting from the crown polygons (smallest size), typically associated with higher NDVI threshold value, if it was spatially within the crown polygon (larger) identified by the lower threshold value, it was deleted; otherwise, it was added to the final single tree crown polygons map. By iterating this step, we created a final version of single tree crown polygons map (Figure 2D). Finally, the tree locations were extracted from the centroid coordinates of all the segmented tree crown polygons.

For quality control, the extracted tree locations were plotted over the CERES imagery for visual examination. For example, those trees with very small or large crowns were carefully examined against CERES imagery to ensure the location accuracy. To further ensure the

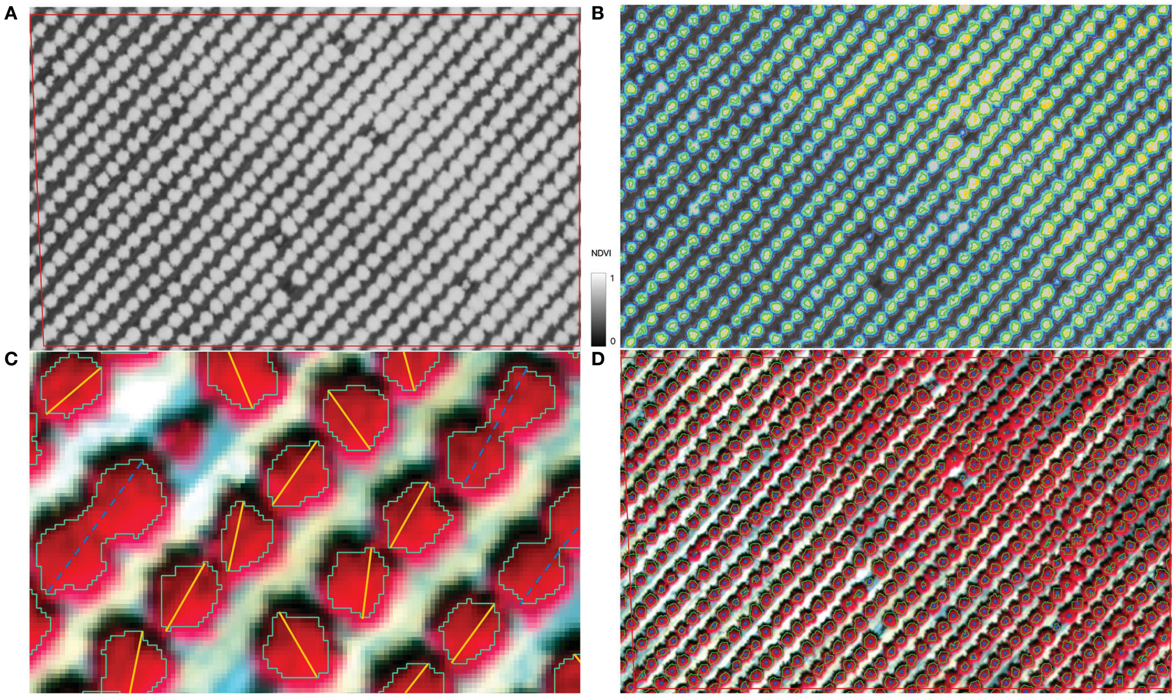


FIGURE 2
Illustration of individual tree identification workflow: **(A)**. NDVI map from CERES imagery; **(B)**. Segmented tree crowns with various NDVI threshold values, e.g., the blue polygon represents the boundaries from the segmentation with a NDVI threshold of 0.6; **(C)**. For each polygon layer identified using a particular NDVI threshold, remove those crown polygons whose major axis (dashed blue line) were longer than the expected maximum tree crown diameter, roughly the tree planting spacing along the orchard row; **(D)**. Final tree crowns by combining all layers of potential crown polygons and center locations of all individual trees.

alignment with the locations of the individually harvested trees, a visual check of the locations of starting, ending, and some randomly selected trees within the harvested rows was also conducted. All these processes were done in Python and QGIS.

3 Methods

3.1 Convolutional neural network architecture

The Convolutional neural network (CNN), a most established deep learning algorithm, is developed to estimate fresh almond yield with multi-spectral aerial images as inputs. CNN has a unique ability to automatically and adaptively learn spatial hierarchies of important features that summarize the presence of detected features in the input image for a particular predictive modeling problem (LeCun et al., 2015). The extreme efficiency in dimensionality reduction of the CNN model makes it unnecessary to conduct any feature extraction work, which increases computation efficiency and improves estimation accuracy. A surge of interest in CNN deep learning has emerged in recent years due to its superior performance in various fields (Lobell et al., 2015; Yamashita et al., 2018; Kattenborn et al., 2021; Li et al., 2021).

A CNN is typically composed of a stacking of three types of layers, i.e., convolution, pooling, and fully connected layers (LeCun et al., 2015). The first two perform feature extraction, whereas the third maps the extracted features into final output, such as yield. As a fundamental component of the CNN architecture, a convolutional layer typically consists of a combination of linear and nonlinear operations, i.e., convolution operation and activation function. A convolution is a simple application of a spatial filter (or kernel) to an input image that results in an activation. Repeated application of the same filter to an input result in a map of activations called a feature map. A small grid of parameters called kernel, an optimizable feature extractor, is applied at each image position, which makes CNNs highly efficient for image processing. The kernel values are optimized during the model training process to extract features from input data based on the model's task. The outputs of a linear operation such as convolution are then passed through a nonlinear activation function, e.g., the most commonly used rectified linear unit (ReLU). Batch normalization can also be applied as an optimization strategy to increase the model training efficiency, although it is not a solid requirement of the CNN model. To reduce the dimensionality of the extracted feature maps, a pooling layer provides a down-sampling operation by aggregating the adjacent values with a selected aggregation function, such as taking maximum value within the predefined window size. Similar to convolution operations, hyperparameters including filter size, stride, and padding are set in pooling operations. As one layer feeds its output into the next layer, extracted features can hierarchically and progressively become more complex.

To improve CNN model's overall performance, the spatial attention module is recently introduced into the CNN architecture by combining a global average pooling layer and the following dense layers (Woo et al., 2018; Sun et al., 2022; Zhang et al., 2022). Global average pooling layer is usually applied once to downscale the feature maps into 1-D array by averaging all the elements in each feature map, while retaining the depth of the feature maps. Dense layer then connects the final feature maps to the final output of the model with learnable weights *via* model training. The combination of a global average pooling layer and the following dense layers helps the CNN model focus more on the relevant features and thus improves.

3.2 CNN configuration and optimization

TensorFlow (Abadi et al., 2016), Keras (Chollet, 2015), and KerasTuner (O'Malley et al., 2019) libraries in Python were used for CNN model tuning and training processes. The CNN model took the image blocks, centered around each individual almond tree crown, from CERES images at 0.3 m resolution, for 4 reflectance bands (R, G, NIR, and RE) as inputs to estimate the individual tree almond yield (Figure 3). We started with the minimum block size of 21 × 21 pixels, equivalent to a 3m radius centered around each tree crown center and thus representing areas slightly bigger than one tree crown size. For each tree sample, we first identified the corresponding CERES pixel containing the tree center (as described in Section 2.3 location), and then clipped an image block extending 10 pixels towards all four directions from the center, for each band. This step resulted in 21 × 21 × 4 multi-spectral imagery associated with each individual tree crown as the input to the CNN model.

The CNN model training process is to find kernels in the convolutional layers and weights in the dense layers to minimize the differences between model estimations and ground measurements on a training dataset. The Mean Squared Error (MSE) loss function was applied for the CNN model training, which calculates the average of the squared differences between model estimations and actual values. To efficiently optimize the kernels and weights within the CNN model, the Adam optimization algorithm (Kingma and Ba, 2014) is used, which extends the stochastic gradient descent algorithm by calculating individual learning rates for different parameters based on the estimates of first and second moments of gradients. 5-fold cross validation (CV) is applied to randomly split the data into separate training and testing sets. The overall model performance is evaluated based on the average performance over the testing set in each fold. The Bayesian optimization algorithm is developed to select the CNN hyper-parameters automatically.

The general setup of the possible CNN structures for the Bayesian optimization algorithm are as follows: three to four convolutional blocks followed by a spatial attention module with a global average pooling layer and two fully connected dense layers. For the first dense layer, there are 30 to 100 neurons followed by a dropout layer. For each convolutional block, there are 16 to 128 convolutional layers (kernels) followed by a batch normalization and pooling layers, then another 16 to 128 convolutional layers followed by a batch normalization, pooling and ReLU activation layers. The pooling layers in each convolutional block can be either average pooling or max pooling. The overall architecture of the CNN model for the Bayesian optimization algorithm is shown in Figure S3. For model compiler, the Bayesian optimization algorithm selects learning rate varying from 10^{-4} to 10^{-2} with Adam optimizer. For the Bayesian optimization algorithm itself, the maximum trail number was set to 50, and for each trail, the batch size is 128 with 100 epochs.

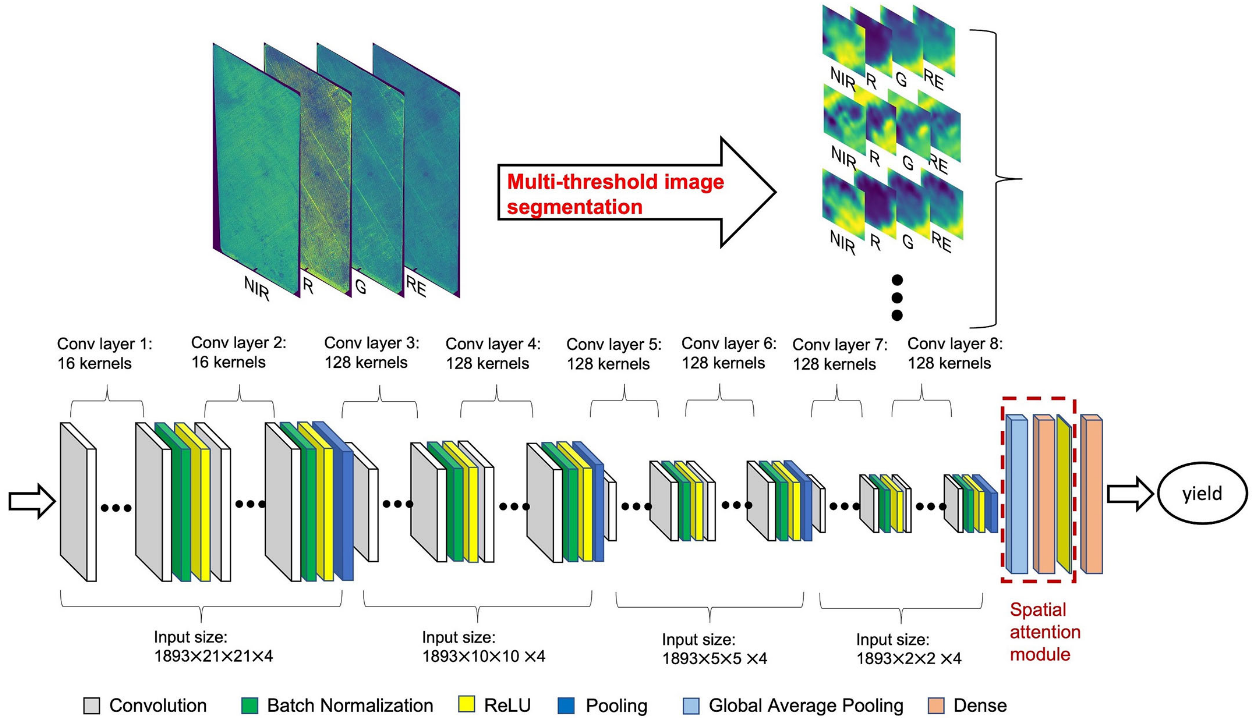


FIGURE 3
CNN model structure for tree-level yield estimation with multi-spectral aerial imagery. Input size represents total number of tree samples × image block height × image block width × number of bands. For each tree sample, an image block was clipped for each one of the four band imagery, with 21 by 21 pixels (at a 30cm resolution) centered at the identified tree center location.

To investigate the impact of input image block size used for the CNN model and explore how the neighboring trees potentially influence yield estimation, another two separate CNN models were built with an input image size of 41 × 41 pixels (roughly 6m radius) and 61 × 61 pixels (9m radius), respectively. To understand the contribution of the red edge band to the yield estimation, a reduced CNN model was constructed by excluding red edge reflectance as input, hereafter called "reduced CNN model", considering that red edge band is not as widely used for aerial imaging as the other three bands. Similarly, another 14 sets of reduced CNN models were further built with all the combinations of different reflectance bands as input and compared how they influenced model's yield estimation accuracy (Table S2).

3.3 Traditional machine learning model estimations

For comparison purposes, Other statistical models were also built for individual tree level almond yield estimation, including stepwise linear regression as a baseline for linear relationships and four traditional machine learning approaches. The Scikit-learn (Buitinck et al., 2013) and hyperopt (Bergstra et al., 2013) libraries were used for building support vector regressor (SVR) (Platt, 1999), random forest (RF) (Breiman, 2001), and extreme gradient boosting (XGB) models (Chen and Guestrin, 2016). Additionally, a DNN model was also developed using the same libraries as CNN model. The traditional machine learning models use the human-engineered index-based feature extraction method to predict almond yield, which differs from the CNN model that directly takes imagery as input. By comparing traditional machine learning models against CNN model, it helps to evaluate the advantages of applying learning-based feature extraction in yield prediction.

Regression models were built using features at individual tree level as inputs, including VIs and texture. 13 commonly used vegetation indices (VIs) were calculated from CERES multi-spectral imagery, including those sensitive to structure, greenness, and chlorophyll content (as described and summarized in Table S3 in the supplementary material). A circular buffer with a 2.5-meter radius was used to calculate the zonal statistics of remote sensing metrics, since most tree crowns have diameters less than 5 meters. Tree crown pixels were identified with NDVI greater than 0.5, and the fractional coverage of tree crown within the buffer area was then calculated to represent the size of crown. The average of VI values over the identified crown pixels within the buffer area were also derived to represent the overall biomass of an individual tree. In total, 14 variables were calculated including 13 VIs and one fractional coverage variable.

To extract textural features for each of the four band images, the gray level co-occurrence matrix (GLCM) (Haralick et al., 1973) was applied. The GLCMs were constructed with a moving distance of one pixel and four moving directions. Eight texture measures were calculated from reflectance imagery with a 2x2 moving window, including contrast, dissimilarity, homogeneity, angular second moment, correlation, mean, variance, and entropy (Nichol and Sarker, 2011; Wood et al., 2012). For each individual tree, the corresponding texture features were extracted and averaged from textural images, resulting in a total of 32 texture features.

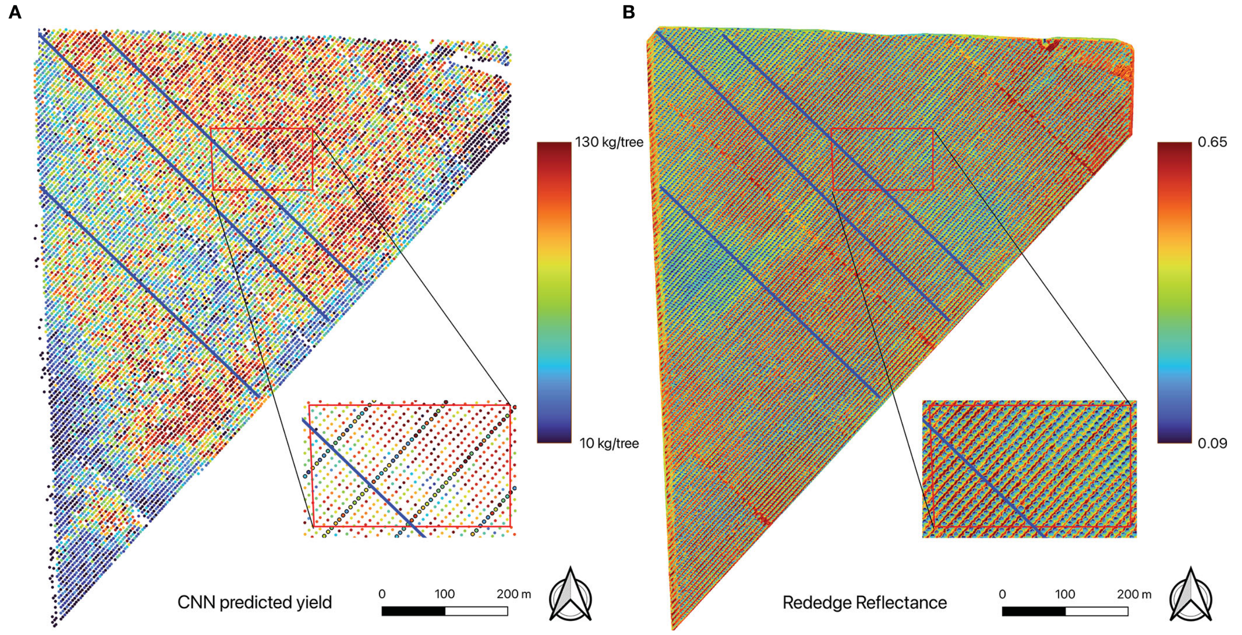


FIGURE 4
Maps of **(A)** individual tree yield estimated by the CNN mode, showing within-field yield variation, and **(B)** red edge reflectance. A close-up of the yield map shows detailed spatial distribution of the estimated tree yield (those trees with ground yield measurement were indicated by black circles). Also shown are three transects (blue lines) for detailed tree-to-tree yield variation analysis.

3.4 Accuracy assessment and yield variability analysis

To evaluate models' performance in predicting almond yield, the predicted and observed individual tree yield from the reserved testing samples were compared, and the coefficient of determination (R^2), Root Mean Squared Error (RMSE), and RMSE normalized by averaged yield measurement (NRMSE) were calculated. Statistics of these metrics were reported based on 5-fold cross validation.

For the model with highest accuracy, its capability to capture the within-field yield variations, such as overall spatial patterns, row to row variations, and tree to tree variations along selected transects was also evaluated. For all harvested rows, the yield distribution for all trees within each individual row was analyzed based on CNN estimations. Furthermore, three transects that are perpendicular to the row orientation of the orchard were randomly selected to examine the inter-row yield variations. The locations of the selected transects are shown in Figure 4 highlighted in blue lines.

4 Results

4.1 Optimized CNN model and performance

After 50 iterations of Bayesian optimization process during model training, the final optimized CNN model had eight convolutional layers, each of which was followed by a batch normalization and an ReLU activation function. Four max pooling layers were deployed after every two convolutional layers to extract spatial features and reduce image dimension. A global average pooling layer further flatten the image into one-dimension array. A 100-neuron dense layer is introduced. The final one neural dense layer further reduces the input data into a single output value, which directly connects to the tree level yield data (Figure 3).

The trained CNN full model, with four spectral band imagery as inputs, performed very well in predicting almond yield at the individual tree level. The 5-fold cross validation with the testing data showed that it captured 96% (± 0.2%) of tree-to-tree variation in almond yield, with a RMSE of 3.5 kg/tree (± 0.11) and a normalized RMSE of 6.60% (± 0.2%) (Figure 5). The scatter plot of predicted vs. observed tree yield also showed a good agreement (Figure 6). The predicted yield by the full CNN model for all individually harvested trees followed very similar distribution as shown by the measurements (Figure 5), with a mean yield of 52.9 ± 17.2 vs. 53.1 ± 17.6 kg/tree and the interquartile ranges of 23.8 vs. 24.3 kg/tree. No statistically significant difference was found between predicted and observed tree yield based on the two-tailed t-test (p-value of 0.75).

The performance of the full CNN models with all four bands varied, very slightly, with the size of input image blocks (Table 1). For example. when using image blocks covering nine tree crowns, the re-trained CNN model captured 97% of yield variability and had slightly larger uncertainty with a NRMSE 5.2%. However, the estimation bias is larger for CNN models with image blocks covering more tree crowns. Hereafter only the results from the CNN model with 21 × 21 pixels image block size was reported.

4.2 Impact of spectral information

When removing the red edge imagery from the input imagery, the accuracy of the reduced CNN model was reduced significantly, with a lower R^2 of 0.68 (± 0.08) and higher NRMSE of 18.7% (± 2.3%) than the full CNN model with four band imagery as input (Figure 7). Among the reduced models with all possible combinations of three

TABLE 1 Performance of CNN models with different image block sizes of the input aerial image clipped around each individual tree crown center.

Image block size	Test R^2	RMSE (kg/tree)	NRMSE	IQR (kg/tree)	Bias (kg/tree)
21×21 pixels	0.96 (± 0.002)	3.50 (± 0.11)	6.6% (± 0.2%)	23.82	-0.181
41×41 pixels	0.95 (± 0.017)	4.02 (± 0.53)	7.6% (± 1.0%)	23.55	1.46
61×61 pixels	0.97 (± 0.005)	2.77 (± 0.34)	5.2% (± 0.6%)	22.69	-2.35

All four spectral bands were used as input.

bands, the CNN model driven by red edge, NIR, and red reflectance performed the best, with a R^2 of 0.85 (± 0.01) and NRMSE of 12.6% (± 0.7%). For two band combinations, the reduced model with NIR and red edge bands or NIR and green bands had similar performance (R^2 0.85 (± 0.02) and 12.6% (± 0.8%)). When driven by only one single band imagery, the red edge based CNN model still captured 83% (± 2%) of yield variability among individual trees, and NRMSE only increased slightly to 13.8% (± 1.0%). These results demonstrated the importance of red edge imagery in almond yield estimation.

4.3 Comparison with machine learning models

Our comparison showed that CNN model significantly outperformed the linear regression model and the other machine learning models, based on the 5-fold CV, regardless of combinations of input features such as VIs, texture, and raw multi-spectral reflectance (Figure 7). XGB and RF models captured only up to 54% (± 3.8%) of yield variability, similar to linear regression models. In addition to achieving the highest R^2, the CNN model was found more robust and stable as shown by much lower standard deviation of R^2 among different folds of test sets, compared with other models (Figure 7). The scatter plots of predicted vs. measured yield further showed better performance of the CNN model (Figure 6).

4.4 Predicted yield map and spatial patterns

The CNN full model, once trained and validated, allowed us to estimate yield for every individual almond tree in the orchard. The

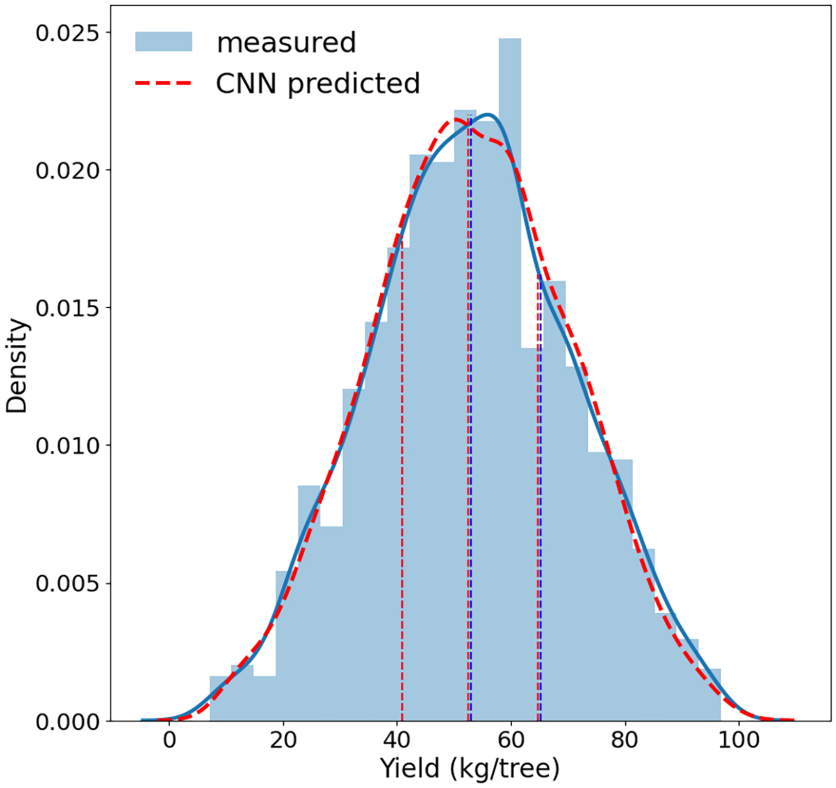


FIGURE 5
Distributions of almond tree yield predicted by the full CNN model (red) vs. measured by individual tree harvester (blue). Dashed vertical lines represents the 25th percentile, median, and 75th percentile respectively.

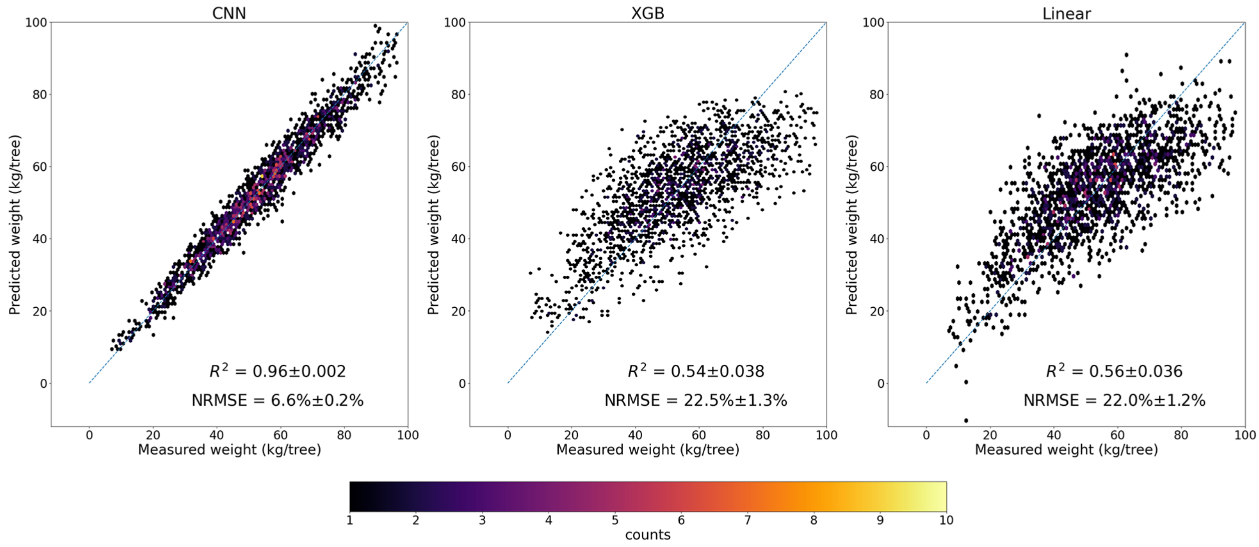


FIGURE 6
Scatter plots of predicted yield by the full CNN model, XGB, and Linear models vs. measured yield.

yield map showed within-field variations of almond yield from tree to tree (Figure 4A). Trees with higher yield were mostly located in the northeast corner of the orchard, while least productive trees were mainly distributed around the orchard boundary. The overall spatial pattern was consistent with the pattern captured by the red edge reflectance (Figure 4B).

When row to row yield variation was examined, the CNN model predicted yield followed similar distribution with the ground measured yields for every seven rows with individual tree yield measurement (Figure 8). Row 14 had the highest yield as shown by both estimation (66.9 ± 15.0 kg per tree) and measurements (68.4 ± 13.3 kg per tree); in contrast, the production of Row 84 was 25% lower (50.3 ± 16.1 kg per tree) and 30% lower (47.6 ± 15.4 kg per tree) for both estimation and ground measurements, respectively. The estimation showed large within-row yield variability, with coefficient of variation (CV) ranging from 20.0% to 44.9% and inter-quantile range (IQR) ranging from 16.4 to 31.1 kg per tree, similar to the variability observed by the measurements (Figure 8). For rows without ground measurements, the predicted yield also captured similar general trend of row-to-row variation as that from the measurements over the sampled rows.

Furthermore, along the transect lines across rows, the inter-row variability from the CNN predicted tree level yield agreed relatively well with that from the ground measurements (Figure 9). Among the measured rows, for example, the most productive trees were found in Rows 77 (104.1 kg/tree), 7 (85.4 kg/tree), and 77 (84.4 kg/tree), for each transect, respectively, based on the predicted yield map. In contrast, the least productive trees had much lower yield, i.e., 38.7 kg/tree in Row 91 for transect 1, and 35.9 kg/tree in Row 84 for transect 3. These findings were similar to the observations from the harvesting data. The yield distributions along each row and the inter-row yield variations demonstrated the consistent performance of CNN model over space with less spatial dependency and variations.

5 Discussions

5.1 Yield estimation model performance

As a first study on tree level almond yield estimation, our findings showed the high accuracy of the CNN model in capturing the spatial yield variability from tree to tree, when driven by multi-spectral reflectance from high resolution aerial imagery. The comparative analysis in this study showed that the CNN model outperforms the traditional machine learning models. First of all, the CNN model framework is able to automatically learn the complex associations from the multi-spectrum tree crown imagery to fully capture the complexity of tree physiology. The spatial pattern of multi-spectral reflectance over the whole crown plays an important role in yield estimation, which cannot be acquired by the average values. For the traditional ML models, the models' performance generally agrees with literatures using similar features as input for soybean and corn yield estimations. One study focusing on soybean yield estimation with multi-spectrum UAV images shows that models with VIs and thermal information have R2 varying from 0.520 to 0.625 (Maimaitijiang et al., 2020). Based on linear, RF, and XGB results, adding texture features improve model's ability to explain almond yield variation by 1%, 3%, and 3%, respectively. Some literatures focusing on row crops also have similar finding, but the texture features play a more important role than tree-based plants (Maimaitijiang et al., 2020; Wang et al., 2021). In the soybean study, the VIs, thermal, and structure information explain 52% to 63% of the yield variation with different methods, but adding texture features improves the estimation to explain 65% to 72% of the yield variation, which means that adding the texture features improves about 20% of the estimation accuracy (Maimaitijiang et al., 2020); another rice yield estimation study shows that growing stage VIs explain 56.6% of yield variation and adding extra texture features helps to explain 65.5%

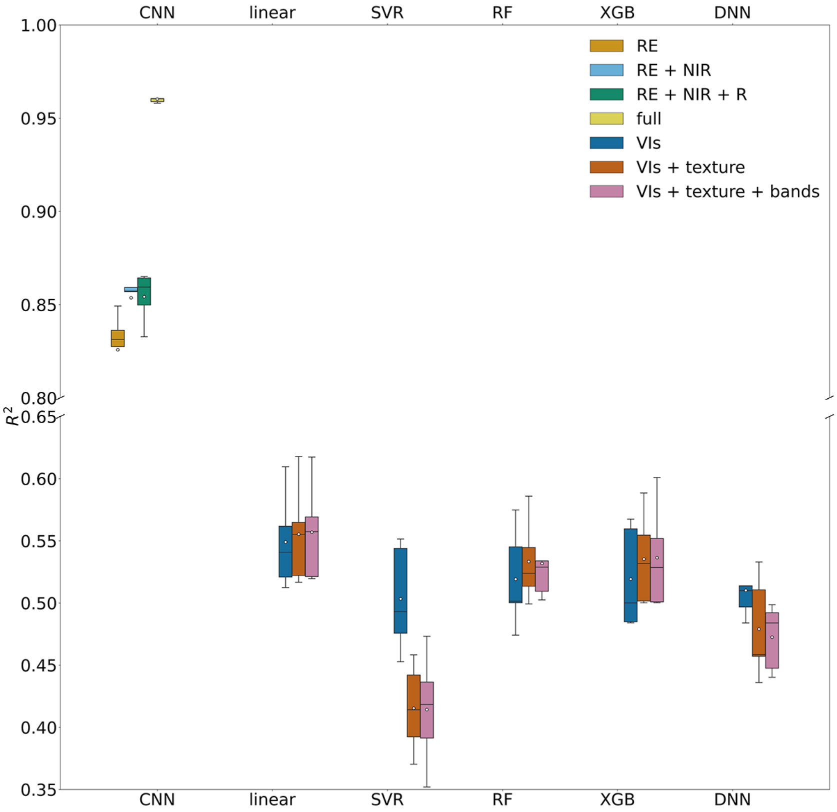


FIGURE 7
Model performance in predicting tree level yield, quantified by R^2 with the test data set, for CNN models with different spectral bands and machine learning models with different combinations of input features.

yield variation, which increases estimation accuracy by 16% (Wang et al., 2021).

Second, the human-engineered features commonly used by traditional statistical approaches may not fully capture the characteristics influencing yield variation. Most of previous studies focused on crop yield estimation with human-engineered features including VIs and textures, with both ML and AI models showing R^2s between 0.7 to 0.9 for mostly row crops including wheat, soybean, corn and so on (Kuwata and Shibasaki, 2016; Hunt et al., 2019; Jin et al., 2019; Maimaitijiang et al., 2020; Ma et al., 2021; Wang et al., 2021) and almond orchards at the block level (Zhang et al., 2019). Although these studies use various indices from multi-spectral and thermal UAV images to satellite-based radar backscatter, the estimation accuracy are in general lower than our CNN model with multi-band reflectance as direct inputs. This suggests that human-engineered features may not be comprehensive to fully capture the canopy structures and conditions and yield variations. For example, some information may be lost by only using the well-known remote sensing indices.

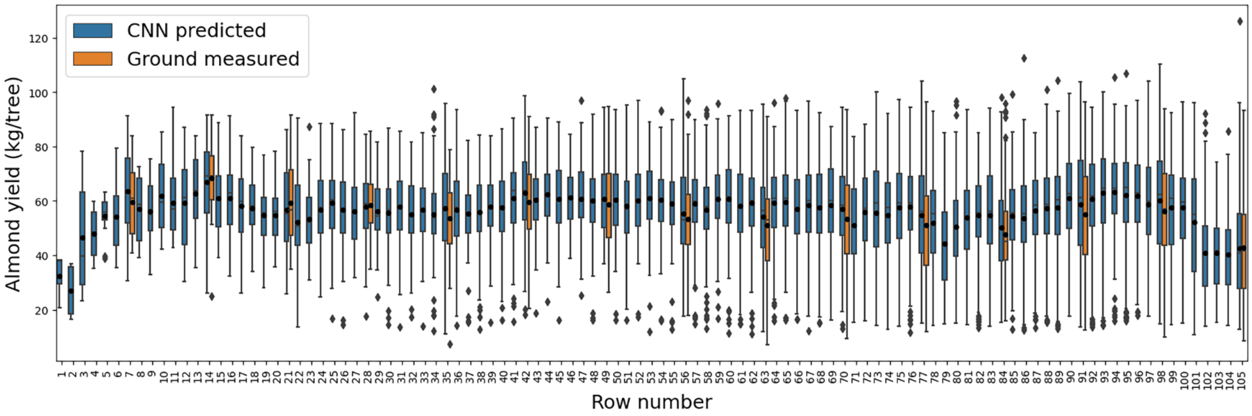


FIGURE 8
Yield variation within each row as represented by the boxplots of the tree-level yield estimated by the CNN model (blue), and across individual rows. The boxplots of measured yield record for those rows with individual tree harvesting are also shown here in orange for comparison.

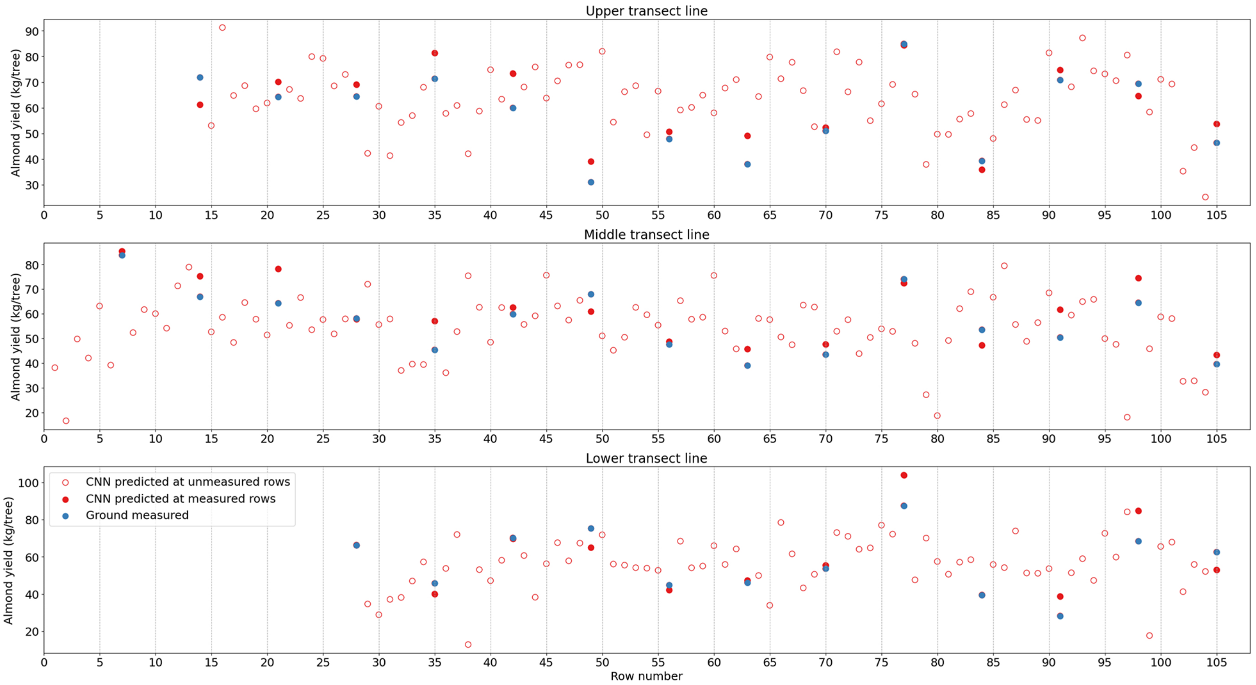


FIGURE 9
Almond yield variation from tree to tree along three selected transects as shown in Figure 4. CNN-estimated yield is represented by red while harvest data in blue; red open circles are for CNN estimation at rows without individual tree yield measurements.

Third, super high spatial resolution imagery may improve yield estimation accuracy with more details, especially for deep learning approaches. Gavahi et al. (2021) developed a DeepYield model, which combines convolutional long-short term memory for soybean yield estimation using MODIS Terra and Aqua surface reflectance, land cover type, and surface temperature products. Their results show that the DeepYield model outperforms CNN model with R^2s of 0.864 over 0.80, which are generally better than many indices-based yield estimation studies. But their yield estimation accuracy is still lower than our CNN model, which is possibly due to their low spatial resolution of input image (500 m and 1 km of MODIS Terra and Aqua products).

5.2 Importance of red edge band

From the CNN model result, reflectance in the red edge band was found to play a vital role in almond yield estimation. The red edge spectral band covers a transitional wavelength region from the red band, where the absorption by chlorophyll is dominant, to near infrared where strong scattering by leaf cell structure is further enhanced by multiple scattering among layers of leaves. Reflectance in the red edge band serves as a critical proxy for canopy size and leaf volume. Previous study shows that the red edge band is less saturated at high biomass condition than its adjacent wavelengths and the common vegetation indices such as NDVI (Todd et al., 1998; Mutanga and Skidmore, 2004; Aklilu Tesfaye and Gessesse Awoke, 2021). Moreover, the change in the red edge reflectance may capture some stress conditions of plants, as shown by a recent study on grapevine water stress detection with drone imagery (Tang et al., 2022). Our finding also indicates the potential utility of red edge imagery from Sentinel 2A and 2B satellites for scaling up yield estimation at a large scale.

5.3 Uncertainties and future work

This is the first study attempted for the tree-level yield estimation, especially capturing the spatial variability of almond yield within an individual orchard. Although it proves the concept of integrating aerial and drone-based images with deep learning techniques for high resolution yield estimation, some uncertainties still exist. Potential errors, for example, may exist in the harvest yield records used for the model training and testing, as this was the first time the individual tree harvester was designed and tested in the almond field. The sampling strategy, designed by the other group for individual tree harvesting, i.e., every seventh row, prevented us from taking full advantage of the spatial information from neighboring trees for yield estimation in the model building process.

The success of integrating the CNN model with multi-spectral imagery in estimating the within field variability is likely because the imagery at various wavelengths captures the information on the tree structure and plant conditions due to the light-matter interaction. The structural variability such as canopy size can result from cumulative impacts on plant growth by soil properties and long-term climate, while weather variability can also affect the plant health during a particular season. Nonetheless, our study was still constrained by the availability of the yield records for individual trees in one orchard over one single year. Although the unique yield dataset provided sample data covering the gradient of spatial yield variation within a single orchard, it does not represent the yield variability across different orchards where climate and soils may vary significantly. Similarly, the lack of yield record at the tree level from multiple years has prevented us to incorporate weather information in our modeling approach. Future work is needed to collect more ground truthing data and include additional predictors such as soil properties and weather variables for more robust yield estimation and prediction (Zhang et al., 2019).

With rapid advancement in deep learning technology, an important next step is to explore the potential and utility of other powerful approaches such as transformer networks (Vaswani et al., 2017; Liu et al., 2022) and generative adversarial network (Goodfellow Ian et al., 2014). This is particularly helpful for developing a scalable yield estimation workflow, when integrating the time series of high-resolution satellite-based or aerial-based imagery, sometimes at different spatial scales and from different sensors. Remote sensing imagery during the whole growing season and possibly from previous year, for example, can be utilized to integrate the phenological information, e.g., bloom development (Chen et al., 2019), to further improve yield estimation accuracy.

6 Conclusion

Individual tree level yield estimation is critical for precision on-farm management and for improving our understanding of yield variability within a field. The challenge of matching efficient supply of inputs like water and fertilizer with tree scale demand is hampered by a lack of understanding of yield variation within orchard blocks. Our work makes a significant step toward bringing awareness to the problem by coupling high-resolution imagery and modeling and paves the way for future innovation in precision orchard management. A CNN deep learning models in estimating almond yield was developed and evaluated, by taking advantage of a unique tree yield data and super high resolution of multi-spectral aerial imagery in 2021 over a single cultivar almond orchard in California's Central Valley. The 5-fold cross validation showed that the CNN model with spatial attention module, driven by 4-band block imagery of 21 by 21 pixels, captured 96% (±0.2%) of tree-to-tree variation within the study almond orchard with a very low RMSE 3.50 kg/tree and NRMSE of 6.6% (± 0.2%). The reduced CNN model with the red edge band reflectance alone had a R^2 of 0.83 (± 0.02) and NRMSE of 13.8% (± 1.0%). The CNN model performed significantly better than traditional machine learning methods and stepwise linear regression driven by tree-level features such as VIs and texture.

The almond yield for all individual trees predicted by the CNN model also captured well the spatial patterns and variability of almond yield from row-to-row and from tree-to-tree both within a row and along a transect perpendicular to the row orientation. Our findings demonstrated the potential of applying deep learning technology to integrate high resolution multi-spectral aerial images for accurate and robust tree level yield estimation. The data-driven approach developed here fills an important gap in tree level yield estimation critical for site-specific orchard resource management, ultimately contributing to agriculture sustainability.

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

MT: conceptualization, methodology, analysis, writing, review and editing. DS, CP, SV, BK, SK, PB: data collection, project coordination, review and editing. YJ: conceptualization, methodology, writing, review and editing. All authors contributed to the article and approved the submitted version.

Funding

This study is supported by funding from the Almond Board of California (HORT-66) and the US-Israel Binational Agricultural Research and Development Fund (BARD IS-5430-21).

Acknowledgments

Jin is also supported by th NIFA project CA-D-LAW-2620-H. The authors appreciated the help by the CERRES Imaging with the aerial imagery acquisition and some image preprocessing for this study. The authors were also thankful for the owner and staff of the Independence orchard.

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.2023.1070699/full#supplementary-material

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "{TensorFlow}: A system for {Large-scale} machine learning," in *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (Savannah, GA, USA,). 265–283.

Aklilu Tesfaye, A., and Gessesse Awoke, B. (2021). Evaluation of the saturation property of vegetation indices derived from sentinel-2 in mixed crop-forest ecosystem. *Spatial Inf. Res.* 29 (1), 109–121. doi: 10.1007/s41324-020-00339-5

Archontoulis, S. V., Castellano, M. J., Licht, M. A., Nichols, V., Baum, M., Huber, I., et al. (2020). Predicting crop yields and soil-plant nitrogen dynamics in the US corn belt. *Crop Sci.* 60 (2), 721–738. doi: 10.1002/csc2.20039

Athwal, N. (2021). *The lifecycle of an almond* (Forbes). Available at: https://www.forbes.com/sites/navathwal/2021/08/30/the-lifecycle-of-an-almond/?sh=3a132ade4610 (Accessed 9 September 2022).

Ball, J. E., Anderson, D. T., and Chan, C. S.Sr. (2017). Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *J. Appl. Remote Sens.* 11 (4), 42609. doi: 10.1117/1.JRS.11.042609

Bergstra, J., Yamins, D., and Cox, D. (2013). "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *International conference on machine learning* (Atlanta GA USA). 115–123.

BestPlaces (2022). (Vacaville, California: Sperling's Best Places). Available at: https://www.bestplaces.net/climate/city/california/vacaville.

BoR (2022). *Solano project, bureau of reclamation*. Available at: https://www.usbr.gov/projects/index.php?id=421 (Accessed 9 September 2022).

Breiman, L. (2001). Random forests. *Mach. Learn.* 45 (1), 5–32. doi: 10.1023/A:1010933404324

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., et al. (2013). API design for machine learning software: Experiences from the scikit-learn project.. *arXiv preprint arXiv*:1309.0238.

Burke, M., and Lobell, D. B. (2017). Satellite-based assessment of yield variation and its determinants in smallholder African systems. *Proc. Natl. Acad. Sci. U. S. A.* 114 (9), 2189–2194. doi: 10.1073/pnas.1616919114

Cai, Y., Kaiyu, G., Jian, P., Shaowen, W., Christopher, S., Brian, W., et al. (2018). A high-performance and in-season classification system of field-level crop types using time-series landsat data and a machine learning approach. *Remote Sens. Environ.* 210, 35–47. doi: 10.1016/J.RSE.2018.02.045

CDFA (2022). *California Agricultural production statistics* (California Department of Food and Agriculture). Available at: https://www.cdfa.ca.gov/Statistics/ (Accessed 7 January 2022).

Cedar Lake Ventures, I. (2022). *Climate and average weather year round in vacaville California, united states*. Available at: https://weatherspark.com/y/1159/Average-Weather-in-Vacaville-California-United-States-Year-Round.

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 (13), 1584. doi: 10.3390/RS11131584

Chen, T., and Guestrin, C. (2016). "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (San Francisco, CA, USA), 785–794.

Chen, B., Jin, Y., and Brown, P. (2019). An enhanced bloom index for quantifying floral phenology using multi-scale remote sensing observations. *ISPRS J. Photogrammetry Remote Sens.* 156 (August), 108–120. doi: 10.1016/j.isprsjprs.2019.08.006

Chollet, F. (2015). *Keras* (GitHub). Available at: https://github.com/fchollet/keras%7D%7D.

Dzotsi, K. A., Basso, B., and Jones, J. W. (2013). Development, uncertainty and sensitivity analysis of the simple SALUS crop model in DSSAT. *Ecol. Model.* 260, 62–76. doi: 10.1016/j.ecolmodel.2013.03.017

FAO (2022). *Food and agriculture organization of united nations (FAO)*. Available at: https://www.fao.org/faostat/en/#data/QCL.

Fulton, J., Norton, M., and Shilling, F. (2019). Water-indexed benefits and impacts of California almonds. *Ecol. Indic.* 96, 711–717. doi: 10.1016/J.ECOLIND.2017.12.063

Gavahi, K., Abbaszadeh, P., and Moradkhani, H. (2021). DeepYield: A combined convolutional neural network with long short-term memory for crop yield forecasting. *Expert Syst. Appl.* 184, 115511. doi: 10.1016/J.ESWA.2021.115511

Goodfellow Ian, J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). "Generative adversarial nets," in *Proceedings of the 27th international conference on neural information processing systems* (Montreal, Quebec, Canada). 2672–2680.

Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *IEEE Trans. Systems Man Cybernetics* 3 (6), 610–621. doi: 10.1109/TSMC.1973.4309314

Harter, T. (2009). Agricultural impacts on groundwater nitrate.

Hodges, T., Botner, D., Sakamoto, C., and Haug Hays, J. (1987). Using the CERES-maize model to estimate production for the US cornbelt. *Agric. For. Meteorol.* 40 (4), 293–303. doi: 10.1016/0168-1923(87)90043-8

Hunt, M. L., Blackburn Alan, G., Carrasco, L., Redhead, J. W., and Rowland, C. S. (2019). High resolution wheat yield mapping using sentinel-2. *Remote Sens. Environ.* 233 (December 2018). doi: 10.1016/j.rse.2019.111410

Jagtap, S. S., and Jones, J. W. (2002). Adaptation and evaluation of the CROPGRO-soybean model to predict regional yield and production. *Agriculture Ecosyst. Environ.* 93 (1–3), 73–85. doi: 10.1016/S0167-8809(01)00358-9

Jin, Z., George, A. A., Calum, Y., Stefania, D. T., Stephen, A., Marshall, B., et al. (2019). Smallholder maize area and yield mapping at national scales with Google earth engine. *Remote Sens. Environ.* 228 (March), 115–128. doi: 10.1016/j.rse.2019.04.016

Jin, Y., Chen, B., Lampinen, B. D., and Brown, P. H. (2020). Advancing agricultural production with machine learning analytics: Yield determinants for california's almond orchards. *Front. Plant Sci.* 11. doi: 10.3389/fpls.2020.00290

Kang, Y., Ozdogan, M., Zhu, X., Ye, Z., Hain, C., Anderson, M., et al. (2020). Comparative assessment of environmental variables and machine learning algorithms for maize yield prediction in the US Midwest. *Environ. Res. Lett.* 15 (6). doi: 10.1088/1748-9326/ab7df9

Kang, Y., and Özdoğan, M. (2019). Field-level crop yield mapping with landsat using a hierarchical data assimilation approach. *Remote Sens. Environ.* 228 (March), 144–163. doi: 10.1016/j.rse.2019.04.005

Kattenborn, T., Kattenborn, T., Leitloff, J., Schiefer, F., and Hinz, S. (2021). Review on convolutional neural networks (CNN) in vegetation remote sensing. *ISPRS J. Photogrammetry Remote Sens.* 173, 24–49. doi: 10.1016/j.isprsjprs.2020.12.010

Keating, B. A., Carberry, P. S., Hammer, G. L., Probert, M. E., Robertson, M. J., Holzworth, D. P., et al. (2003). An overview of APSIM, a model designed for farming systems simulation. *Eur. J. Agron.* 18 (3–4), 267–288. doi: 10.1016/S1161-0301(02)00108-9

Khaki, S., and Wang, L. (2019). Crop yield prediction using deep neural networks. *Front. Plant Sci.* 10. doi: 10.3389/FPLS.2019.00621/BIBTEX

Khaki, S., Wang, L., and Archontoulis, S. V. (2020). A CNN-RNN framework for crop yield prediction. *Front. Plant Sci.* 10 (January). doi: 10.3389/fpls.2019.01750

Khalsa, S. D. S., Rudnick, J., Lubell, M., Sears, M., and Brown, P. H. (2022). Linking agronomic and knowledge barriers to adoption of conservation practices for nitrogen management. *Front. Agron.* 4 (June). doi: 10.3389/fagro.2022.915378

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv* 1412, 6980.

Kuwata, K., and Shibasaki, R. (2016). Estimating corn yield in the United States with modis evi and machine learning methods. ISPRS Ann. *Photogramm. Remote Sens. Spat. Inf. Sci.* 3(8), 131-136.

Lambert, M. J., Sibiry Traoré, PC, Blaes, X, Baret, P, and Defourny, P. (2018). Estimating smallholder crops production at village level from sentinel-2 time series in mali's cotton belt. *Remote Sens. Environ.* 216 (September 2017), 647–657. doi: 10.1016/j.rse.2018.06.036

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

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2021). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Networks Learn. Syst.* 33 (12), 1–21. doi: 10.1109/tnnls.2021.3084827

Liu, Y., Liu, Y., Wang, S., Chen, J., Chen, B., Wang, X., et al. (2022). Rice yield prediction and model interpretation based on satellite and climatic indicators using a transformer method. *Remote Sens.* 14 (19). doi: 10.3390/rs14195045

Lobell, D. B., Cahill, K. N., and Field, C. B. (2007). Historical effects of temperature and precipitation on California crop yields. *Climatic Change* 81 (2), 187–203. doi: 10.1007/s10584-006-9141-3

Lobell, D. B., Thau, D., Seifert, C., Engle, E., and Little, B. (2015). A scalable satellite-based crop yield mapper. *Remote Sens. Environ.* 164, 324–333. doi: 10.1016/j.rse.2015.04.021

Lobell, D. B., and Field, C. B. (2011). California Perennial crops in a changing climate. *Climatic Change* 109 (1), 317–333. doi: 10.1007/s10584-011-0303-6

Ma, Y., Zhang, Z., Kang, Y., and Özdoğan, M. (2021). Corn yield prediction and uncertainty analysis based on remotely sensed variables using a Bayesian neural network approach. *Remote Sens. Environ.* 259. doi: 10.1016/j.rse.2021.112408

Maimaitijiang, M., Sagan, V., Sidike, P., Hartling, S., Esposito, F., and Fritschi, F. (2020). Soybean yield prediction from UAV using multimodal data fusion and deep learning. *Remote Sens. Environ.* 237. doi: 10.1016/j.rse.2019.111599

Medar, R. A., and Rajpurohit, V. S. (2014). A survey on data mining techniques for crop yield prediction. *Int. J. Advance Res. Comput. Sci. Manage. Stud.* 2 (9), 59–64.

Mourtzinis, S., Specht, J., Lindsey, L., Wiebold, W., Ross, J., Nafziger, E., et al. (2015). Climate-induced reduction in US-wide soybean yields underpinned by region-and in-season-specific responses. *Nat. Plants* 1 (February), 8–11. doi: 10.1038/nplants.2014.26

Muruganantham, P., Wibowo, S., Grandhi, S., Samrat, N. H., and Islam, N. (2022). A systematic literature review on crop yield prediction with deep learning and remote sensing. *Remote Sens.* 14 (9). doi: 10.3390/rs14091990

Mutanga, O., and Skidmore, A. K. (2004). Narrow band vegetation indices overcome the saturation problem in biomass estimation. *Int. J. Remote Sens.* 25 (19), 3999–4014. doi: 10.1080/01431160310001654923

Nichol, J. E., and Sarker, M. L. R. (2011). Improved biomass estimation using the texture parameters of two high-resolution optical sensors. *IEEE Trans. Geosci. Remote Sens.* 49 (3), 930–948. doi: 10.1109/TGRS.2010.2068574

O'Malley, T., et al. (2019) *KerasTuner*. Available at: https://github.com/keras-team/keras-tuner.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. large margin classifiers* 10 (3), 61–74.

Rashid, M., Bari, B. S., Yusup, Y., Kamaruddin, M. A., and Khan, N. (2021). A comprehensive review of crop yield prediction using machine learning approaches with special emphasis on palm oil yield prediction. *IEEE Access* 9, 63406–63439. doi: 10.1109/ACCESS.2021.3075159

Rudnick, J., Lubell, M., Khalsa, S. D., Tatge, S., Wood, L., Sears, M., et al. (2021). A farm systems approach to the adoption of sustainable nitrogen management practices in California. *Agric. Hum. Values* 38 (3), 783–801. doi: 10.1007/s10460-021-10190-5

Sakamoto, T., Gitelson, A. A., and Arkebauer, T. J. (2013). MODIS-based corn grain yield estimation model incorporating crop phenology information. *Remote Sens. Environ.* 131, 215–231. doi: 10.1016/j.rse.2012.12.017

Shahhosseini, M., Hu, G., Huber, I., and Archontoulis, S. V. (2021). Coupling machine learning and crop modeling improves crop yield prediction in the US corn belt. *Sci. Rep.* 11 (1), 1–15. doi: 10.1038/s41598-020-80820-1

Shahhosseini, M., Hu, G., and Archontoulis, S. V. (2020). Forecasting corn yield with machine learning ensembles. *Front. Plant Sci.* 11 (July). doi: 10.3389/fpls.2020.01120

SID (2012). *Rules and regulations governing the operation and distribution of irrigation water within the solano irrigation disctrict service area*. Available at: https://www.sidwater.org/DocumentCenter/View/450/SID-Rules-and-Regulations-2012.

Sidike, P., Sagan, V., Maimaitijiang, M., Maimaitiyiming, M., Shakoor, N., Burken, J., et al. (2019). dPEN: deep progressively expanded network for mapping heterogeneous agricultural landscape using WorldView-3 satellite imagery. *Remote Sens. Environ.* 221, 756–772. doi: 10.1016/J.RSE.2018.11.031

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. doi: 10.34133/2022/9757948

Tang, Z., Jin, Y., Alsina, M. M., McElrone, A. J., Bambach, N., and Kustas, W. P. (2022). Vine water status mapping with multispectral UAV imagery and machine learning. *Irrigation Sci.* 1, 1–16. doi: 10.1007/S00271-022-00788-W

Todd, S. W., Hoffer, R. M., and Milchunas, D. G. (1998). Biomass estimation on grazed and ungrazed rangelands using spectral indices. *Int. J. Remote Sens.* 19 (3), 427–438. doi: 10.1080/014311698216071

van Klompenburg, T., Kassahun, A., and Catal, C. (2020). Crop yield prediction using machine learning: A systematic literature review. *Comput. Electron. Agric.* 177 (August). doi: 10.1016/j.compag.2020.105709

Vaswani, A., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., et al. (2017). "Attention is all you need," in *Advances in neural information processing systems*, vol. 30. .

Wang, F., Yi, Q., Hu, J., Xie, L., Yao, X., Xu, T., et al. (2021). Combining spectral and textural information in UAV hyperspectral images to estimate rice grain yield. *Int. J. Appl. Earth Observation Geoinfo.* 102, 102397. doi: 10.1016/J.JAG.2021.102397

Wood, E. M., Pidgeon, A. M., Radeloff, V. C., and Keuler, N. S. (2012). Image texture as a remotely sensed measure of vegetation structure. *Remote Sens. Environ.* 121, 516–526. doi: 10.1016/j.rse.2012.01.003

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)* (Munich, Germany,). 3–19.

WRCC (2022). *Vacaville, CA, West region climate center (WRCC)*. Available at: https://wrcc.dri.edu/cgi-bin/cliMAIN.pl?ca9200 (Accessed 9 September 2022).

Xu, H., Twine, T. E., and Girvetz, E. (2016). Climate change and maize yield in Iowa. *PloS One* 11 (5), 1–20. doi: 10.1371/journal.pone.0156083

Yamashita, R., Nishio, M., Do, R. K., and Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging* 9 (4), 611–629. doi: 10.1007/s13244-018-0639-9

You, J., Li, X., Low, M., Lobell, D., and Ermon, S. (2017). "Deep gaussian process for crop yield prediction based on remote sensing data," in *Thirty-First AAAI conference on artificial intelligence* (San Francisco, California, USA).

Zhang, Z., Jin, Y., Chen, B., and Brown, P. (2019). California Almond yield prediction at the orchard level with a machine learning approach. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.00809

Zhang, L., Nishio, M., Do, R. K., and Togashi, K. (2022). Prediction of oil content in single maize kernel based on hyperspectral imaging and attention convolution neural network. *Food Chem.* 395, 133563. doi: 10.1016/J.FOODCHEM.2022.133563

TYPE Original Research
PUBLISHED 03 April 2023
DOI 10.3389/fpls.2023.1133060

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU), United States

REVIEWED BY
Jun Su,
Fujian Agriculture and Forestry University, China
Sijia Yu,
Rutgers, The State University of New Jersey - Busch Campus, United States

*CORRESPONDENCE
Chunlong Xian
✉ chlxian@scau.edu.cn

SPECIALTY SECTION
This article was submitted to Technical Advances in Plant Science, a section of the journal Frontiers in Plant Science

RECEIVED 28 December 2022
ACCEPTED 21 March 2023
PUBLISHED 03 April 2023

CITATION
Peng H, Xu H, Gao Z, Zhou Z, Tian X, Deng Q, He H and Xian C (2023) Crop pest image classification based on improved densely connected convolutional network. *Front. Plant Sci.* 14:1133060. doi: 10.3389/fpls.2023.1133060

Crop pest image classification based on improved densely connected convolutional network

Hongxing Peng[1,2], Huiming Xu[1], Zongmei Gao[3], Zhiyan Zhou[4], Xingguo Tian[5], Qianting Deng[6], Huijun He[1] and Chunlong Xian[7]*

[1]College of Mathematics and Informatics, South China Agricultural University, Guangzhou, China, [2]Key Laboratory of Smart Agricultural Technology in Tropical South China, Ministry of Agriculture and Rural Affairs, Guangzhou, China, [3]Center for Precision and Automated Agricultural Systems, Department of Biological Systems Engineering, Washington State University, Prosser, WA, United States, [4]College of Engineering, South China Agricultural University, Guangzhou, China, [5]College of Food Science, South China Agricultural University, Guangzhou, China, [6]Department of Asset and Laboratory Management, South China Agricultural University, Guangzhou, China, [7]College of Economics and Management, South China Agricultural University, Guangzhou, China

Introduction: Crop pests have a great impact on the quality and yield of crops. The use of deep learning for the identification of crop pests is important for crop precise management.

Methods: To address the lack of data set and poor classification accuracy in current pest research, a large-scale pest data set named HQIP102 is built and the pest identification model named MADN is proposed. There are some problems with the IP102 large crop pest dataset, such as some pest categories are wrong and pest subjects are missing from the images. In this study, the IP102 data set was carefully filtered to obtain the HQIP102 data set, which contains 47,393 images of 102 pest classes on eight crops. The MADN model improves the representation capability of DenseNet in three aspects. Firstly, the Selective Kernel unit is introduced into the DenseNet model, which can adaptively adjust the size of the receptive field according to the input and capture target objects of different sizes more effectively. Secondly, in order to make the features obey a stable distribution, the Representative Batch Normalization module is used in the DenseNet model. In addition, adaptive selection of whether to activate neurons can improve the performance of the network, for which the ACON activation function is used in the DenseNet model. Finally, the MADN model is constituted by ensemble learning.

Results: Experimental results show that MADN achieved an accuracy and F1Score of 75.28% and 65.46% on the HQIP102 data set, an improvement of 5.17 percentage points and 5.20 percentage points compared to the pre-improvement DenseNet-121. Compared with ResNet-101, the accuracy and F1Score of MADN model improved by 10.48 percentage points and 10.56 percentage points, while the parameters size decreased by 35.37%. Deploying models to cloud servers with mobile application provides help in securing crop yield and quality.

KEYWORDS

pest image classification, selective kernel unit, representative batch normalization, DenseNet-121, ensemble learning

1 Introduction

Agricultural pests have long posed a severe threat to the growth of crops and the storage of agricultural products (Cheng et al., 2017). The Food and Agriculture Organization (FAO) reported that these pests cause between 20 and 40 percent loss of global crop production every year. Because of relatively cheaper operational cost, farmers use a variety of chemicals such as pesticides to control pests, which has a negative impact on the agroecosystem (Geiger et al., 2010). If the location, time and listing of species and populations of invertebrate in the fields were available, instead of heavily relying upon pesticide, integrated pest management would use the optimized combination of mechanical, chemical, biological and genetic tools to mitigate harmful effects and enhance beneficial effects (Liu et al., 2016). Timely and accurate pest detection and classification are of great significance to its prevention and control, and early detection is a prerequisite to making an effective pest management plan and can reduce pollution.

Traditional crop pest classification relies mainly on manual observation or expert guidance, which is slow, inefficient, costly, and subjective. With the development of machine learning methods and computer vision techniques, researchers are beginning to use information technology to identify images of crop pests. The traditional machine learning classification framework consists of two main modules: the feature representation of the pest and the classifier. The normal used hand-crafted features include GIST (Oliva and Torralba, 2001), Scale Invariant Feature Transform (SIFT) (Lowe, 2004), Speeded Up Robust Feature (SURF), etc. The main classifiers commonly used include K-nearest neighbor classification algorithms (KNN), Support Vector Machines (SVM), etc. It is difficult to determine which of many features is optimal, and if the feature extraction is not correct, the subsequent classifier will make mistakes in identifying pests. With the advent of efficient learning algorithms for deep learning, it has achieved significant improvements in classification accuracy on many traditional classification tasks (Krizhevsky et al., 2017). In particular, convolutional neural networks (CNNs) are rapidly becoming the method of choice for overcoming certain challenges (Barbedo, 2018).

Recently, smart agriculture has been introduced to apply artificial intelligence (AI) technology, information and wireless communication technology applications. In addition, crop health monitoring is considered to be a major application of smart agriculture (Ayaz et al., 2019). Researchers are gradually turning their attention to designing mobile applications to identify pests. Karar et al. (2021) designed a mobile application using technologies such as Apache Cordova framework and Flask Web, and achieved good results in pest identification using deep learning techniques, but it used a relatively small dataset and identified only five categories of pests. Deep learning-based pest detection requires a large number of pest samples for supervised learning (Liu and Wang, 2021), and building an application that can identify multiple classes of pests in common crops is also in urgent need of development. It is well known that the ImageNet Large Scale Visual Classification Challenge (ILSVRC) (Deng et al., 2009) marks the beginning of the rapid development of deep learning, demonstrating that large-scale image data set play a key role in driving deep learning progress. However, most deep learning methods on insect pests are limited to small data set, and most public data set are collected indoor, which does not meet the needs of insect pest classification in field conditions. The IP102 large pest data set (Wu et al., 2019), which contains 75,222 images with a total of 102 classes from 8 crops, has alleviated this problem to some extent. However, the data set suffers from poor screening and misplaced pest categories, with a reported classification accuracy of only 49.4%. To address this issue, we invited agricultural experts and volunteers to further screen the IP102 data set. The new data set is of Higher Quality compared to IP102 and is named HQIP102.

The context of pest images in real environments is complex and suffers from large intra-class variation and small inter-class variation of pests. Existing models such as Densenet and ResNet do not work well on large pest datasets. To better identify larger pest data set, the DenseNet network (Huang et al., 2017), which performed well in the ImageNet task, is used as the base network. To improve the pest classification accuracy, we propose the MADN convolutional neural network model, which improves DenseNet-121 in three aspects: channel attention mechanism, input information feature enhancement and adaptive activation function. These improvements can improve the model's pest classification performance.

The goal and objectives of our study are summarized as follows:

·Two criteria are used to further filter the IP102 large pest data set and improve the overall quality of the original data set, named HQIP102.

·Several techniques and the MADN convolutional neural network model are proposed to improve the representation capability of the DenseNet-121 network and improve its classification accuracy on large pest data set.

2 Related work

Research on crop pest classification based on computer vision has been a hot topic. In recent years, many computer-aided insect pest classification systems (Rani and Amsini, 2016; ; Alfarisy et al., 2018) are presented in the vision community. The methods involved mainly include machine learning and deep learning.

Machine learning often uses hand-crafted features such as SIFT, HOG (Dalal and Triggs, 2005), etc. Hand-crafted feature-based methods are the primary solutions for insect pest classification traditionally (Wu et al., 2019). Bisgin et al. (2018) used SVM to classify feature information such as size, color, basic pattern and texture extracted from 15 classes of food beetles, ultimately obtaining good classification results on a data set of 6900 images. Ebrahimi et al. (2017) designed an SVM structure with difference kernel function for thrips detection using the ratio of major diameter to minor diameter as region index as well as Hue, Saturation and Intensify as color indexes with a mean error of less than 2.25% for the best classification. Xiao et al. (2018) used SIFT image descriptor as well as SVM classifier to identify four important vegetable pests Whiteflies, Phyllotreta Striolata, Plutella Xylostella and Thrips with an average accuracy of 91.56% on 80 experimental images. Traditional machine learning algorithms rely on complex image processing techniques and handcrafted features, which often have limited robustness and generalization on large data set.

The successful application of deep learning in other fields has led to an increasing interest in agriculture, which is currently the most cutting-edge, modern, and promising technology (Kamilaris and Prenafeta-Boldú, 2018). Tetila et al. (2020) used transfer learning strategy to fine-tune Inception-v3, Resnet-50, VGG-16, VGG-19 and Xception to identify a data set containing 5000 soybean pest images. It has better performance compared to traditional feature extraction methods such as SIFT and SURF. Liu and Chahl (2021) used a novel approach to generate a virtual database that was successfully used to train a deep residual CNN with 97.8% accuracy in detecting four pests in agricultural environments. Khanramaki et al. (2021) proposed an ensemble classifier of deep convolutional neural networks to identify three common citrus pests with 99.04% accuracy on a data set containing 1774 images of citrus leaves. Ayan et al. (2020) used a weighted voting method to ensemble the pre-trained Inception-V3, Xception and MobileNet, which was named GAEnsemble, and its classification accuracy on the IP102 data set was 67.13%. Unlike Ayan et al. (2020), which used a fine-tuning strategy to combine existing models, this paper improves the DenseNet network and uses ensemble learning to combine the improved models.

Existing studies have shown that small datasets containing only a few pest classes have higher identification accuracy, while classification accuracy is low on the large data set IP102. To address the problem of misplacing pest categories in the IP102 data set, we built a Higher Quality pest data set named HQIP102. We also proposed the MADN convolutional neural network model for improving classification accuracy of existing models.

3 Materials and methods

3.1 Data set construction

Since IP102 contains more than 70,000 images of 102 categories, it inevitably has problems such as misplacement of some pest categories and lack of detailed screening.

To obtain a higher quality pest data set, we invited agricultural experts and volunteers to further screen the IP102 data set according to the following two criteria. (1) obviously misplaced categories; (2) basically background, does not contain any target objects. The new data set is of higher quality and is named HQIP102. Low quality images are removed directly from the data set, Then the HQIP102 contains 102 pest categories for eight crops, including rice and wheat etc. Some of the pest image samples are shown in Figure 1.

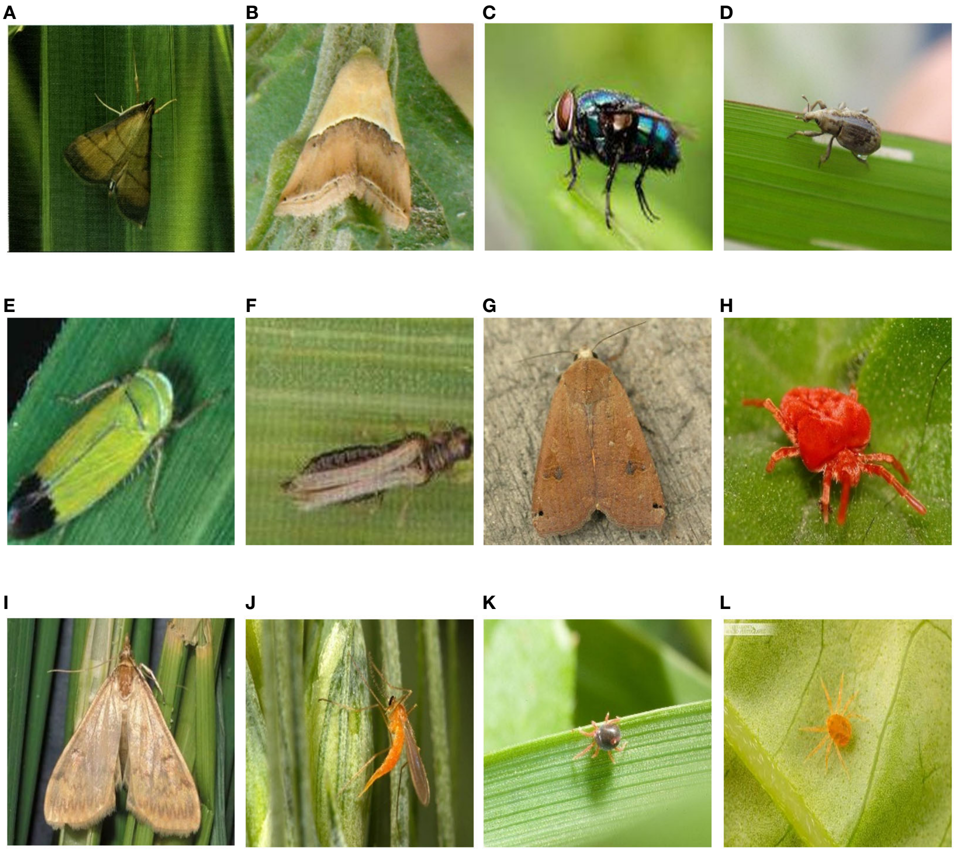


FIGURE 1
Sample images of some pests **(A)** rice leaf roller; **(B)** rice leaf caterpillar; **(C)** paddy stem maggot; **(D)** rice water weevil; **(E)** rice leafhopper; **(F)** grain spreader thrips; **(G)** yellow cutworm; **(H)** red spider; **(I)** corn borer; **(J)** wheat blossom midge; **(K)** penthaleus major; **(L)** longlegged spider mite;.

As can be seen in Figure 1, the pest background in the HQIP102 data set is complex, the main part of the pest is small, and the similarity between some pest categories is high, which increases the overall classification difficulty. HQIP102 was filtered for each category of pests in IP102, with fewer images remaining for low quality pest categories, and the final HQIP102 pest data set contained 47,393 images. A comparison of HQIP102 with IP102 on eight crops corresponding to the pest category as well as the number of pests is shown in Table 1.

As can be seen from Table 1, HQIP102 filtered out more images on Rice, Corn, Beet, and Alfalfa, while fewer pest images were removed on the Wheat, Vitis, Citrus, and Mango categories. Among Rice crops, the rice leaf roller and asiatic rice borer categories have a higher number of deletions. In Corn crops, the corn borer and aphids categories removed more images. There are more images deleted from the beet army worm class in the Beet crop. In Alfalfa crops, alfalfa plant bug and blister beetle classes have more images deleted.

3.2 Data set split and dynamic data augmentation

The data set is divided into training set, validation set and test set according to the ratio of 7.5:1:1.5. The number of samples for certain pests in the data set is insufficient, and the use of data augmentation can increase the amount of data available for training, thus improving the generalization ability of the model. After splitting the data set, a dynamic data expansion method based on the number of pests in each class is proposed in this paper in order to solve the data imbalance problem in the HQIP102 training set, see Eq.1.

$$N = \begin{cases} 12N, 0 < N \leq 30 \\ 7N, 30 < N \leq 60 \\ 4N, 60 < N \leq 100 \\ 3N, 100 < N \leq 150 \\ 2N, 150 < N \leq 200 \end{cases} \tag{1}$$

Where Ndenotes the number of images in the training set for a particular type of pest. $N$is determined based on the average number of images of the pest category in the data set. The average number of images per pest category in the IP102 dataset is 460. And the specific pest image increase multiplier in the Eq.1 is adjusted manually, in which the range of the parameter N and the number of additional images are obtained by manual setting, to achieve the right amount of supplementary pest image data. With dynamic data augmentation, the data imbalance can be mitigated with a small amount of additional data, which is the basis for the parameter determination in Eq.1.

The data augmentation methods used were mainly a combination of center cropping, brightness contrast saturation adjustment, random horizontal flip, and random vertical flip. Specifically, the image is cropped to a size of 224 × 224 and has a 50% probability of random horizontal flipping and random vertical flipping. The probability of brightness and contrast adjustment is also 50%. The images are then saved to the original dataset after using data augmentation.

Using dynamic data enhancement, the total number of HQIP102 pest data set increased from 47,393 to 62,060 images, with the training set increasing from 35,607 to 50,274 and the validation and test sets remaining unchanged with 4734 and 7052. After using data augmentation, the ratio of training set, test set and validation set is about 8:1:1.

3.3 Dense convolutional network (DenseNet)

DenseNets (DenseNet-121, DenseNet-169, DenseNet-201, and DenseNet-264) alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and reduces the number of parameters to some extent. In addition, the structure used by DenseNets shows good performance on large ImageNet datasets. For each layer, the feature-maps of all preceding layers are used as inputs, and its own feature maps are used as inputs into all subsequent layers. As shown in Figure 2, the network

TABLE 1 Comparison of HQIP102 and IP102 on 8 crops.

Crop Category	Number of pest categories	IP102 Total	HQIP102 Total
Rice	14	8417	3006
Corn	13	14015	6373
Wheat	9	3418	2110
Beet	8	4420	1942
Alfalfa	13	10390	5611
Vitis	16	17551	14555
Citrus	19	7272	5173
Mango	10	9739	8623
Total	102	75222	47393

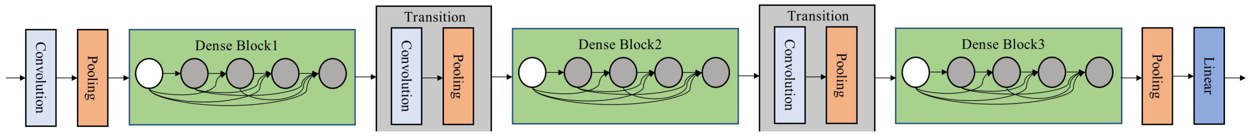


FIGURE 2
Structure of DenseNet with three dense blocks.

structure of DenseNet consists mainly of Dense Block and Transition.

In Dense Block, each layer has the same feature map size and can be concatenated in the channel dimension. All layers in the Dense Block output $k$feature maps after convolution, where the hyperparameter $k$is called the growth rate. We refer to each layer in a Dense Block as its substructure. Assuming that the number of channels in the feature map of the input layer is k_0, then the number of channels in the input of layer $l$is $k_0 + k(l - 1)$.

The Dense Block inside the DenseNet-B structure uses bottleneck layers to reduce the amount of computation. Transition layer, is mainly used to connect two adjacent Dense Blocks, and to reduce the size of the feature map. Its structure is Batch Normalization (BatchNorm) + ReLU + 1×1 Convolution + 2×2 AvgPooling. The Transition layer of the DenseNet-C structure also introduces a compression factor $\theta(<1)$, which reduces the number of features in the output. When using bottleneck layers as well as transition layers with $\theta(<1)$, such a model is called DenseNet-BC.

3.4 MADN convolutional neural network

The MADN model focuses on improving the Dense Block structure in DenseNet in three ways, while the rest of the model is consistent with DenseNet. It introduces the Selective Kernel Unit (MADN-SK), the Representative Batch Normalization (MADN-RBN) module, and the ACON activation function (MADN-ACON) into the DenseNet. It is worth noting that MADN is not an end-to-end model, but combines 3 improved DenseNet models. Specifically, Using DenseNet-121 as the base network, MADN-SK, MADN-RBN and MADN-ACON are combined through ensemble learning to form the entire MADN model as shown in Figure 3. A detailed architectural comparison of DenseNet-121 with MADN-SK, MADN-RBN and MADN-ACON is shown in Table 2.

Sections 3.4.1 to 3.4.3 are the improvements of three aspects of DenseNet-121 in this study, each individual improvement is a complete model, and the final three models named MADN-SK, MADN-RBN, and MADN-ACON are obtained. Section 3.4.4 is an introduction to the ensemble learning used in this paper.

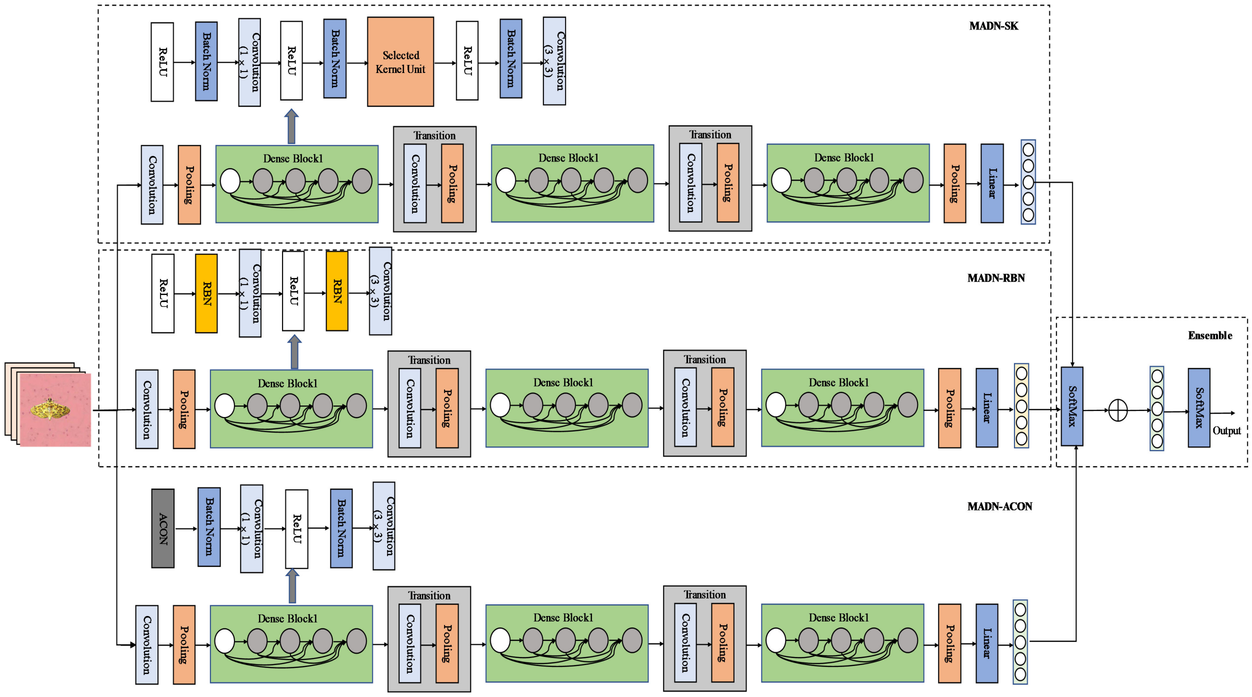


FIGURE 3
Structure of the MADN network model. The dense connection lines are omitted from the diagram, and the connections are made in the same way as the original DenseNet.

TABLE 2 Structural comparison of DenseNet-121 and modified models.

Layers	Output Size	DenseNet121	MADN_SK	MADN_RBN	MADN_ACON
Convolution	112×112	BN-ReLU-7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			

Dense Block(1)	56×56	BNReLU conv1;
		BN ReLu conv2
		(6x)
			ReLu BN conv1;
			ReLu BN SK;
			ReLu BN conv2
			(6x)
				ReLU RBN conv1; ReLu RBN conv2
				(6x)
					ACON BN conv1; ReLu BN conv2
					(6x)

Transition Layer (1)	56×56	BN-ReLU-1×1 conv			
	28×28	2×2 average pool, stride 2			

Dense Block(2)	28×28	BN ReLU conv1;
		BN ReLu conv2
		(12x)
			ReLu BN conv1;
			ReLu BN SK;
			ReLu BN conv2
			(12x)
				ReLU RBN conv1; ReLu RBN conv2
				(12x)
					ACON BN conv1; ReLu BN conv2
					(12x)

Transition Layer (2)	28×28	BN-ReLU-1×1 conv			
	14×14	2×2 average pool, stride 2			

Dense Block(3)	14×14	BN ReLU conv1;
		BN ReLu conv2
		(24x)
			ReLu BN conv1;
			ReLu BN SK;
			ReLu BN conv2
			(24x)
				ReLU RBN conv1; ReLu RBN conv2
				(24x)
					ACON BN conv1; ReLu BN conv2
					(24x)

Transition Layer (3)	14×14	BN-ReLU-1×1 conv			
	7×7	2×2 average pool, stride 2			

Dense Block(4)	7×7	BN ReLU conv1;
		BN ReLu conv2
		(16x)
			ReLu BN conv1;
			ReLu BN SK;
			ReLu BN conv2
			(16x)
				ReLU RBN conv1; ReLu RBN conv2
				(16x)
					ACON BN conv1; ReLu BN conv2
					(16x)

Classification Layer	1×1	7×7 global average pool			
		102D fully-connected, softmax			

where conv1 denotes a 1×1 convolution, and conv2 denotes a 3×3 convolution. MADN_SK, MADN_RBN, and MADN_ACON are the structures of the above modified DenseNet.

3.4.1 MADN-SK

Li et al. (2019) proposes a dynamic selection mechanism in CNNs that allows each neuron to adaptively adjust its receptive field size based on multiple scales of input information. Figure 4 shows the building blocks of the Selective Kernel (SK) unit.

In this building block, multiple branches with different kernel sizes are fused with softmax attention guided by information from these branches. The MADN-SK network is capable of adaptively adjusting the size of the receptive field according to the input to effectively capture target objects of different sizes, and its improved Dense Block substructure is shown in Figure 4.

3.4.2 MADN-RBN

The BatchNorm module is widely used as it allows for more stable training of models. However, its centralization and scaling steps need to rely on the variance obtained from the sample statistics, ignoring the representation differences among instances. Gao et al. (2021) propose to add a simple yet effective feature calibration scheme into the centering and scaling operations of BatchNorm, namely Representative BatchNorm (RBN). The RBN is also divided into two steps: centering calibration and scaling calibration. For the entire process, see Eq.2.

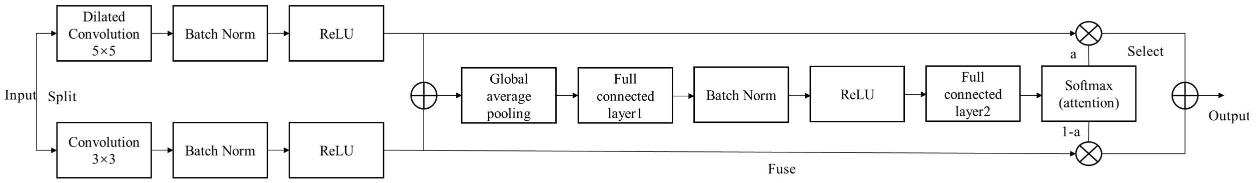


FIGURE 4
SK unit construction.

Centering Calibration:

$$X_{\mathrm{cm}} = X + w_m K_m \quad ,$$

Centering:

$$X_m = X_{cm} - E(X_{cm}) \quad ,$$

Scaling:

$$X_s = \frac{X_m}{\sqrt{Var(X_{cm}) + \epsilon}} \quad , (2)$$

Scaling Calibration:

$$X_{cs} = X_s R(w_v K_s + w_b) \quad ,$$

Affine:

$$Y = X_{cs}\gamma + \beta$$

Where the input features $X \in R^{N\times C\times H\times W}$, w_m, w_v, $w_b$are the learnable weight vector. K_m, $K_s$represent the statistics of feature of each instance, which can be obtained using global average pooling. $R()$is a restriction function, often using sigmoid. $E(X)$and $Var(X)$ denote the mean and variance and are used for centering and scaling. $\gamma$and $\beta$are learned scale and bias factors for affine transformation, and ϵ is used to avoid zero variance.

The use of RBN to replace BN in DenseNet-121 allows better identification of crop pests, and experiments were conducted to verify this.

3.4.3 MADN-ACON

Ma et al. (2021) propose a simple, effective, and general activation function ActivateOrNot (ACON), which learns to activate the neurons or not. ACON-C, see Eq. 3. ACON-C is one of the better-performing activation functions in ACON.

$$(p_1 - p_2)x \cdot \sigma(\beta(p_1 - p_2)x) + p_2 x \tag{3}$$

where β, $p_1$and $p_2$are learnable parameters and are channel-wise, the parameters are initialised randomly. We introduce ACON into the MADN model, which can improve the performance of the whole network.

3.4.4 Ensemble learning

In the area of decision and risk analysis, information from several experts is aggregated by the decision maker, which can improve the accuracy of forecasts. For the ensemble of MADN-SK, MADN-RBN, MADN-ACON we considered the outputs of their classification layers, which determined the confidence values for each pest category. We used the sum of the normalized confidence values for each pest category on these three models as the final measure, see Eq.4.

$$p'i = \frac{\sum_{j=1}^{m} p_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}}, i = 1, \ldots, n \tag{4}$$

Where p_{ij}denotes the confidence value of the j-th network output for the i-th type of pest (in this paper $m = 3$, $n = 102$). p_i' denotes the normalized value of the combined three network confidence values. The i-th pest label corresponding to the largest $p_i'$is chosen as the final prediction.

3.5 Experiment settings

To ensure fairness in the experimental comparisons, all experiments were built under the same conditions. The experiments were conducted on Ubuntu 18.04 with Intel(R) Core (TM) i9-10900K CPU and NVIDIA RTX3090 GPU with 24G memory. The RAM used is 32GB of DDR4, the deep learning tool is Pytorch 1.8, and the CUDA version is 11.4.The size of the input image was fixed at 224 ×224 and the optimizers were all used Adam (Adaptive momentum) (Kingma and Ba, 2014), the batch size was set to 64, the number of iterations was set to 50, and the learning rate was initialized to 0.001. The learning rate was reduced to half of the original rate if the model showed an increase in loss on the validation set during training.

3.6 Evaluation metrics

To better measure the classification performance of different models on the HQIP102 dataset, we chose Accuracy, Precision, Recall and F1Score as the evaluation metrics of the models.

Accuracy (Acc): The proportion of results predicted to be correct to the total sample, see Eq.5.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{5}$$

Precision (Pre): The probability that all samples with a positive prediction are actually positive, see Eq.6.

$$\Pr e = \frac{TP}{TP + FP} \times 100\% \tag{6}$$

Recall (Rec): The probability of all samples that are actually positive being predicted to be positive, see Eq.7.

$$\mathrm{Re}\, c = \frac{TP}{TP + FP} \times 100\% \tag{7}$$

F1Score (F1): The harmonic mean of precision and recall, see Eq.8.

$$F1 = \frac{2 \times \Pr e \times \mathrm{Re}\, c}{\Pr e + \mathrm{Re}\, c} \times 100\% \tag{8}$$

In equations (5-7), TP indicates a true positive: the predicted is a positive sample and the actual is also a positive sample. TN indicates true negative: predicted negative sample, actual negative sample. FP indicates false positive: predicted positive sample, actual negative sample. FN indicates false negative: predicted negative sample, actual positive sample.

In addition, the model parameters, the GPU memory occupied during training, and the total training time were also used to measure the overall performance of the model. In particular, use the nvidia command in ubuntu to view the model's GPU memory occupation, and the torch summary package in Pytorch to view the model's parameters. Also, the inference time of each model for a single pest image is taken into account.

4 Results and discussion

4.1 Dynamic data augmentation experiments

On the training set of the original HQIP102 data set, we performed dynamic data augmentation based on the number of images of each type of pest. Using DenseNet-121 as the base network, the experimental results on the test set are shown in Table 3, keeping all factors consistent except for the different training data. As can be seen from Table 3, compared to the original data set, the DenseNet-121 network improved the accuracy by 0.41% and the F1 by 1.46%, the MADN network improved the accuracy by 1.15% and the F1 by 1.81%.Experiments show that the use of dynamic data augmentation techniques alleviates the problems caused by data imbalance to some extent with a small increase in the number of training samples.

4.2 Ablation experiments and comparative analysis

Ablation experiments were conducted to demonstrate the effectiveness of a series of improvements to the DenseNet-121 model. Accuracy and F1Score on the test set were used as metrics. The ablation experiments include the effect of using only SK units, RBN modules, ACON activation function and the final model after using ensemble learning. The Dense Block of DenseNet has been modified. When the SK unit is introduced, the model is named MADN-SK; when the RBN module is used, the model is named MADN-RBN, and when the ACON activation function is used to replace ReLU, the model is named MADN-ACON. Using ensemble learning to combine the advantages of the three modified models, the final model is named MADN. The results of the ablation experiments on the test set are shown in Table 4.

As can be seen in Table 4, the improved MADN-SK, MADN-RBN, MADN-ACON and MADN all show better accuracy and F1Score compared to the DenseNet-121 model. MADN-SK obtained by introducing the Selective Kernel unit, which improved the accuracy on the test set by 1.94 percentage points and the F1Score by 2.1 percentage points compared to the pre-modified DenseNet-121;MADN-RBN, obtained using Representative BatchNorm, improved the accuracy and F1Score on the test set by 1.03 percentage points and 0.74 percentage points respectively; The MADN-ACON using the ACON activation function showed an accuracy improvement of 1.32 percentage points and an F1Score improvement of 0.8 percentage points on the test set. The MADN model using ensemble learning improved better, with accuracy and F1Score improvements of 4.76 and 4.34 percentage points respectively. As can be seen in Figure 5, During 50 iterations of training, the accuracy of the model gradually smoothed out on the validation set. And the improved MADN-SK, MADN-RBN and MADN-ACON have higher accuracy on the validation set compared to the original DenseNet-121 as the number of training iterations increases. From the experimental results in Table 4, it can be concluded that the improved MADN-

TABLE 3 Dynamic data augmentation comparison experiments.

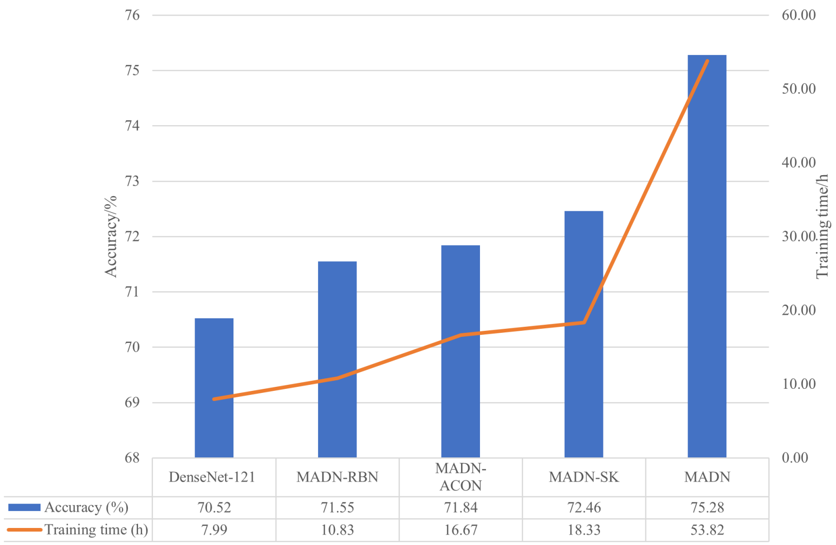
Data set	Method	Acc (%)	Pre (%)	Rec (%)	F1 (%)
HQIP102	DenseNet-121	70.11	61.43	58.96	59.66
HQIP102*	DenseNet-121	70.52	63.21	60.09	61.12
HQIP102	MADN	74.13	67.94	60.78	63.65
HQIP102*	MADN	**75.28**	**69.56**	**62.91**	**65.46**

HQIP102* indicates the HQIP102 data set after using dynamic data augmentation. The bold values indicate the best values in this experiment.

TABLE 4 Results of ablation experiments on the HQIP102 test set.

Model	Improvement method			Acc (%)	F1 (%)
	Selective Kernel unit	Representative BatchNorm	ACON activation		
DenseNet-121				70.52	61.12
MADN_SK	√			72.46	63.22
MADN_RBN		√		71.55	61.86
MADN_ACON			√	71.84	61.92
MADN	√	√	√	**75.28**	**65.46**

MADN is composed by ensemble learning. The bold values indicate the best values in this experiment.



	DenseNet-121	MADN-RBN	MADN-ACON	MADN-SK	MADN
Accuracy (%)	70.52	71.55	71.84	72.46	75.28
Training time (h)	7.99	10.83	16.67	18.33	53.82

FIGURE 5
Comparison of training time and test set accuracy for DenseNet-121 and improved models.

SK, MADN-RBN, MADN-ACON and MADN are valid in improving the accuracy and F1Score compared to the origin DenseNet-121.

We compare the accuracy and training time of the DenseNet-121 as well as the improved classification model in Figure 5.

As can be seen in Figure 5, the improved MADN-RBN, MADN-ACON, and MADN-SK have improved accuracy on the test set at the expense of training time. MADN uses an ensemble learning strategy that requires pre-training of the MADN-RBN, MADN-ACON and MADN-SK models, so it requires more training time, but also higher accuracy on the test set. Although the training phase of a CNN model is usually time-consuming, it does not matter for the classification task, since the classifier is trained offline.

4.3 Comparison experiments with other models

To better evaluate the performance of the improved MADN-SK, MADN-RBN, MADN-ACON, and MADN in this paper,

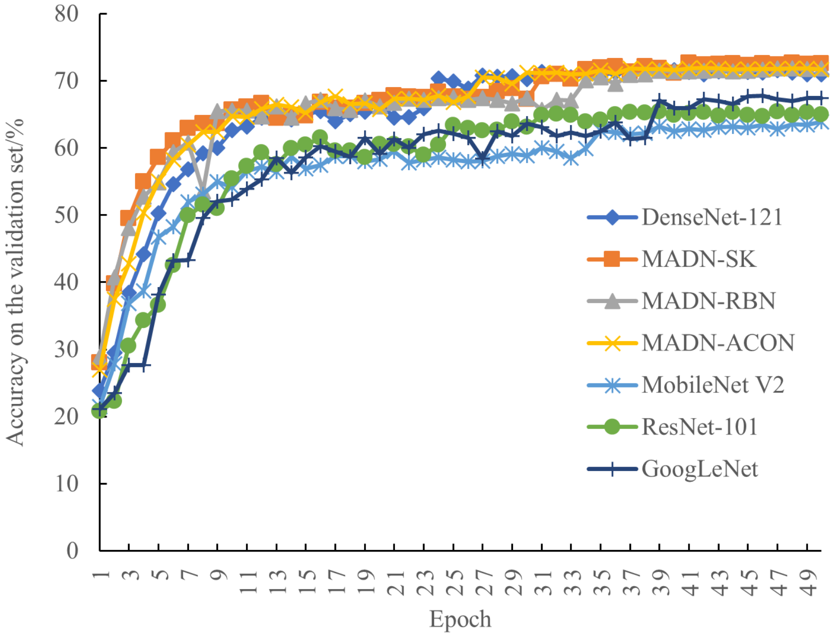


FIGURE 6
Classification accuracy of the model for each iteration on the validation set.

accuracy, precision, recall, F1Score, GPU memory, training time, and parameters of the model were used as measures against ResNet-101 (He et al., 2016), GoogLeNet (Szegedy et al., 2015), MobileNet V2 (Sandler et al., 2018) for comparison experiments. The accuracy of each iteration on the validation set during training is shown in Figure 6, and the final experimental results on the test set are shown in Table 5.

As can be seen in Figure 6, the performance of each model on the validation set tends to stabilize as the iterations progress. Compared to the ResNet-101 and GoogLeNet models, MobileNet V2 performed relatively poorly. And compared to the other models, the improved MADN-SK, MADN-RBN and MADN-ACON show higher classification accuracy on the validation set.

As can be seen in Table 5, the lightweight model MobileNet V2 is optimal in terms of GPU capacity, training time and number of parameters, but performs poorly in terms of accuracy and F1Score on the test set; And compared to ResNet-101, GoogLeNet has a somewhat better overall performance; Although the improved MADN require more GPU memory and longer training time for training, they have better accuracy and F1Score compared to other models, and fewer number of parameters compared to the ResNet-101 model, which is more suitable for the practical needs of identifying pests and more suitable for deployment to cloud servers. Although the inference time of the MADN proposed in this paper is longer for a single pest image compared to other models, the application scenario of this study is to deploy the model to a cloud server, and the network transmission on the cloud server is inherently delayed, so the focus task of this study is to achieve better pest identification accuracy.

4.4 Experimental comparison of MADN and DenseNet-121 at the crop level

Considering the need for pest classification at the specific crop level, the test set accuracy of the improved MADN and DenseNet-121 models were compared on eight crops, as shown in Table 6.

From Table 6 we can see that the MADN network has improved accuracy for all eight crops, with classification accuracy exceeding 80% for both Vitis and Mango crops, an respective improvement of 3.91% and 5.2% compared to the pre-improvement DenseNet-121. Accuracy improvements were greater on Alfalfa and Wheat at 6.23% and 6.09% respectively. The accuracy of the model on different crops may be related to the size of the main part of the pest in different crops and the influence of background disturbances.

TABLE 6 Experimental results of MADN and DenseNet-121 on eight crops test set.

Crop-Class	DenseNet-121	MADN
	Test set Acc	
Rice	59.68	**63.51**
Corn	70.54	**75.82**
Wheat	47.44	**53.53**
Beet	58.19	**64.11**
Alfalfa	61.08	**67.31**
Vitis	78.75	**82.66**
Citrus	68.54	**72.98**
Mango	75.37	**80.57**

The bold values indicate the best values in this experiment.

5 Conclusion

In this study, we filtered the IP102 data set and proposed a higher quality HQIP102 data set for pest classification, which includes 102 pest categories from eight crops with more than 40,000 images. To address the data imbalance, a dynamic data augmentation method is proposed, and the effectiveness of the method is experimentally demonstrated. The accuracy of the DenseNet-121 and MADN models on the HQIP102 dataset was improved by 0.41 and 1.15 percentage points, respectively, after using the data augmentation method. To resolve the issue of low classification accuracy of existing deep learning models on large pest data set, the DenseNet-121 was selected as the base network to be improved. In details, the DenseNet-121 was improved in three ways, i.e., MADN-SK, MADN-RBN and MADN-ACON networks. Also, such networks were combined to propose the MADN network. Validation experiments results showed the effectiveness of these improved methods was potential *via* increased accuracy, precision, recall and F1Score. Compared with the original DenseNet-121, the accuracy and F1Score of the MADN model on

TABLE 5 Performance of the model on the test set.

Model	Test set				Training phase		Parameters size (MB)	Inference time (ms)
	Acc (%)	Pre (%)	Rec (%)	F1(%)	GPU Memory (MB)	Training time(h)		
ResNet-101	64.8	56.88	54.19	54.9	11157	9.65	162.92	82.34
GoogLeNet	67.68	59.66	57.39	57.88	5687	2.85	21.76	17.67
MobileNet V2	63.63	55.65	53.79	54.25	6133	**2.44**	**8.98**	**13.41**
MADN	**75.28**	**69.56**	**62.91**	**65.46**	–	53.82	105.29	290.75

Since MADN is not an end-to-end network, it comes from combining 3 improved DenseNet networks by ensemble learning. Therefore, MADN cannot be trained alone, so "-" is used to indicate that the item does not exist. The bold values indicate the best values in this experiment.

the HQIP102 dataset improved by 4.76 and 4.34 percentage points, respectively. We also carried out analysis at the crop species level, and experiments showed that the MADN network was more accurate for pest classification in Vitis and Mango, which could also be useful for related crop studies. Overall, the proposed deep networks will be helpful for crop pest precise management.

MADN is a combination of 3 improved DenseNet-121 models by ensemble learning, which cannot be trained end-to-end, and needs to train MADN-SK, MADN-ACON and MADN-RBN models first, so the consumption of inference time and training time are larger. In future work, we consider using end-to-end lightweight networks to reduce the training and inference time in scenarios with high requirements for recognition speed.

There are several possible reasons why MADN networks do not significantly improve prediction accuracy.

1. the HQIP102 dataset contains a large number of pest categories, and the similarity between different categories is large.

2. the background interference of pests is large, and the improved method can only improve the classification accuracy to a certain extent.

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

HP and CX designed research. HX and HH conducted experiments and data analysis. HX wrote the manuscript. ZG and ZZ revised the manuscript. XT and QD are responsible for contacting experts to filter the data set. All authors contributed to the article and approved the submitted version.

Funding

This research was supported by Guangzhou Science and Technology Project (Grant No.202002020016,202102080337), Project of Rural Revitalization Strategy in Guangdong Province (Grant No. 2020KJ261), National Natural Science Foundation of China (Grant No. 61863011,32071912), Key Research and Development Program of Guangdong Province (Grant No.2019B0202170003), Natural Science Foundation of Guangdong Province (Grant No. 2018A030313330), Second Batch of Industry-Education Cooperation Collaborative Projects in 2019, Ministry of Education (Grant No. 201902062040), Guangzhou Key Laboratory of Intelligent Agriculture (201902010081), Applied Science and Technology Special Fund Project, Meizhou (Grant No. 2019B0201005) and Project of Ser-vice System of Agricultural Technology Extension and Policy advisory in Colleges and Universities, Agricultural and Rural Department of Guangdong Province (Grant No. F19406).

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

Alfarisy, A. A., Chen, Q., and Guo, M. (2018). "Deep learning based classification for paddy pests & diseases recognition," in *Proceedings of 2018 International Conference on Mathematics and Artificial Intelligence - (ICMAI '18)*. Association for Computing Machinery, New York, NY, USA, 21–25. doi: 10.1145/3208788.3208795

Ayan, E., Erbay, H., and Varçın, F. (2020). Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks. *Comput. Electron. Agric.* 179, 105809. doi: 10.1016/j.compag.2020.105809

Ayaz, M., Ammad-Uddin, M., Sharif, Z., Mansour, A., and Aggoune, E. H. M. (2019). Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk. *IEEE Access* 7, 129551–129583. doi: 10.1109/ACCESS.2019.2932609

Barbedo, J. G. (2018). Factors influencing the use of deep learning for plant disease recognition. *Biosyst. Eng.* 172, 84–91. doi: 10.1016/j.biosystemseng.2018.05.013

Bisgin, H., Bera, T., Ding, H., Semey, H. G., Wu, L., Liu, Z., et al. (2018). Comparing SVM and ANN based machine learning methods for species identification of food contaminating beetles. *Sci. Rep.* 8 (1), 1–12. doi: 10.1038/s41598-018-24926-7

Cheng, X., Zhang, Y., Chen, Y., Wu, Y., and Yue, Y. (2017). Pest identification *via* deep residual learning in complex background. *Comput. Electron. Agric.* 141, 351–356. doi: 10.1016/j.compag.2017.08.005

Dalal, N., and Triggs, B. "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, California, 886–893. doi: 10.1109/cvpr.2005.177

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Li, F.-F. (2009). "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 248–255. doi: 10.1109/cvpr.2009.5206848

Ebrahimi, M. A., Khoshtaghaza, M. H., Minaei, S., and Jamshidi, B. (2017). Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* 137, 52–58. doi: 10.1016/j.compag.2017.03.016

Food and Agriculture Organization of the United Nations (FAO) (2020) *New standards to curb the global spread of plant pests and diseases*. Available at: http://www.fao.org/news/story/en/item/1187738/icode/ (Accessed 16-05-2022).

Gao, S. H., Han, Q., Li, D., Cheng, M. M., and Peng, P. (2021). "Representative batch normalization with feature calibration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA, 8028–8038. doi: 10.1109/CVPR46437.2021.00856

Geiger, F., Bengtsson, J., Berendse, F., Weisser, W. W., Emmerson, M., Morales, M. B., et al. (2010). Persistent negative effects of pesticides on biodiversity and biological control potential on European farmland. *Basic Appl. Ecol.* 11 (2), 97–105. doi: 10.1016/j.baae.2009.12.001

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

Huang, G., Liu, Z., Maaten, L.v. d., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision* *and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2261–2269. doi: 10.1109/cvpr.2017.243

Kamilaris, A., and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Comput. Electron. Agric.* 147, 70–90. doi: 10.1016/j.compag.2018.02.016

Karar, M. E., Alsunaydi, F., Albusaymi, S., and Alotaibi, S. (2021). A new mobile application of agricultural pests recognition using deep learning in cloud computing system. *Alexandria Eng. J.* 60 (5), 4423–4432. doi: 10.1016/j.aej.2021.03.009

Khanramaki, M., Askari Asli-Ardeh, E., and Kozegar, E. (2021). Citrus pests classification using an ensemble of deep learning models. *Comput. Electron. Agric.* 186, 106192. doi: 10.1016/j.compag.2021.106192

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. doi:10.48550/arXiv.1412.6980

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM* 60 (6), 84–90. doi: 10.1145/3065386

Li, X., Wang, W., Hu, X., and Yang, J. (2019). "Selective kernel networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Long Beach, CA, USA, 510–519. doi: 10.1109/CVPR.2019.00060

Liu, H., and Chahl, J. S. (2021). Proximal detecting invertebrate pests on crops using a deep residual convolutional neural network trained by virtual images. *Artif. Intell. Agric.* 5, 13–23. doi: 10.1016/j.aiia.2021.01.003

Liu, H., Lee, S.-H., and Chahl, J. S. (2016). A review of recent sensing technologies to detect invertebrates on crops. *Precis. Agric.* 18 (4), 635–666. doi: 10.1007/s11119-016-9473-6

Liu, J., and Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods* 17 (1), 1–18. doi: 10.1186/s13007-021-00722-9

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60 (2), 91–110. doi: 10.1023/B:VISI.0000029664.99615.94

Ma, N., Zhang, X., Liu, M., and Sun, J. (2021). "Activate or not: Learning customized activation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA, 8028–8038. doi: 10.1109/CVPR46437.2021.00794

Oliva, A., and Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. *Int. J. Comput. Vision* 42 (3), 145–175. doi: 10.1023/A:1011139631724

Rani, R. U., and Amsini, P. (2016). Pest identification in leaf images using SVM classifier. *Int. J. Comput. Intell. Inf.* 6 (1), 248–260. doi: 10.13140/RG.2.2.11632.30721

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). "MobileNetV2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 4510–4520. doi: 10.1109/cvpr.2018.00474

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

Tetila, E. C., Machado, B. B., Astolfi, G., Belete, N. A., de, S., Amorim, W. P., et al. (2020). Detection and classification of soybean pests using deep learning with UAV images. *Comput. Electron. Agric.* 179, 105836. doi: 10.1016/j.compag.2020.105836

Wu, X., Zhan, C., Lai, Y.-K., Cheng, M.-M., and Yang, J. (2019). "IP102: A Large-scale benchmark dataset for insect pest recognition," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 8779–8788. doi: 10.1109/cvpr.2019.00899

Xiao, D., Feng, J., Lin, T., Pang, C., and Ye, Y. (2018). Classification and recognition scheme for vegetable pests based on the BOF-SVM model. *Int. J. Agric. Biol. Eng.* 11 (3), 190–196. doi: 10.25165/j.ijabe.20181103.3477

TYPE Original Research
PUBLISHED 25 April 2023
DOI 10.3389/fpls.2023.1162435

Mobile robotics platform for strawberry temporal–spatial yield monitoring within precision indoor farming systems

Guoqiang Ren[1,2,3], Hangyu Wu[4], Anbo Bao[5], Tao Lin[1,3], Kuan-Chong Ting[1,2,6] and Yibin Ying[1,3]*

[1]College of Biosystems Engineering and Food Science, Zhejiang University, Hangzhou, Zhejiang, China, [2]Zhejiang University-University of Illinois Urbana-Champaign Institute (ZJU-UIUC), International Campus, Zhejiang University, Haining, Zhejiang, China, [3]Key Laboratory of Intelligent Equipment and Robotics for Agriculture of Zhejiang Province, Hangzhou, China, [4]College of Control Science and Engineering, Zhejiang University, Hangzhou, Zhejiang, China, [5]Department of Automation, Shanghai Jiao Tong University, Shanghai, China, [6]Department of Agricultural and Biological Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, United States

OPEN ACCESS

EDITED BY
Longsheng Fu,
Northwest A&F University, China

REVIEWED BY
Abbas Atefi,
California Polytechnic State University, United States
Zichen Huang,
Kyoto University, Japan

*CORRESPONDENCE
Yibin Ying
✉ybying@zju.edu.cn

SPECIALTY SECTION
This article was submitted to Technical Advances in Plant Science, a section of the journal Frontiers in Plant Science

RECEIVED 09 February 2023
ACCEPTED 21 March 2023
PUBLISHED 25 April 2023

CITATION
Ren G, Wu H, Bao A, Lin T, Ting K-C and Ying Y (2023) Mobile robotics platform for strawberry temporal–spatial yield monitoring within precision indoor farming systems.
Front. Plant Sci. 14:1162435.
doi: 10.3389/fpls.2023.1162435

Plant phenotyping and production management are emerging fields to facilitate Genetics, Environment, & Management (GEM) research and provide production guidance. Precision indoor farming systems (PIFS), vertical farms with artificial light (aka plant factories) in particular, have long been suitable production scenes due to the advantages of efficient land utilization and year-round cultivation. In this study, a mobile robotics platform (MRP) within a commercial plant factory has been developed to dynamically understand plant growth and provide data support for growth model construction and production management by periodical monitoring of individual strawberry plants and fruit. Yield monitoring, where yield = the total number of ripe strawberry fruit detected, is a critical task to provide information on plant phenotyping. The MRP consists of an autonomous mobile robot (AMR) and a multilayer perception robot (MPR), i.e., MRP = the MPR installed on top of the AMR. The AMR is capable of traveling along the aisles between plant growing rows. The MPR consists of a data acquisition module that can be raised to the height of any plant growing tier of each row by a lifting module. Adding AprilTag observations (captured by a monocular camera) into the inertial navigation system to form an ATI navigation system has enhanced the MRP navigation within the repetitive and narrow physical structure of a plant factory to capture and correlate the growth and position information of each individual strawberry plant. The MRP performed robustly at various traveling speeds with a positioning accuracy of 13.0 mm. The temporal–spatial yield monitoring within a whole plant factory can be achieved to guide farmers to harvest strawberries on schedule through the MRP's periodical inspection. The yield monitoring performance was found to have an error rate of 6.26% when the plants were inspected at a constant MRP traveling speed of 0.2 m/s. The MRP's functions are expected to be transferable and expandable to other crop production monitoring and cultural tasks.

KEYWORDS

mobile robotics platform, indoor vertical farming systems, GPS-denied navigation, temporal–spatial data collection, yield monitoring

1 Introduction

Strawberries (Fragaria × ananassa) are favored by consumers due to their rich nutrition and distinctive flavor. Precision indoor farming systems (PIFS), vertical farms with artificial light (aka plant factories) in particular, have long been suitable plant production scenes due to the advantages of efficient land utilization and year-round cultivation. In recent years, some companies, including Bowery Farming, Oishii Farm, and 4D Bios, successfully cultivated strawberries in plant factories. Farmers and researchers need to understand how plants grow and provide what plants need to increase fruit yield and quality. Plant phenotyping, an emerging science that describes the formation process of the functional plant body (phenotype) under the influence of dynamic interaction between the genotypic differences (genotype) and the corresponding environmental conditions (Walter et al., 2015), can provide valuable information for crop genetic selection and production management. People usually go to fields or laboratories to manually obtain plant phenotypic data. Such practices are highly labor-intensive, time-consuming, non-robust, and sometimes destructive and, therefore, may be limited by experimental scale, collection accuracy, and human subjective differences (Bao et al., 2019). A field-based, large-scale, and high-throughput plant phenotyping approach to overcome the bottleneck of manual operation is urgently needed (Araus et al., 2018).

Internet of Things (IoT) devices, which focus on collecting environmental data, are prevalent within PIFS as the monitoring system. Experience-oriented growth regulation decision-making can be built using environmental data by production managers. However, the decision-making process based on experience is indirect and delayed. The plant phenotypic data should be added to form a closed-loop decision-making pipeline. Considering fine-grained data collection is positively correlated with the number of camera sensors, the coverage and accuracy of data acquired by traditional IoT systems cannot be readily achieved within reasonable budgets. Mobile robots equipped with multiple sensors (the concept of quasi-IoT) present a great potential to acquire desired phenotyping data automatically. In the past few years, reported examples of phenotyping robots, emphasizing mobility-enabled field trials, have been increasing (Mueller-Sim et al., 2017; Shafiekhani et al., 2017; Higuti et al., 2019). However, there has been limited published work on mobile robots that have the capability of autonomously capturing phenotypic data within PIFS. We aimed to develop a mobile robotics platform (MRP) with the capabilities of periodical monitoring of individual strawberry plants and fruit within the entirety of a commercial plant factory. Fine-grained plant growth data captured by the MRP can provide production guidance and facilitate integrated GEM research.

An MRP applied in agricultural scenarios should have two primary capabilities: providing navigation for multiple-location data acquisition and data-driven decision support. Navigation in indoor scenarios is challenging due to the lack of GPS. As an alternative approach to GPS used in indoor scenarios, ultra-wideband (UWB) is high-precision but high-cost (Flueratoru et al., 2022). The stability of the navigation is closely related to the strength of signals that suffer from occlusion and attenuation errors under plant growing structures. Furthermore, UWB provides relatively static information that cannot detect unexpected obstacles. Light Detection and Ranging (LiDAR) sensors have been widely used in agricultural navigation that can actively acquire accurate depth information with an extensive detection range and a low sensitivity to lighting changes compared to other sensors (Debeunne and Vivet, 2020). A random sample consensus (RANSAC) algorithm was applied to discern maize rows fast and robustly while navigating in a well-structured greenhouse (Reiser et al., 2016). However, in complex environments like plant factories with repetitive shelves and narrow aisles, LiDAR can only obtain a limited number of signals representing the presence of objects. There is no semantic information for effectively completing the scene restoration. In contrast, visual navigation is limited by the low accuracy in depth estimation and the weak robustness against lighting changes (Zhang et al., 2012). A robot cannot safely and robustly navigate within plant factories using only one sensor as the single perception source. Multi-sensor fusion approaches, which can significantly improve the fault tolerance of a system while increasing the system's redundancy to increase the accuracy of object localization, have been proven to show great potential to solve navigation problems in complex scenes like urban traffic (Urmson et al., 2008). In consideration of a GPS-denied environment like PIFS, simultaneous localization and mapping (SLAM) technology can be a feasible navigation approach (Chen et al., 2020). The state-of-the-art LiDAR-SLAM Cartographer (Hess et al., 2016) and visual–inertial system (VINS) (Qin et al., 2018) are all open-source tools in the ROS (Robot Operating System) community. These algorithms, which can be easily implemented on a mobile robot, can potentially address navigation challenges. However, SLAM has some limitations, such as computational cost and lack of feature extraction ability; therefore, it is not directly applicable to this research. In this study, we report our research on a novel approach of fusing wheel odometry, inertial measurement unit (IMU), and AprilTag observations (captured by a monocular camera) to achieve accurate navigation within repetitive and narrow passages of PIFS.

Providing data-driven decision support based on the plant growth information is the other critical capability of the MRP. There exist some common decision-making pipelines in both academia and industry, including ripeness detection (Talha et al., 2021), diseases and pest identification (Lee et al., 2022), and fruit counting (Kirk et al., 2021). Image data captured by various perception systems have been widely used to achieve the above purpose (Gongal et al., 2015). In recent years, AlexNet brought about a renewed understanding of deep CNN and evolved into the foundation of contemporary computer vision (Krizhevsky et al., 2012). The powerful end-to-end learning makes the decisions possible, especially in the detection-based task from static images (Zhou et al., 2020; Perez-Borrero et al., 2021). The computing power of MRP limits the development of efficient CNN architectures as the neural network deepens (Zhang et al., 2018). Both occlusions from neighboring fruit and foliage and illumination changes could cause variations in fruit appearance (Chen et al., 2017). Compared to tasks, like ripeness and disease detection, counting from videos is challenging due to bias in fruit localization and tracking errors originating from occlusions and illumination changes (Liu et al., 2018b). Some traditional algorithms, including Optical Flow, Hungarian algorithm, and Kalman Filters, were used to track multiple fruits among sequential video frames. Liu et al. combined fruit segmentation and Structure from Motion (SfM) pipelines for counting apples and oranges grown on trees. The extra introduction of relative size distribution estimation and 3D localization could eliminate parts of double-counted fruits to further enhance the counting accuracy. Strawberry fruit is of small sizes and has complex ripe stages and dense growth scenes, which bring real challenges to the detection and tracking process.

This paper reports the current state of development and testing of the MRP's abilities of periodical monitoring of individual strawberry plants and fruit within a commercial plant factory. The challenges of navigation within narrow and repetitive indoor environments for temporal–spatial plant data acquisition and accurate yield monitoring for production management and harvesting scheduling in the MRP's periodical inspection operations need to be taken into consideration. In summary, the objectives of our research are as follows:

1. To develop the software and hardware of an MRP, consisting of an autonomous mobile robot (AMR) and a multilayer perception robot (MPR), which can capture temporal–spatial phenotypic data within a whole strawberry factory.
2. To achieve accurate navigation within the repetitive and narrow structural environments of a PIFS through an AprilTag and inertial navigation (ATI navigation) algorithm.
3. To evaluate the performance of strawberry yield monitoring through a novel pipeline that combines keyframes extraction, fruit detection, and postprocessing technologies.

2 Mobile robotics platform

In this study, an MRP to operate within a PIFS with multiple plant growing tiers has been developed to dynamically monitor plant growth and provide data for supporting crop growth model construction and production management. The modularly designed MRP (Figure 1) consists of an AMR, i.e., the mobile base, and an MPR, i.e., the lifting module + perception module, where MRP = MPR installed on top of AMR. The AMR is capable of traveling along the aisles between plant growing rows (i.e., *x* direction) with high positioning accuracy (PA) and robust navigation capability. The MPR has a perception module (for data acquisition) that can be raised by a lifting module to reach the heights (*z* direction) of all plant growing tiers of every row within the PIFS. The assembly of the AMR and MPR can perform automatic acquisition, storage, and transmission of phenotypic data of all individual plants within the entirety of a plant factory. Furthermore, multiple fault detection measures were designed and installed in the MRP. The MRP has been operating in a commercial strawberry production plant factory since July 2022, and has been working as expected so far.

The AMR is a differential drive mobile robot with two 165-mm hub motors, which has the ability to turn on the spot. The cylinder shape mobile base has a diameter of 500 mm and a height of 240 mm, which can travel at a maximum speed of 1.5 m/s through an aisle (with a minimum width of 600 mm) within a plant factory. An Intel® Core™ i5-8265U/1.6 GHz industrial computer is mounted inside the robot to run all navigation, data acquisition, and data transmission programs. The speed control commands from the industrial computer can be received by a low-level control board to drive the AMR to move. Wheel encoders, an IMU (US$40) mounted inside the mobile base, and a downward viewing monocular camera (US$25) to detect AprilTags on the floor are integrated to realize accurate localizations within PIFS, and a 2D LiDAR is used to detect obstacles. An emergency button is directly connected to the low-level control board to stop the motors when necessary.

The MPR is for use to perform data acquisition. The perception module of the MPR is an Intel® RealSense™ D435i depth camera (Intel Corporation, California, USA) mounted on a servo motor that provides the camera with the pitch motion to capture multiple images from various camera angles. The perception module can be raised to 2.8 m, the height of the top tier of each plant growing row, by the lifting module. The phenotypic data of each plant within a strawberry PIFS can be collected by the MRP's periodical inspection of the entire facility. Data of all plants on one of the five tiers were collected on one inspection route. The data of plants and the MRP's motion can be recorded in the rosbag format at a unified timestamp, which facilitates the data analysis and decision support processes. During the experiments on data acquisition, the MRP traveled at the speeds of 0.2, 0.3, and 0.4 m/s along the aisle between plant growing rows. The distance between the center of the MRP and the sides of the plant growing rows was kept at approximately 410 mm.

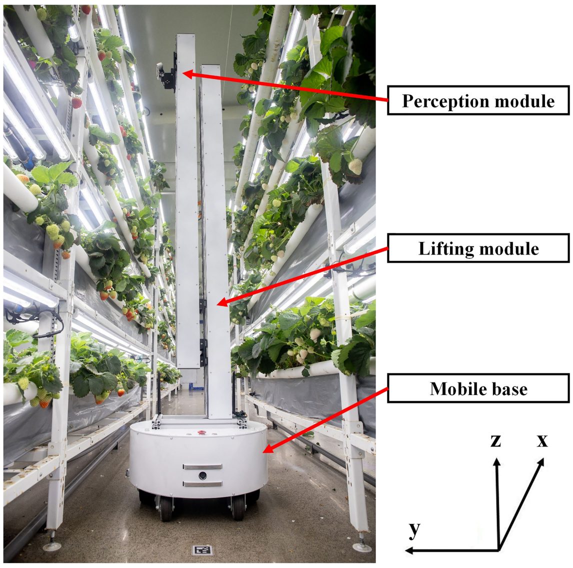


FIGURE 1
Hardware of the mobile robotics platform (MRP).

The resolution of the RealSense camera was set to 1,280 × 720 at 30 frames per second (FPS). The camera was set to be parallel to the side of a plant growing row by a servo motor and at the same height as the fruit by the lifting module. The same procedure was conducted to ensure the success of data acquisition on each tier.

3 Methods

This section presents two basic capabilities of the MRP: navigation for multiple-location data acquisition and strawberry yield monitoring.

3.1 Navigation

The navigation system installed in the AMR included navigation sensors, an industrial computer, and a low-level control system (Figure 2). The ROS was implemented in the industrial computer to collect data and conduct the navigation pipeline. There were five ROS nodes in the navigation pipeline, including an obstacle detection node, a localization node, a navigation node, a state machine node, and a low-level communication node. The real-time poses (position and heading) of MRP were calculated from the camera, IMU, and wheel encoders, through the localization node. The poses were received by the navigation node to conduct the global path planning and local path tracking, which, in turn, generated the target angular velocity and linear velocity of the MRP at a frequency of 50 Hz. The obstacle information captured by a 2D LiDAR from the obstacle detection node and the localization state (success or failure) from the localization node were sent to the state machine node. The updated state of the system from the state machine node and the target velocity from the navigation node were transmitted to the low-level communication node, which then calculated the target speed of the two motors and sent them to the low-level control board through serial communication.

An ATI navigation algorithm was developed to address the challenges of accurate navigation within the repetitive and narrow structural environments of a PIFS. The ATI navigation algorithm consists of four parts: mapping, localization, planning, and control.

The purpose of mapping in this study was to chart the moving route of MRP. The research was carried out at a commercial strawberry factory (4D Bios Inc., Hangzhou, China). A total of 45 AprilTags (Olson, 2011) of 40 × 40 mm in size were pasted on the grounds of both sides of each plant growing row. An 875 Prolaser® (KAPRO TOOLS LTD., Jiangsu Province, China) was used to ensure that all tags were on a designated straight line. The distance between the two neighboring tags was approximately 1.3 m. When collecting data for developing the ground map of a production facility, the MRP was first moved to Tag 0, which is the location of the charging pile (Figure 3). The MRP was controlled by a joystick to pass above the tags in order while simultaneously recording the data of the monocular camera, IMU, and wheel encoders. The mapping dataset was built after MRP had traveled along all the tags and returned to Tag 0.

The tag ID and the homogeneous transform of the tag relative to the monocular camera mounted on the MRP were both calculated by the AprilTag detection algorithm (Wang and Olson, 2016). The wheel encoders and IMU were fused to calculate the trajectory of the MRP using Equation 1.

$$\begin{cases} \theta_{k+1} = \theta_k + \Delta\theta_{imu} \\ x_{k+1} = x_k + (\Delta s_l + \Delta s_r)\cos(\theta_k)/2 \\ y_{k+1} = y_k + (\Delta s_l + \Delta s_r)\sin(\theta_k)/2 \end{cases} \tag{1}$$

where $\Delta\theta_{imu}$ is the heading variation of IMU between timestamps of k and $k+1$. Δs_l and Δs_r represent the motions of the left and right wheel obtained by optical encoder during two timestamps, respectively.

The tag IDs were further used to conduct the loop closing optimization through the pose graph optimization (PGO) algorithm. The vertices were represented by processed global poses of the tags, and the edges were denoted by relative pose changes of the odometer while MRP accessed two neighboring tags. We cast this as a nonlinear least squares problem

$$\arg\min_x \frac{1}{2}\sum_{ij} e_{ij}^T \Omega_{ij} e_{ij}$$

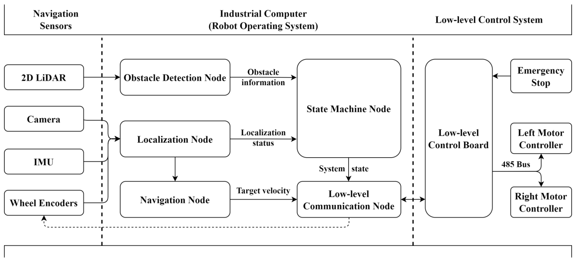


FIGURE 2
Navigation system architecture.

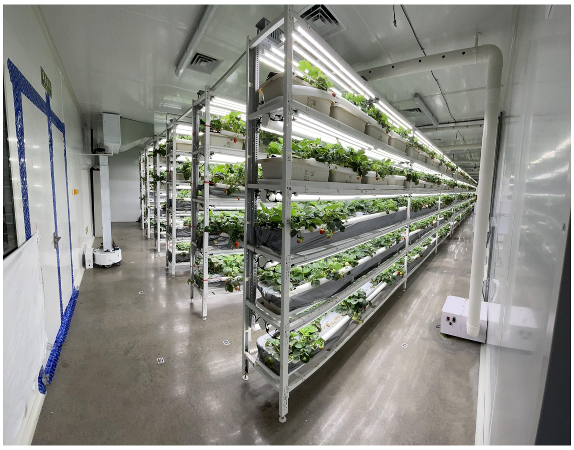


FIGURE 3
The MRP is being charged in the commercial strawberry factory.

where the state of the tag is denoted by a 2D coordinate vector and a heading angle, $x = \{p, \theta\}$. The information matrix Ω_{ij} is used to assign weights to different errors. The error e_{ij} between the expected observation and the real observation from Tag i and Tag j, can be calculated by Equation 2.

$$e_{ij} = \begin{pmatrix} R_i^T(p_j - p_i) - \hat{p}_{ij} \\ \theta_j - \theta_i - \hat{\theta}_{ij} \end{pmatrix} \tag{2}$$

R_i is the rotation matrix corresponding to the heading angle in x_i. $\hat{p}_{ij}$ and $\hat{\theta}_{ij}$ represent the relative pose changes of edges. Levenberg–Marquardt (L-M) algorithm was used to optimize the poses of all tags and generate the map. The accurate poses of the tags could be obtained in the process of mapping.

Based on whether one of the AprilTags was detected at the current timestamp, the estimations of localization could be divided into two situations. When the tag was correctly detected by the monocular camera, the global pose of the MRP at this timestamp could be calculated by the global pose of the tag in the existing map and the pose transform of the tag relative to the MRP. Otherwise, the detection result of the last tag in the existing map and the odometry changes from the timestamp when the last tag was detected to the current timestamp were used to estimate the global pose of the MRP.

In path planning, based on the destination, on the mapped route, entered by a human operator, a trajectory composed of a sequential set of locations could be generated by MRP's global path planner as the waypoints. Based on whether the destination is a tagged position, global path planning can be divided into two cases. If the destination is the position of one of the tags on the undirected map, the shortest path can be obtained through the breadth-first search (BFS) algorithm. If not, a virtual tag representing the destination will be temporarily inserted between two adjacent tags on the undirected map. The optimal path could be calculated by the BFS algorithm performed on the newly constructed undirected map.

After obtaining the global path, the MRP can be navigated through a series of local paths at the angular and linear velocities issued by the low-level control board (Figure 2). For a straight global path consisting of more than or equal to three tags, the local path target position is set to Tag_{i+2} with MRP passing Tag_i, which will keep the velocity of the MRP along the planned route stable. Angular velocity is calculated by the anti-windup pi controller to adjust the heading toward the target position. The linear velocity is calculated by a proportional controller to prevent system overshoot. The target speed of the left and right motors will be further obtained according to the differential motion model.

3.2 Yield monitoring

The growth condition of strawberries on each tier of the plant growing rows could be recorded in a video format after the inspection by the MRP. In this study, we have developed a strawberry yield monitoring method. The counting-from-video method consisted of two phases: detection and counting of ripe fruit.

3.2.1 Fruit detection

Ripeness detection is the first step in the yield monitoring pipeline. Considering that the detection task has high requirements for speed and accuracy, the single-stage detector YOLOv5 is chosen to detect the ripe strawberry (Jocher et al., 2022). The framework of the detector can be divided into four parts: Input with mosaic data augmentation, CSPDarknet53 (Bochkovskiy et al., 2020) as Backbone, Neck applying Feature Pyramid Network (FPN) (Lin et al., 2017) and Path Aggregation Network (PAN) (Liu et al., 2018a), and Prediction using GIoU loss (Rezatofighi et al., 2019). The framework extracts and aggregates semantically and spatially strong features more efficiently. More efficient representation improves the performance of multi-scale object recognition. Various variants have been generated by adjusting the depth and width of the network. YOLOv5l6 was used in this research, with an inference time of 15.1 ms running on an NVIDIA® V100 Tensor Core GPU.

3.2.2 Fruit counting

A fruit counting pipeline was presented to count ripe strawberries on video, including keyframe extraction, fruit detection, and postprocessing (Figure 4).

3.2.2.1 Keyframe extraction

Considering that any individual strawberry fruit could appear in multiple frames of the video captured, the number of times a fruit might be counted was not fixed. Therefore, fruit detection results could not be directly accumulated to obtain the counting results. The concept of keyframe extraction was applied to fix the number of times of repetitive counting, r. The pixel distance of two neighboring keyframes in the pixel coordinate system, d_p, was calculated by Equation 3.

$$d_p = \frac{w}{r} \tag{3}$$

where w was the image width. All strawberries in the video were required to appear at least twice in all extracted frames; therefore, r was greater than or equal to 2. Figure 5 shows example series of keyframes at various values of r.

The pixel distance between keyframes was converted to the movement of fruit in the camera coordinate system to further

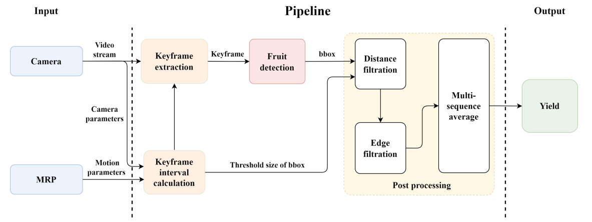


FIGURE 4
The overall yield monitoring pipeline.

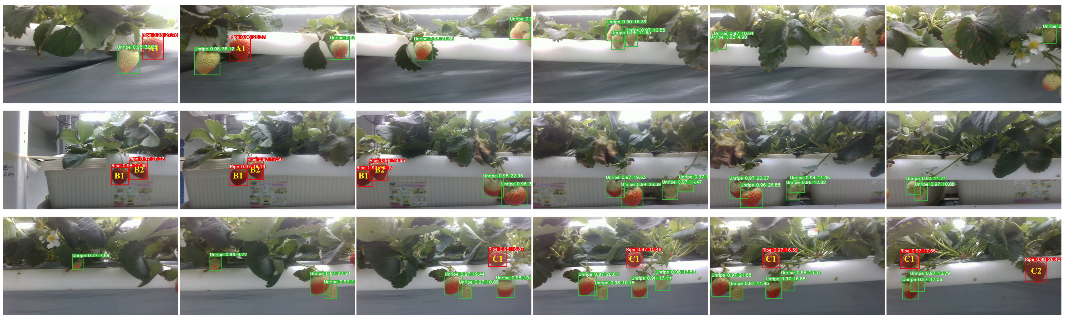


FIGURE 5
Example series of keyframes at various values of *r*: *r* = 2 in the upper row, *r* = 3 in the middle row, and *r* = 4 in the bottom row.

calculate the interval between keyframes in the video. The theoretical interval of keyframes, i_t , could be calculated by Equation 4.

$$i_t = \frac{d_p \times d \times fps}{f_x \times v} \tag{4}$$

where *fps* is the frame rate of the video. f_x denotes the intrinsic parameters of the RealSense camera, *v* represents the traveling speed of MRP, and *d* stands for the average distance between the camera and the fruit. Equation 5 was used to calculate the nearest integer of i_t to obtain the actual interval of keyframes, *i*.

$$i = int(i_t) = int(\frac{w \times d \times fps}{f_x \times v \times r}) \tag{5}$$

where the variable *d* was assumed to be a constant in this study. *i* is only related to values of *v* and *r*, where $i = g(v \times r)$. The counting-from-video problem was transformed into the statistics of fruit detection results of keyframes.

3.2.2.2 Postprocessing

Postprocessing approaches were integrated to further improve the counting accuracy, including distance filtration, edge filtration, and multi-sequence average. Strawberries on other plant growing rows might enter the camera's field of view during the MRP inspection process. The distance filtration approach based on the bounding box (bbox) size of the detection results was developed to eliminate the interference to counting by the strawberries located outside experimental areas. An edge filtration approach was used to prevent partially visible strawberries at the edge of the image from being counted repeatedly. Only the strawberries that appeared on the left edge were counted, and the strawberries that appeared on the right edge were ignored. Figure 6 shows the two situations described above.

There existed errors in frame extraction between the actual interval of keyframes *i* and the theoretical interval of keyframes i_t, $e = |i - i_t|$. A multi-sequence averaging algorithm was developed to reduce the counting errors caused by the errors that occurred in the keyframe extraction process. The yield monitoring algorithm was presented as Algorithm 1:

```
Input: Threshold of keyframe interval $i^s$, Threshold of errors of frame extraction $e^s$, Threshold of the number of repetitive counting $r^s$, MRP traveling speed $v$, Inspection
```

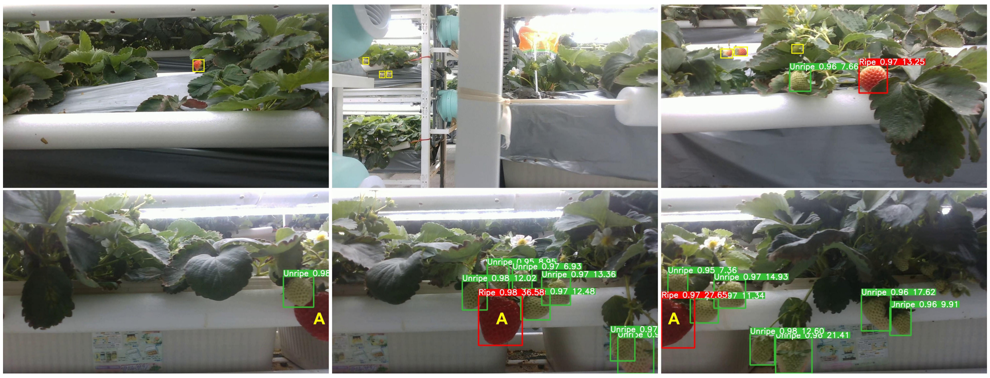


FIGURE 6
Upper row: Three example cases that needed to be processed by distance filtration. Strawberries annotated with yellow bboxes were not in the experimental areas and were not counted. Bottom row: Edge filtration was applied to process three consecutive keyframes (*r* = 2). The ripe fruit A was not counted since it was partly visible on the right edge of the left image. Fruit A was counted after it had moved to the left edge of the right image.

```
video $\mathbf{V}$
Output: The number of ripe fruits in the video $\mathbf{V}$ , $n$
Initialize $i^s$ =4, $e^s$ =0.1, $r^s$=15
Initialize $\mathbf{R} = \{r_j\}_{j=2}^{r^s} = \{2, 3, \ldots, r^s\}$
$\mathbf{I}_t := \{i_t j | i_t j := g(r_j \times v), r_j \in \mathbf{R}\}_{j:=2}^{r^s}$
                                        // Calculated by Eq. (4)
$\mathbf{I} := \{i_j | i_j := (i_{tj}), i_{tj} \in \mathbf{I}_t\}_{j:=2}^{r^s}$
                                        // Calculated by Eq. (5)
$\mathbf{E} := \{e_j | e_j := |i_{tj} - i_j|, i_{tj} \in \mathbf{I}_t, i_j \in \mathbf{I}\}_{j:=2}^{r^s}$
$\mathbf{R}^e := \{r_j | r_j \in \mathbf{R}, e_j > e_s, e_j \in \mathbf{E}\}_{j:=2}^{r^s}$
$\mathbf{R}^i := \{r_j | r_j \in \mathbf{R}, i_j < i_s, i_j \in \mathbf{I}\}_{j:=2}^{r^s}$
$\mathbf{R}^s \ \mathbf{R} - \mathbf{R}^e \cap \mathbf{R}^i$
$\mathbf{E}^s := \{e_j | e_j \in \mathbf{E}, r_j \in \mathbf{R}\}_{j:=2}^{r^s}$
$\tilde{\mathbb{E}}^s := \mathcal{S}ort(\mathbf{E}^s)$  // $\mathcal{S}ort$: To sort $\mathbf{E}^s$ to get an ascending-order array $\tilde{\mathbb{E}}^s$
if $\tilde{\mathbb{E}}^s[2] > e^s$ then
    $\mathbf{E}^c := \{\tilde{\mathbb{E}}^s[0], \tilde{\mathbb{E}}^s[1]\}$
else
    $\mathbf{E}^c := \{\tilde{\mathbb{E}}^s[0], \tilde{\mathbb{E}}^s[1], \tilde{\mathbb{E}}^s[2]\}$
$\mathbf{R}^c := \{r_j | r_j \in \mathbf{R}^s, e_j \in E^c\}_{j:=2}^{r^s}$
$\mathbf{I}^c := \{i_j | i_j \in \mathbf{I}, r_j \in \mathbf{R}^c\}_{j:=2}^{r^s}$ // $\mathbf{R}^C$: The group of filtered intervals of keyframes
$\mathbf{S} := \{s_{r_j} | s_{r_j} := \mathcal{E}(\mathbf{V}, i_j), r_j \in \mathbf{R}^c, i_j \in \mathbf{I}^c\}$ // $\mathcal{E}$: To extract keyframes from V at interval $i_j$
$\mathbf{S}^F := \{s_r^F | s_r^F := \mathcal{F}(s_r), s_r \in \mathbf{S}, r \in \mathbf{R}^c\}$ // $\mathcal{F}$: To apply distance and edge filtration
$\mathbf{N} := \{n_r | n_r := \frac{\mathcal{C}(s_r)}{r}, s_r \in \mathbf{S}^F, r \in \mathbf{R}^c\}$// $\mathcal{C}$: To count the ripe fruit in $s_r$
$n := \mathcal{A}verage(\mathbf{N})$ // $\mathcal{A}verage$: To average all the sequence results in $\mathbf{N}$.
```

ALGORITHM 1
Yield monitoring.

4 Procedure of experiments

In this study, experiments were carried out at a commercial strawberry plant factory (Figure 2) in December 2022. Fragaria × ananassa Duch. cv. Yuexin plants bred by the Zhejiang Academy of Agricultural Sciences (Hangzhou, Zhejiang, China) were cultivated on four-tier planting structures. The experiments were conducted on a row of three four-tier planting structures near a wall. There were 12 planting pots in every tier of each planting structure, and five strawberry plants were grown in each planting pot. Experiments were carried out on a total of 720 strawberry plants (i.e., 5 plants/pot × 12 pots/tier × 4 tiers/planting structure × 3 planting structures = 720 plants). Figure 7 shows the floor layout of the research facility and the MRP inspection route.

4.1 Navigation capability

4.1.1 Mapping

The typical configuration of a plant factory is a corridor environment with repetitive and narrow planting structures, which brings significant challenges to the LiDAR-based SLAM algorithm in mapping operations. LIO-SAM, one of the advanced LiDAR-based SLAM algorithms, was implemented on the MRP to compare and prove the advantages of the proposed mapping algorithm. LIO-SAM is a real-time, tightly coupled Lidar-Inertial odometry with high odometry accuracy and good mapping quality (Shan et al., 2020). In order to satisfy the use of the LIO-SAM algorithm, a VLP-16 3D LiDAR scanner (Velodyne Lidar, California, USA) and a WitMotion HWT905 nine-axis attitude and heading reference system (AHRS) sensor (WitMotion, Shenzhen, China) were integrated within the MRP. The collection of the mapping dataset was conducted using the same approach mentioned in *Section 3.1*. The data of 3D LiDAR and nine-axis IMU were used in the LIO-SAM algorithm for pose estimation. The data of the monocular camera, IMU, and wheel encoders were used in the mapping algorithm of the ATI navigation system developed in this research. All optimization processes were conducted offline for the two algorithms. Another experiment was conducted to compare the mapping performances of the ATI navigation system, without and with loop closing optimization, to show the impact of optimization in this research. Mapping trajectories were used to evaluate the mapping performances of the three approaches.

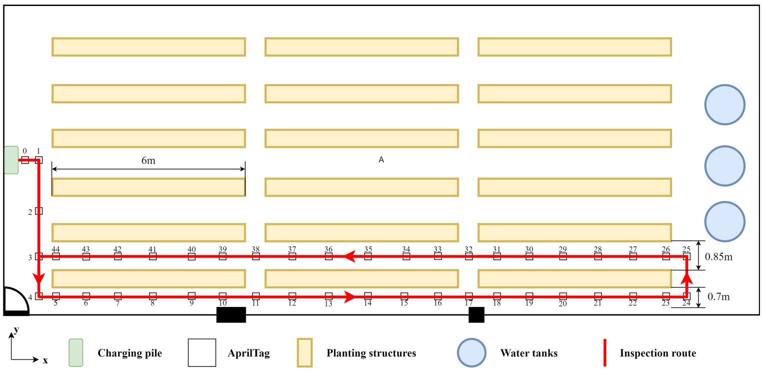


FIGURE 7
Schematic diagram of the experimental scene and inspection route.

4.1.2 Localization

This experiment aims to test the ability of MRP to move to a desired location as expected. PA was used to evaluate the navigation performance in this research. The coordinate system is shown in the lower left corner of Figure 7. The positive direction of the *X*-axis is consistent with the movement direction of the MRP when inspecting strawberry plants. In the autonomous navigation mode, three tags at different positions (Tags 8, 12, and 21) were selected for testing PA. The MRP started from Tag 5 and navigated to the target tag at the traveling speed of 0.4 m/s after entering the Tag ID. The current position of the tag in the image coordinate system was recorded to compare with the tag's position in the map generated by the ATI navigation algorithm. The same operations were repeated five times for each tag. Euclidean distance between two positions was represented as distance deviation, err_d. err_x represents the deviation in the *x* direction, and err_y represents the deviation in the *y* direction. The root mean squared error (RMSE) of five trails per tag was computed by Equation 6, and the RMSE of 15 trails of three tags was computed as PA.

$$RMSE = \sqrt{\frac{1}{k}\sum_{l=1}^{l} err\_d_l^2} \tag{6}$$

where *k* is the number of trails. err_d_l represents the err_d in trail *l*.

4.2 Fruit detection and counting

4.2.1 Fruit detection

A total of 80 videos were captured along the plant growing rows by farmers at a normal walking pace using an Intel® RealSense™ D435i depth camera and a smartphone, under various illumination conditions, different strawberry growth scenes, and various strawberry growth stages (from March to July 2021). The dataset consisted of 1,600 frames that were extracted out of every 10 frames from the videos, with the images without strawberries manually removed. All strawberry fruits in the period of veraison were annotated by growers. Of those, every fruit having an 80% or more red area on its surface was annotated as a ripe fruit (Hayashi et al., 2010). Other fruits were annotated as unripe ones. The dataset, including 2,327 ripe strawberries and 2,492 unripe strawberries, was randomly divided into train, validation, and test sets at the ratio of 8:1:1.

The strawberry ripeness detection model, YOLOv5l6, was implemented using the PyTorch framework. The modeling process was performed on a Linux workstation (Ubuntu 16.04 LTS) with two Intel Xeon E5-2683 Processors (2.1G/16 Core/ 40M), 128 GB of RAM, and four NVIDIA GeForce GTX 1080Ti graphics cards (11 GB of RAM). Taking a mini-batch size of 16, the SGD optimizer was adopted with a decay of 0.0001 and a momentum of 0.937. The best performance was achieved under the initial learning rate of 0.01. The number of warmup epochs and total training epochs were set to 3 and 90, respectively. The best model weight was chosen according to the value of mean average precision (mAP) (Everingham et al., 2010) calculated on the validation set. The chosen model was evaluated on the test set by mAP@0.5 (at the IoU threshold of 0.5).

4.2.2 Fruit counting

False detections and missed detections of fruit in a particular frame cannot be corrected by any other frames. Therefore, in this study, a counting algorithm was developed to count every fruit multiple times (a predetermined number of times that is equal to or greater than 2) in order to improve the accuracy of the fruit counting. The performance of the proposed algorithm was affected by *r*, *i*, and *e*. As mentioned in *Section 3.2*, *i* and *e* were related to the value of *r*. In this experiment, various values of *r* were tested to build the fruit counting algorithm with a robust performance. The MRP traveled at the speed of 0.3 m/s along the aisle between plant growing rows to capture the phenotypic data of each plant in the experimental region. Both video data captured by the RealSense camera at the actual frame rate of 29.72 fps and data from navigation sensors were recorded in the rosbag format at a unified timestamp. The MRP inspected and recorded all the data twice for each tier of plant growing rows. A total of eight videos were collected in this experiment. Fruit detection was performed on the eight videos. The number of ripe strawberry fruit in the results produced by the detection algorithm, n_{GT}^C, was manually counted as the ground truth of the fruit counting algorithm to exclude the impact of the fruit detection algorithm and evaluate the performance of the fruit counting algorithm alone. The yield monitoring algorithm results, *n*, were then estimated using the proposed algorithm without multi-sequence averaging (one of the three postprocessing techniques mentioned in *Section 3.2.2*). The thresholds e^s and i^s, mentioned in Algorithm 1, can be determined by selecting a number of smaller relative error rates of fruit counting, err^C, calculated by Equation 7.

$$err^C = \frac{\left|n - n_{GT}^C\right|}{n_{GT}^C} \times 100\,\% \tag{7}$$

4.3 Inspection capability

In this experiment, the inspection capability of MRP was tested at various traveling speeds of 0.2, 0.3, and 0.4 m/s. The inspection capability was a system performance that included mobility for multiple-location data acquisition and monitoring of strawberry yield.

4.3.1 Motion control

The experiment in this study was conducted three times to test the motion control performance of MRP at three different traveling speeds. In the navigation mode, MRP was programmed to start from the first tag (Tag 5) and stop at the last tag (Tag 23) position in the aisle. The distance error, linear velocity, yaw error, and angular velocity of the MRP were recorded in the rosbag format with a frame rate of 50 Hz as the errors and outputs of the control system.

Motion stability and angular tracking accuracy were considered to evaluate the effectiveness of the proposed method.

4.3.2 Yield monitoring

The accuracy of the yield monitoring algorithm is a system performance to evaluate both fruit detection and counting processes. The variables r, i, and e corresponding to three different traveling speeds could be calculated by repeating the operations mentioned in *Section 4.2.2* in the same experimental area on different dates. This experiment was conducted three times to test the accuracy of the yield monitoring algorithm at three traveling speeds of MRP. For each experiment, MRP inspected and recorded all the data twice for one of the four tiers of the plant growing rows. A total of 24 videos were collected in this experiment. The number of ripe strawberries in the raw video, n_{GT}^{Y}, was determined by growers as the ground truth of the yield. The relative error rate of yield monitoring, err^{Y}, could be calculated by Equation 8.

$$err^{Y} = \frac{\left|n - n_{GT}^{Y}\right|}{n_{GT}^{Y}} \times 100\,\% \qquad (8)$$

5 Results and discussion

5.1 Navigation capability

5.1.1 Mapping

As shown in Figure 8, two continuous and smooth trajectories were obtained using our ATI mapping approach (a and b). The two trajectories almost coincided before Tag 27. The trajectory in Figure 8B was the non-optimized result, which the MRP was not able to return to the charging pile (origin) due to cumulative errors of the system. Figure 8A shows the mapping trajectory processed by the ATI mapping approach with the loop closing optimization that was accomplished by making the path defined by Tags 0, 1, 2, and 3 the beginning segment and the path defined by Tags 3, 2, 1, and 0 the ending segment of the trajectory. The beginning tags (numbers 0, 1, 2, and 3) were detected in a reversed order when MRP was on the way back to the starting point, Tag 0. The global PGO was successfully performed to eliminate the cumulative errors and obtain a consistent and undistorted trajectory during the mapping process. The mapping trajectory coincided with the AprilTags pasted on the ground in the experimental area (Figure 7).

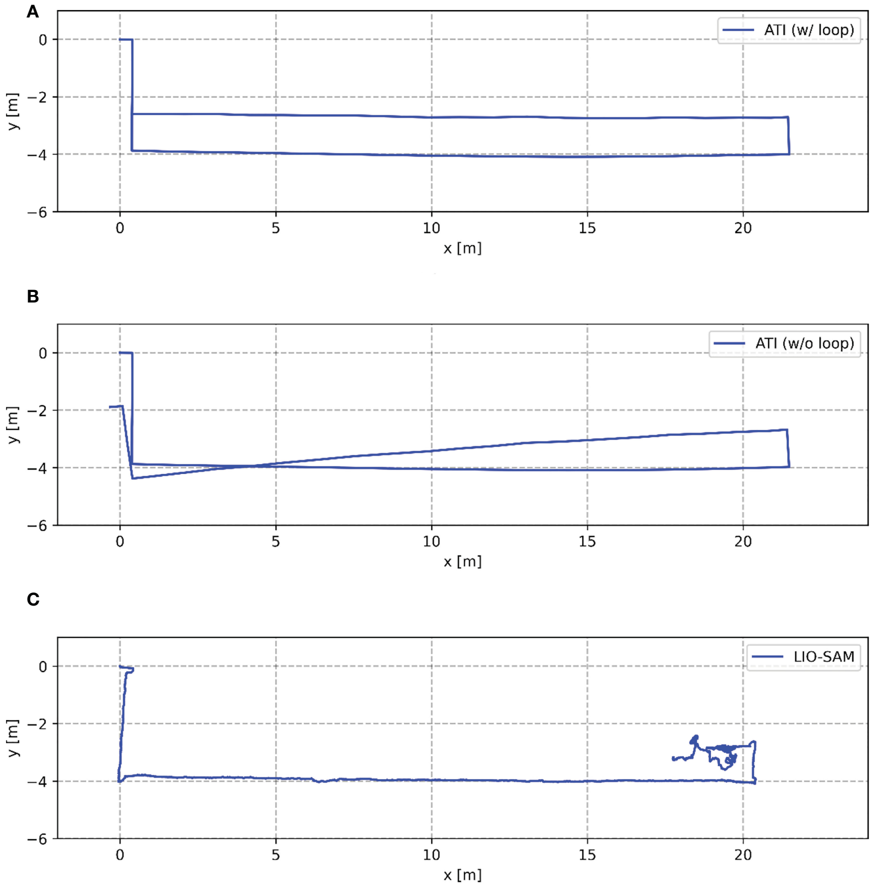


FIGURE 8
Comparison of trajectories obtained by the three mapping approaches running in the experimental area. **(A)** shows the mapping trajectory processed by the ATI mapping approach with the loop closing optimization. **(B)** shows the mapping trajectory processed by the ATI mapping approach without the loop closing optimization. **(C)** shows the mapping trajectory processed by the LIO-SAM algorithm with optimized parameters.

In contrast, a jittery mapping trajectory was obtained by LIO-SAM under the same movement of MRP (Figure 8C). Degeneracy occurred when MRP traveled back and turned to a new long aisle, i.e., starting from the position of Tag 25 in Figure 7. The estimated odometry oscillated around the same position. It is worth mentioning that the mapping results presented in Figure 8C were obtained by the LIO-SAM algorithm with optimized parameters. The original LIO-SAM failed at the second turn of the inspection route, i.e., starting from the position of Tag 4 in Figure 7. The experimental results show that the LiDAR-based SLAM algorithm failed in the environment of the plant factory. Our ATI navigation algorithm was effective and robust in the mapping process.

5.1.2 Localization

In the PA experiment, Tags 8, 12, and 21 were selected as target positions (Figure 7). Tests were repeated five times for each tag. The range of RMSE of each tag was found to be between 8.6 and 14.8 mm (Table 1). The overall RMSE of PA was 13.0 mm. Each tag could be effectively observed using the proposed ATI navigation algorithm, which showed the robustness of the positioning system. The positioning results of the algorithm in the x and y directions are all biased to the same side (Tables 2, 3). The external parameters among the wheel encoders, IMU, and monocular camera were estimated from the mechanical drawings with no calibration process in this research. The PA of the system could be further improved by automatic and accurate calibration of the navigation sensors and the optimization of fusion of wheel encoders and IMU.

5.2 Fruit counting capability

The best model weight was chosen according to the mAP@0.5 value of 0.994 for ripe strawberries calculated on the validation set. We have found that an mAP@0.5 value of 0.945 could be obtained on the test set. Strawberry growth scenes with occlusions could be identified accurately by the fruit detection model.

We have found that there was little change in i_t and i when the value of r was more than 15 and the value of v was 0.2, 0.3, or 0.4 m/s. The value of r was set from 2 to 15, and the value of v was 0.3 m/s in this experiment. The corresponding i and e values and the relative error rate of fruit counting, err^C, were computed and are shown in Table 4 in ascending order according to e values. The value of err^C generally increased as the increase of e. When the value of e was more than 0.1, the err^C was relatively large and fluctuated. When the value of i was relatively small, the impact of e on err^C was more obvious. The value of i^sas set as 4 through the observation of the experimental results. In this experiment, the values of r were chosen as 15, 10, and 6. The final err^C was computed as 3.3%. There also existed several limitations. We assumed that the value of d was constant. However, the variance in the distance between strawberries and the RealSense camera existed in the production scene, which affected the accuracy of the algorithm. The problem could be addressed by dynamically introducing accurate values of d captured by the depth camera into the algorithm. When v is high, the overlaps of two neighboring frames will be fewer. This will, in turn, limit the range of r values and the tolerable error rate will become smaller.

5.3 Inspection capability

5.3.1 Motion control

The motion control system worked stably at the nominal MRP traveling speeds of 0.2, 0.3, and 0.4 m/s. The performance of the distance controller and heading controller at various speeds is shown in Figure 9. The inspection durations at the three set speeds are 113.6, 78.6, and 62.1 s, respectively. The overall average speeds are 0.189, 0.273, and 0.346 m/s, respectively.

On the left of the figure, the blue lines represented the distance between MRP and the target position in the local path planner (*Section 3.1*), Dis_{local}, during the navigation process. At the start, the value of Dis_{local} was approximately 2.4 m, which was the distance between Tag 5 and Tag 7. As the robot moved forward, the value of Dis_{local} decreased linearly. When the MRP reached Tag 6, the local target was updated to Tag 8. At this time, the value of Dis_{local} returned to approximately 2.4 m, which was the distance between Tag 6 and Tag 8. When the MRP reached Tag 22, the local target was no longer updated. The value of Dis_{local} faded to zero as the robot moved towards the global target, Tag 23. MRP accelerated from zero to a set traveling speed, maintained the speed during the inspection, and gradually decelerated until reaching the global target, Tag 23, without an overshoot. On the right of the figure, the red lines represented the heading from MRP to the target position in the local path planner, Yaw_{local}, during the navigation process. The value of Yaw_{local} was within 0.01 rad most of the time and occasionally rose to 0.03 rad due to the updates of the target positions in the local path planner, which had little effect on the phenotypic data acquisition. The control system ensured smooth and low-error motions at various traveling speeds of MRP for stable quality of video collection.

TABLE 1 Positioning accuracy of the ATI navigation algorithm.

Tag ID	*err_d* (mm)						RMSE (Tag)	RMSE (All)
	1	2	3	4	5	Avg		
8	9.6	10.5	8.5	7.2	6.9	8.5	8.6	13.0
12	17.2	16.8	12.6	12.5	14.0	14.6	14.8	
21	13.1	16.6	17.1	16.6	6.6	14.0	14.5	

err _ d, distance deviation is the Euclidean distance between the current position of the tag in the image coordinate system and the position of the tag in the map generated by the ATI navigation algorithm.

TABLE 2 Positioning accuracy in the *x* direction of the ATI navigation algorithm.

Tag ID	*err_x*(mm)						RMSE (Tag)	PA (*x*)
	1	2	3	4	5	Avg		
8	−9.1	−10.4	−8.3	−5.9	−5.1	−7.8	8.0	11.2
12	−16.5	−14.4	−10.1	−11.1	−11.5	−12.7	12.9	
21	−8.6	−15.2	−16.2	−11.7	5.2	−9.3	12.1	

TABLE 3 Positioning accuracy in the *y* direction of the ATI navigation algorithm.

Tag ID	*err_y* (mm)						RMSE (Tag)	PA (*y*)
	1	2	3	4	5	Avg		
8	−3.2	−0.8	−2.0	−4.1	−4.6	−2.9	3.2	6.5
12	−4.7	−8.7	−7.6	−5.8	−8.0	−7.0	7.1	
21	−9.9	−6.7	−5.4	−11.8	−4.0	−7.6	8.1	

5.3.2 Yield monitoring

The err^C and err^Y of 24 test videos (8 videos per MRP traveling speed) were calculated and shown in Table 5. We found that the system showed robust monitoring results at various MRP traveling speeds, of which err^C was between 2% and 3%, and err^Y was between 6% and 10%. The best yield estimation performance was found to have an error rate of 6.26% at the MRP traveling speed of 0.2 m/s. The four ties of plant growing row in the experimental area corresponded to the four strawberry growth densities. Our algorithm had high robustness when dealing with scenes with various fruit densities.

The same strawberry appeared differently in various frames due to the changes in shooting angles during the movement of MRP. An unripe strawberry might be detected as a ripe or unripe one from various angles due to the distribution of red color on the fruit, which made n_{GT}^Y smaller than n_{GT}^C. The proposed yield monitoring approach is a detection-based pipeline, in which false detections caused the higher err^Y. In order to meet the above challenges and

TABLE 4 The relative error rate of fruit counting under different algorithm setups.

Setup				Counting results of various videos								*Avg* err^C
r	i_t	*i*	*e*	1_1	1_2	2_1	2_2	3_1	3_2	4_1	4_2	
15	2.029	2	0.029	28	29	31	31	43	46	37	36	0.032
10	3.044	3	0.044	28	29	31	31	43	46	37	36	0.032
6	5.073	5	0.073	29	29	30	31	43	46	37	35	0.035
5	6.088	6	0.088	28	29	31	31	43	47	37	35	0.038
3	10.146	10	0.146	30	26	29	30	44	46	36	35	0.052
14	2.174	2	0.174	30	31	33	34	46	49	40	38	0.049
8	3.805	4	0.195	27	27	30	30	41	42	35	34	0.072
2	15.219	15	0.219	28	29	31	29	44	49	36	33	0.059
11	2.767	3	0.233	25	26	28	28	39	42	34	32	0.116
13	2.341	2	0.341	33	33	36	36	50	53	43	41	0.133
7	4.348	4	0.348	31	31	34	34	46	48	40	39	0.058
9	3.382	3	0.382	31	32	34	34	48	51	41	40	0.083
4	7.610	8	0.390	27	27	30	30	41	42	35	34	0.072
12	2.537	3	0.4635	23	24	26	26	36	38	31	30	0.185
n_{GT}^C				30	30	32	32	44	44	37	37	

1_1 and 1_2 are the first and second videos of strawberries grown on the first tier, respectively. $Avg\ \ err^C$ is the average relative error rate of fruit counting, n_{GT}^C is the number of ripe strawberry fruit in the detection results.

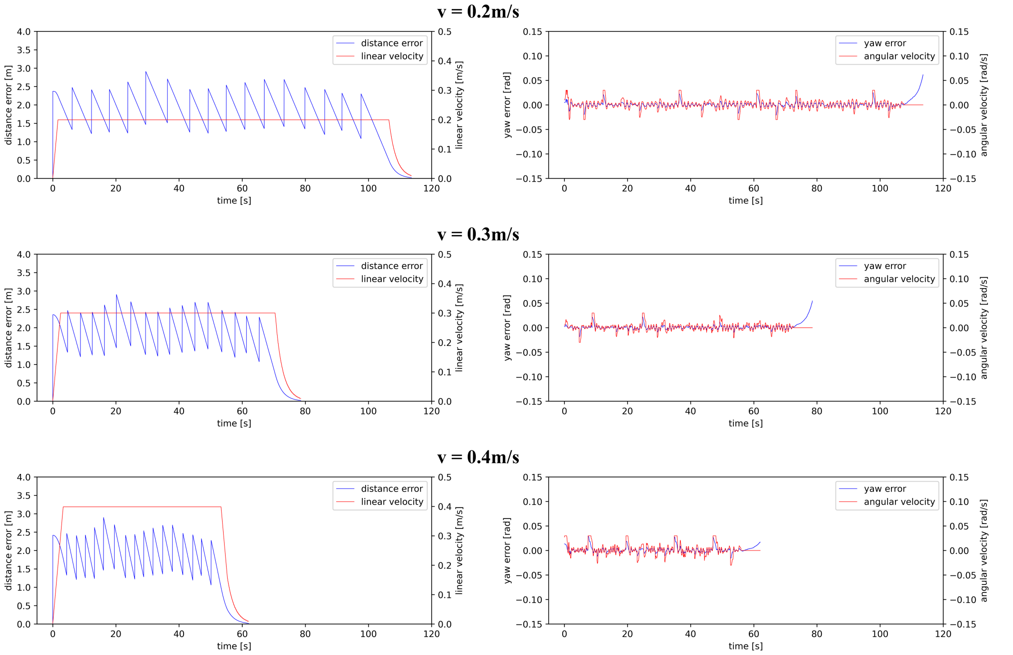


FIGURE 9
The performance of distance and heading controller at various MRP speeds.

obtain higher yield estimation accuracy, there exists a potential solution, which is to process with the original video data. Videos captured by the MRP could provide both spatial and temporal information for better tracking and detecting a single fruit. However, a large amount of needed computational time was the limitation of this solution.

6 Conclusion

In this study, we have developed software and hardware of an MRP, consisting of an AMR and an MPR, which can capture temporal–spatial phenotypic data within the whole strawberry factory. This paper reported two basic capabilities of the MRP, navigation for multiple-location data acquisition and strawberry yield monitoring. An ATI navigation algorithm was developed to address the challenges of accurate navigation within the repetitive and narrow structural environments of a plant factory. The MRP performed robustly at various traveling speeds tested with a PA of 13.0 mm. A counting-from-video yield monitoring method that incorporated keyframes extraction, fruit detection, and postprocessing technologies was presented to process the video data captured by MRP's inspection for production management and harvesting

TABLE 5 Yield monitoring performance comparison at various speeds of MRP.

Setup				Video ID	*n*				*Avg err^C*	*Avg err^Y*
v (m/s)	*r*	*i*	*e*		T1	T2	T3	T4		
0.2	15	3	0.044	1	34	52	87	70	0.0265	0.0626
	9	5	0.073	2	35	55	88	69		
0.3	15	2	0.029	1	37	54	88	71	0.0229	0.0905
	10	3	0.044	2	36	53	90	72		
	6	5	0.073							
0.4	11	2	0.075	1	37	51	85	71	0.0252	0.0711
	6	4	0.195	2	38	52	84	70		
n_{GT}^{C}				1	36	54	85	70		
n_{GT}^{Y}				2	32	51	83	65		

T1 is the first tier of the plant growing row in the experimental area. n is the result of the yield monitoring algorithm. n_{GT}^{C} is the number of ripe strawberry fruit in the detection results. n_{GT}^{Y} is the number of ripe strawberries in the raw video. err^{C} is the relative error rate of fruit counting. err^{Y} is the relative error rate of yield monitoring.

schedules. The yield monitoring performance was found to have an error rate of 6.26% when the plants were inspected at a constant MRP traveling speed of 0.2 m/s. The temporal–spatial phenotypic data within the whole strawberry factory captured by the MRP could be further used to dynamically understand plant growth and provide data support for growth model construction and production management. The MRP's functions are expected to be transferable and expandable to other crop production monitoring and cultural tasks.

Data availability statement

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

Author contributions

GR, TL, YY, and KT: conception and design of the research; GR and HW: Hardware design; GR, HW, and AB: data preprocessing, model generation and testing, visualization, and writing—original draft; GR, TL, YY, and KT: writing—review and editing. GR, HW, and AB: validation. TL, YY, and KT: Supervision. All authors contributed to the article and approved the submitted version.

Funding

This research was partly funded by the Key R&D Program of Zhejiang Province under Grant Number 2022C02003 and the Student Research and Entrepreneurship Project of Zhejiang University.

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.2023.1162435/full#supplementary-material

References

Araus, J. L., Kefauver, S. C., Zaman-Allah, M., Olsen, M. S., and Cairns, J. E. (2018). Translating high-throughput phenotyping into genetic gain. *Trends Plant Sci.* 23, 451–466. doi: 10.1016/j.tplants.2018.02.001

Bao, Y., Tang, L., Breitzman, M. W., Fernandez, M. G. S., and Schnable, P. S. (2019). Field-based robotic phenotyping of sorghum plant architecture using stereo vision. *J. Field Robot.* 36, 397–415. doi: 10.1002/rob.21830

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, S. W., Nardari, G. V., Lee, E. S., Qu, C., Liu, X., Romero, R. A. F., et al. (2020). Sloam: Semantic lidar odometry and mapping for forest inventory. *IEEE Robot. Autom. Lett.* 5, 612–619. doi: 10.1109/LRA.2019.2963823

Chen, S. W., Shivakumar, S. S., Dcunha, S., Das, J., Okon, E., Qu, C., et al. (2017). Counting apples and oranges with deep learning: A data-driven approach. *IEEE Robot. Autom. Lett.* 2, 781–788. doi: 10.1109/LRA.2017.2651944

Debeunne, C., and Vivet, D. (2020). A review of visual-LiDAR fusion based simultaneous localization and mapping. *Sens* 20, 2068. doi: 10.3390/s20072068

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* 88, 303–338. doi: 10.1007/s11263-009-0275-4

Flueratoru, L., Wehrli, S., Magno, M., Lohan, E. S., and Niculescu, D. (2022). High-accuracy ranging and localization with ultrawideband communications for energy-constrained devices. *IEEE Internet Things J.* 9, 7463–7480. doi: 10.1109/JIOT.2021.3125256

Gongal, A., Amatya, S., Karkee, M., Zhang, Q., and Lewis, K. (2015). Sensors and systems for fruit detection and localization: A review. *Comput. Electron. Agric.* 116, 8–19. doi: 10.1016/j.compag.2015.05.021

Hayashi, S., Shigematsu, K., Yamamoto, S., Kobayashi, K., Kohno, Y., Kamata, J., et al. (2010). Evaluation of a strawberry-harvesting robot in a field test. *Biosyst. Eng.* 105, 160–171. doi: 10.1016/j.biosystemseng.2009.09.011

Hess, W., Kohler, D., Rapp, H., and Andor, D. (2016). "Real-time loop closure in 2D LIDAR SLAM," in *2016 IEEE international conference on robotics and automation (ICRA)* (IEEE), 1271–1278. doi: 10.1109/ICRA.2016.7487258

Higuti, V. A. H., Velasquez, A. E. B., Magalhaes, D. V., Becker, M., and Chowdhary, G. (2019). Under canopy light detection and ranging-based autonomous navigation. *J. Field Robot.* 36, 547–567. doi: 10.1002/rob.21852

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Kalen, M., et al. (2022). *ultralytics/yolov5: v7.0 - YOLOv5 SOTA realtime instance segmentation. v7.0* (Zenodo). NanoCode012.

Kirk, R., Mangan, M., and Cielniak, G. (2021). "Robust counting of soft fruit through occlusions with re-identification," in *2021 international conference on computer vision systems (ICVS)* (Verlag: Springer), 211–222. doi: 10.1007/978-3-030-87156-7_17

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "Imagenet classification with deep convolutional neural networks," in *2012 Advances in neural information processing systems (NIPS)*, Lake Tahoe, Nevada, USA. 1097–1105. doi: 10.1145/3065386

Lee, S., Arora, A. S., and Yun, C. M. (2022). Detecting strawberry diseases and pest infections in the very early stage with an ensemble deep-learning model. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.991134

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. 936–944. doi: 10.1109/CVPR.2017.106

Liu, X., Chen, S. W., Aditya, S., Sivakumar, N., Dcunha, S., Qu, C., et al. (2018b). "Robust fruit counting: Combining deep learning, tracking, and structure from motion," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain. 1045–1052. doi: 10.1109/IROS.2018.8594239

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018a). "Path aggregation network for instance segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA. 8759–8768. doi: 10.1109/CVPR.2018.00913

Mueller-Sim, T., Jenkins, M., Abel, J., and Kantor, G. (2017). "The robotanist: a ground-based agricultural robot for high-throughput crop phenotyping," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. (Singapore: IEEE), 3634–3639. doi: 10.1109/ICRA.2017.7989418

Olson, E. (2011). "AprilTag: A robust and flexible visual fiducial system," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China. 3400–3407. doi: 10.1109/ICRA.2011.5979561

Perez-Borrero, I., Marin-Santos, D., Vasallo-Vazquez, M. J., and Gegundez-Arias, M. E. (2021). A new deep-learning strawberry instance segmentation methodology based on a fully convolutional neural network. *Neural. Comput. Appl.* 33, 15059–15071. doi: 10.1007/s00521-021-06131-2

Qin, T., Li, P., and Shen, S. (2018). Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* 34, 1004–1020. doi: 10.1109/TRO.2018.2853729

Reiser, D., Miguel, G., Arellano, M. V., Griepentrog, H. W., and Paraforos, D. S. (2016). "Crop row detection in maize for developing navigation algorithms under changing plant growth stages," in *Robot 2015: Second Iberian Robotics Conference. Advances in Intelligent Systems and Computing*. 371–382 (Lisbon, Portugal: Springer). doi: 10.1007/978-3-319-27146-0_29

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). "Generalized intersection over union: A metric and a loss for bounding box regression," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, California, USA. 658–666. doi: 10.1109/CVPR.2019.00075

Shafiekhani, A., Kadam, S., Fritschi, F. B., and DeSouza, G. N. (2017). Vinobot and vinoculer: Two robotic platforms for high-throughput field phenotyping. *Sens* 17, 214. doi: 10.3390/s17010214

Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., and Rus, D. (2020). "LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA. 5135–5142. doi: 10.1109/IROS45743.2020.9341176

Talha, I., Muhammad, U., Abbas, K., and Hyongsuk, K. (2021). DAM: Hierarchical adaptive feature selection using convolution encoder decoder network for strawberry segmentation. *Front. Plant Sci.* 12. doi: 10.3389/fpls.2021.591333

Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M. N., et al. (2008). Autonomous driving in urban environments: Boss and the urban challenge. *J. Field Robot.* 25, 425–466. doi: 10.1002/rob.20255

Walter, A., Liebisch, F., and Hund, A. (2015). Plant phenotyping: from bean weighing to image analysis. *Plant Methods* 11, 1–11. doi: 10.1186/s13007-015-0056-8

Wang, J., and Olson, E. (2016). "AprilTag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea (South). 4193–4198. doi: 10.1109/IROS.2016.7759617

Zhang, J., Kantor, G., Bergerman, M., and Singh, S. (2012). "Monocular visual navigation of an autonomous vehicle in natural scene corridor-like environments," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura-Algarve, Portugal. 3659–3666. doi: 10.1109/IROS.2012.6385479

Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA. 6848–6856. doi: 10.1109/CVPR.2018.00716

Zhou, C., Hu, J., Xu, Z., Yue, J., Ye, H., and Yang, G. (2020). A novel greenhouse-based system for the detection and plumpness assessment of strawberry using an improved deep learning technique. *Front. Plant Sci.* 11. doi: 10.3389/fpls.2020.00559

frontiers | Frontiers in Plant Science

TYPE Review
PUBLISHED 22 May 2023
DOI 10.3389/fpls.2023.1156734

Check for updates

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU),
United States

REVIEWED BY
Sanjay Singh Rathore,
Indian Agricultural Research Institute
(ICAR), India
Silvia Pampana,
University of Pisa, Italy

*CORRESPONDENCE
Fernando Auat Cheein
f.auat@hw.ac.uk

RECEIVED 01 February 2023
ACCEPTED 04 April 2023
PUBLISHED 22 May 2023

CITATION
Johnson CM and Auat Cheein F (2023) Machinery for potato harvesting: a state-of-the-art review.
Front. Plant Sci. 14:1156734.
doi: 10.3389/fpls.2023.1156734

Machinery for potato harvesting: a state-of-the-art review

Ciaran Miceal Johnson and Fernando Auat Cheein*

UK National Robotarium, School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, United Kingdom

Potatoes are the fourth most important crop for human consumption. In the 18 century, potatoes saved the European population from starvation, and since then, it has become one of the primary crops cultivated in countries such as Spain, France, Germany, Ukraine and the United Kingdom. Potato production worldwide reached 368.8 million tonnes in 2019, 371.1 million tonnes in 2020, and 376.1 million tonnes in 2021, with production expected to grow alongside the worldwide population. However, the agricultural sector is currently suffering from urbanization. With the next generation of farmers relocating to cities, there is a diminishing and ageing agricultural workforce. Consequently, farms urgently need innovation, particularly from a technology perspective. As a result, this work is focused on reviewing the worldwide developments in potato harvesting, with an emphasis on mechatronics, the use of intelligent systems and the opportunities that arise from applications utilising the Internet of Things (IoT). Our work covers worldwide scientific publications in the last five years, sustained by public data made available from different governments. We end our review by providing a discussion on the future trends derived from our analysis.

KEYWORDS

potato harvesting, automation, machinery, internet of things, artificial intelligence, robotics

1 Introduction

Around the world, the strain placed upon agriculture is compounding. A diminishing pool of skilled laborers, the impact of climate change, and an ever-increasing human population are a few of the challenges facing modern agriculture. Potatoes, as the fourth most grown crop in the world behind wheat, rice, and corn, will play a large role in feeding the increasing population [Zhang et al. (2017); Jennings et al. (2020); Issa et al. (2020)]. Ensuring an efficient potato production pipeline is of great importance. The stage of the potato production pipeline which suffers the greatest losses is harvesting [Spang and Stevens (2018)]. Potato harvesting is the process of separating and collecting potato tubers from the soil. During this, losses occur as potatoes are damaged or left in the field.

There is not a single potato harvesting solution which generalizes well to all farms, geographies, and soil types. The mechanical design of potato harvesters depends heavily on the environment in which it operates. Regional factors along with the available harvesting methods can greatly impact potato production [Wei et al. (2019)], as can be seen in Figure 1. The production in the northern and central parts of the globe, which use mechanical harvesting, is significantly higher than in the southern hemisphere. There is also a great variation within hemispheres which is worth exploring in more detail. It is important not to simplify the problem, but to view the geographical and political issues which may arise when proposing certain solutions to potato harvesting.

Potatoes can be harvested using a variety of equipment. The most simplistic method of harvesting is manual. This can be done using a hand hoe, spading fork or even without any equipment. Harvesting by hand is a time-consuming and labor-intensive task [Gulati (2019)]. Therefore animal-drawn harvesters, such as the traditional plough, were deployed to solve these problems. Both methods of harvesting are still common practice in many parts of the world, despite draught animals being neglected and even sometimes harmed. Many veterinarians and animal welfare organizations continually advocate for an improvement in their living and working conditions [Ramaswamy (1998); Mota-Rojas et al. (2020)]. A step up in complexity introduces semi- and fully-mechanised harvesters. The difference is that fully-mechanised harvesters collect the potatoes in a trolley or bunker during harvesting, saving the manual labor required to collect the potatoes from the field by hand after harvesting. Mechanical harvesters are considered an improvement on the first two methods of harvesting as they reduce harvesting time, cost, and losses [Nasr et al. (2019); Soethoudt and Castelein (2021)]. Finally, there has been discussion regarding the automation of potato harvesters, though there is no working prototype in academic literature or at an industrial scale implementation [McPhee et al. (2020)].

This review will begin by looking into the current state of global potato harvesting, diving into the geographical differences and discussing reasons for these differences. Followed by potato harvesting constraints which may impact harvesting. These are potato and soil characteristics. The technology used in potato harvesting will be reviewed, starting with the mechanical harvester specifications and design. Followed by the future trends of potato harvesting. Finally, a discussion will be provided on the state of potato harvesting around the world, with the goal of specifying an automation level for the top-producing potato countries in each continent.

This work only considers scientific journal articles released from January 2017 – December 2022, and information available from governmental agencies. For a fair analysis, we kept our emphasis on articles from countries with available agricultural information. The selected articles were obtained through the Scopus database. Articles under the subject areas of Chemistry; Medicine; and Biochemistry, Genetics and Molecular Biology were automatically filtered out from the search. We also restricted the articles to only those with an English version. The focus on the selection of articles was put on the machinery for potato harvesting.

2 Potato harvesting: an international assessment

Potato harvesting is complex, with various different factors preventing farmers and scientists from finding an optimum –and unique– harvesting solution. The geographical location for example can impact the optimum harvesting solution due to variations in terrain, climate and soil characteristics. Consequently, farmers around the world require bespoke solutions to harvesting. The societal role of potatoes around the world also varies. The majority of potato farms in Asia, South America, and Africa are smallholders [Devaux et al. (2021)]. They treat potatoes as a staple crop and not necessarily as a cash crop. A staple crop is used to feed the general population and constitutes a significant proportion of the nation's diet. Cash crops on the other hand are grown in order to generate profit. There is a drive for these smallholders to increase their productivity by utilising modern farming techniques [Devaux et al. (2021); Wu et al. (2018)]. However, such techniques must be tailored to the farm in which they are deployed.

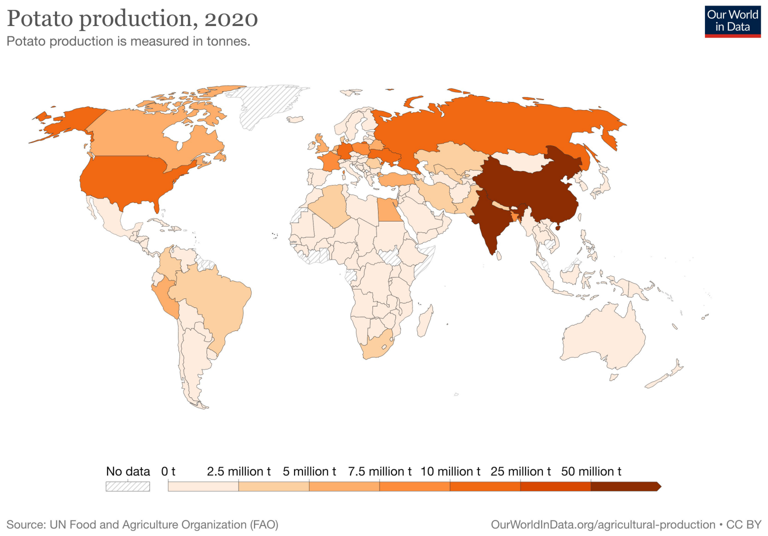

Potato production, 2020
Potato production is measured in tonnes.
Our World in Data
No data
0 t
2.5 million t
5 million t
7.5 million t
10 million t
25 million t
50 million t
Source: UN Food and Agriculture Organization (FAO)
OurWorldInData.org/agricultural-production • CC BY

FIGURE 1
Heatmap of the global production of potatoes in 2020 (in millions of tonnes). Image taken from Ritchie et al. (2023) using data made available by Dataset FAO (2022b).

2.1 Potato production by continent and country

The worldwide potato production landscape has changed in recent years, as shown in Figure 2. Formerly the highest potato-producing continent, Europe has experienced a large decline in potato production being surpassed by Asia as the top-producing continent. Africa also shows a rapid increase in potato production, while Oceania, South and North America display steadier growth.

The top potato-producing countries in each continent will be studied in this section. These are China, Ukraine, the USA, Peru, and Australia. Another country included in this section is India, as they are the second largest potato producer in the world after China. Germany, as they are the largest Western European potato producer. And the UK, as they recently left the European Union. The potato production, in tonnes, for each of these countries from 1961–2021 can be found in Figure 3.

An in-depth study of these countries will be provided for the years 2017–2021 since this review only considers scientific journal articles released from 2017–2022. Data for 2022 is not provided as it was not made available at the time of this review.

Potato production provides a one-dimensional view of a country's ability to grow and harvest potatoes. Larger countries can dedicate more land to growing and ultimately will produce more potatoes. This does not mean that they are efficient with their land use. In order to provide an insight into their efficiency we look at yield. Yield is the quantity of potatoes produced in a given area. Finally, the population of a country is discussed. A higher population may result in a greater need to produce potatoes in order to feed their population. Though a high population may also restrict their land use.

2.2 Asia

China and India are the top potato-producing countries in the world. Since China achieved the top spot in 1993, the nation has been pushing campaigns to increase its consumption of this food group [Devaux et al. (2021)]. Harvesting in China is split between fully- and semi-mechanized harvesters, with the majority of harvesting being semi-mechanized [Wei et al. (2019); Issa et al. (2020); Fu et al. (2022)]. Due to the heavy clay soil found in Northern China, their research revolves around removing soil after extraction [Fu et al. (2022); Wei et al. (2019)]. Currently, soil clods and stones are removed manually after harvesting. Though China is doing research into the use of computer vision to automate their removal (see Fu et al. (2022) and the references therein). India is also primarily a semi-mechanized harvesting nation [Gulati (2019)]. Though they are moving towards fully-mechanised harvesters, such as the one proposed by Gulati (2019).

By 2050, Rosegrant et al. (2017) predicts that China will be surpassed by India as the top potato-producing country. The results found by Rosegrant et al. (2017) was adapted by Devaux et al. (2021) producing the bar chart seen in Figure 4. Currently, India ranks second in potato production, population and area harvested. Although India has a higher yield than China it is still far smaller than other countries included in the survey. It is unclear whether improving yield will lead to higher production, as the reduction in yield may be due to factors such as continuous monoculture growing. Continuous monoculture growing can lead to disease 107 which reduces yield however continually growing potatoes may be the reason for higher production.

2.3 Western Europe

Before Brexit, 60% of the European (EU-28) potato production was produced in five Northwestern European countries. These countries are referred to in Goffart et al. (2022) and Devaux et al. (2021) as the NWEC-05. The NWEC-05 is made up of Germany, Belgium, France, Netherlands, and the UK. It is worth mentioning that the UK is no longer a member of the European Union, and therefore will be discussed separately.

The high level of mechanization seen in NWEC-05 is expensive [Goffart et al. (2022)]. Such costs are justified as these are advanced

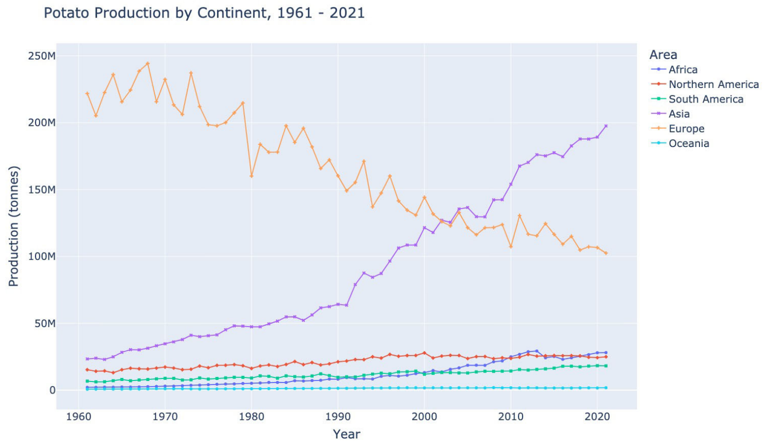


FIGURE 2
Potato production by continent for the years 1961 - 2021 (in millions of tonnes), using data made available by Dataset FAO (2022b).

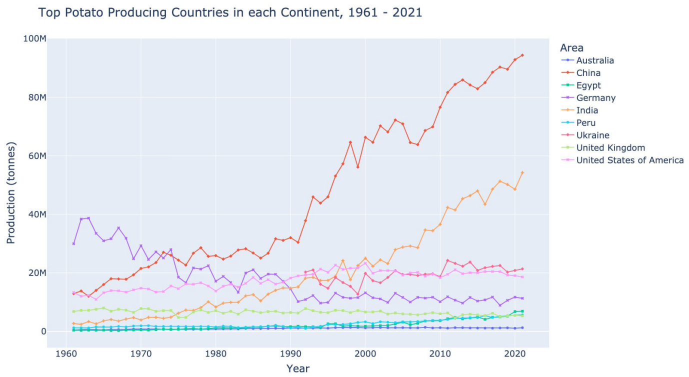


FIGURE 3
Top potato-producing countries by continent, including India, Germany, and the United Kingdom for the years 1961 - 2021 (in millions of tonnes), using data made available by Dataset FAO (2022b).

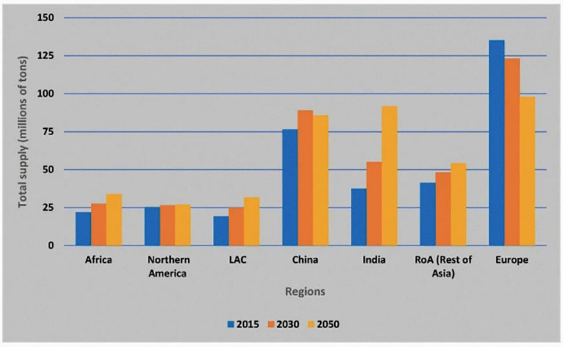


FIGURE 4
Prediction of future potato production taken from Devaux et al. (2021) adapted from Rosegrant et al. (2017).

and profitable sectors for the countries. This high level of automation in industrial farms is partially due to the fact that potatoes are seen as cash crops as well as staple crops in these countries. A high proportion of the crops are sold to processing companies. For example, in Belgium, only 20% of the potatoes are sold as fresh produce while the remaining 80% are sold for processing [Devaux et al. (2021)].

The top potato-producing country in Western Europe is Germany. As can be seen in Figure 2, potato production in Europe is declining. This is evident in the data provided by Germany. Between 2017–2021, a slight decline in production and an increase in the harvested area saw a large reduction in Germany's yield. Germany also had the smallest variation in population across the five years.

2.3.1 United Kingdom

As a member of NWEC-05, the UK was one of the top-producing potato countries in Europe. Similar to Germany, it has experienced a reduction in yield between 2017–2021. A noticeable difference however is that while Germany produced slightly fewer potatoes (-3.5%) by using more land (+3.1%); the UK produced significantly fewer potatoes (-14.7%) while using less land (-6.2%).

Figure 5, shows the potato production and yield for the three European countries discussed in this review: Germany, the UK, and Ukraine. Both Germany and the UK experience a local maximum in 2017 followed by a steep reduction in production and yield. These values begin to recover towards 2021 with Germany's recovering more quickly. This data shows that production can be greatly disrupted in one year and it may take several years to recover.

2.4 Eastern Europe

The third largest potato-producing country in the world behind China and India is Ukraine. Ukraine is a very active member of the potato harvesting research community. They are a fully-mechanised industry, although a significant number of the machines used to grow potatoes are imported from Russia, Belarus, and Germany [Hrushetsky et al. (2019); Hrushetskyi et al. (2021)]. Due to their heavy loam soil, the majority of research papers discuss the removal of soil clods from the harvesting process [Bulgakov et al. (2017; Bulgakov et al., 2019; Bulgakov et al., 2020; Bulgakov et al., 2021)]. The harvesting may be fully-mechanised however the removal of soil clods is still done manually which can be labor-intensive and expensive [Bulgakov et al. (2021)].

Referring back to Figure 5, it clearly shows that Ukraine produces more potatoes than its European counterparts, with drastically lower yet more stable yields. These low yields may be indicative of the loss found when harvesting in the heavy loam soil. Ukraine, like the UK, experienced a decrease in potato production, yield, and the harvested area between 2017–2021. However; Ukraine alone experienced a steady reduction in population between 2017–2021.

2.5 North America

In North America, like NWEC-05, potatoes are treated as cash crops: with US potato production in 2021 equating to 410 million cwt and processing accounting for 281 million cwt [USDA (2022)].

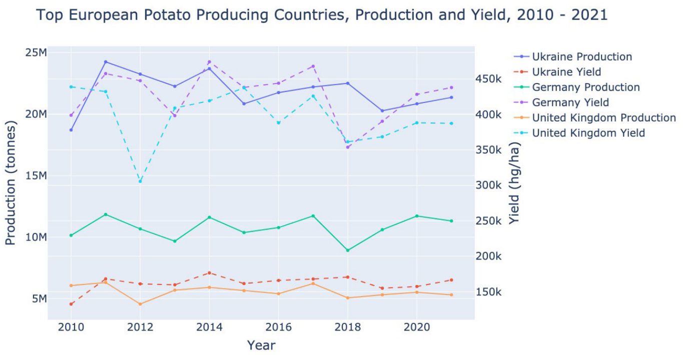


FIGURE 5
Potato production (in tonnes) and yield (in hg/ha) for Germany, Ukraine and the United Kingdom from 2010–2021. Data extracted from Dataset FAO (2022b).

Farmers can optimize their financial returns by meeting certain incentives in their contracts with processing companies [Waxman et al. (2018)]. Mechanical approaches to harvesting can help the farmer meet these incentives.

The United States of America experienced the highest yield of any country in the survey and even managed to increase their yield by +1.4% between 2017–2021. They experienced a decrease in production (-9.2%) however since their harvested area decreased by a large amount (-10.4%), their yield was not negatively affected. They also had an increase in population, which is the third largest population in the world behind China and India. However, unlike the other two, their potato production ranking does not equate to their population ranking.

2.6 South America

In countries such as Argentina, Brazil and Peru, potatoes are harvested mainly by semi-mechanised methods. In Argentina for example, only 10% of fresh potatoes are harvested by fully-mechanised approaches [The Bureau of the Netherlands Agricultural Council in Buenos Aires, (2008)]. Fully-mechanised approaches are more common in processed potato production, these are also often performed on larger areas of land. Semi-mechanised potato harvesters extract the potato from the soil and leave them in rows on top of the soil. The potatoes are then collected by hand and stored in large bags. These bags can remain in the field for up to 12 weeks, which ultimately results in large losses. In The Bureau of the Netherlands Agricultural Council in Buenos Aires (2008), it is suggested that harvesting can be performed better in fresh potato production systems with a mechanical method of picking up, cleaning, grading and bagging the potatoes after extraction from the soil.

Peru is the largest potato-producing country in South America. They experienced the greatest percentage increase in yield (+11.3%) while also having the smallest variation in yield across all countries in the survey. Peru also greatly increased its production (+18.5%) and harvested area (+6.5%) over the five years. Their population almost grew by the largest percentage between 2017–2021, just behind that of Egypt.

2.7 Africa

The papers discussing the agricultural landscape of Egypt and Eritrea state that it is constituted of many smaller farms [Nasr et al. (2018); Ghebreagziabiher et al. (2022)]. Such smallholder farmers will likely require smaller harvesters. This is the exact problem addressed in Nasr et al. (2018), where they proposed a semi-mechanised potato harvester for smallholder farms. Africa has the potential to increase potato production in the next few years through input intensification rather than area expansion, due to the increasing population [Devaux et al. (2021)]. Increasing potato production without increasing the area means an improvement in yield. This is beneficial since Sub-Saharan Africa suffers from a yield gap [Harahagazwe et al. (2018)].

Egypt experienced the highest percentage increase in production (+42.6%), population (+50.7%), and harvested area (+7.3%) between 2017–2021. Despite their yield decreasing by -5.4% during this time period, it remained higher than that of China, India, Ukraine and Peru showing that Egypt does not suffer the same yield gap as that seen in Sub-Saharan Africa.

2.8 Oceania

The top potato-producing country in the Oceanic continent is Australia. Potatoes are of great importance to Western Australia, as behind wine it is their second highest value-adding horticultural industry and their second highest value vegetable crop behind carrots [Dataset Government of Western Australia, A (2018)]. Nevertheless, compared globally, the country's production is low. Recent research conducted in Australia proposed the use of a fleet of small to medium-sized fully-autonomous potato harvesters [McPhee et al. (2020)]. Although this proposal displayed the highest level of automation out of all papers considered for this review, it was never implemented.

Australia experienced the smallest variation in production and area harvested during 2017–2021. Along with India and Peru, it is one of the only countries to experience a percentage increase in all four metrics between 2017–2021. Additionally, Australia had the smallest average production, harvested area and population of any country in the study.

3 Potato harvesting constraints

The efficiency of the potato harvesting process is affected by a number of issues. These range from environmental issues to farm management practices. This section will be focused on two specific issues, those exclusively related to the plant and those related to soil characteristics.

3.1 The potato characteristics that impact harvesting

Understanding the characteristics of different potatoes can result in better designed harvesters. Consideration of such characteristics during the design of mechanical harvesters and post-harvesting hardware can increase yield and reduce waste. For example, Ahangarnezhad et al. (2019) studied the *Agria* variety of potato and split the potato characteristics into physical and mechanical properties. The physical properties include the geometric and arithmetic mean diameter, which is important when designing potato sorting and packaging machines in order to reduce losses during transportation. The mass and volume of the potatoes are also physical properties, which should be considered when designing mechanisms for separating potatoes from other materials during harvesting.

However, when reducing waste, Ahangarnezhad et al. (2019) considers mechanical properties as the fundamental information required to design harvesting or post-harvesting machinery. Mechanical properties include the elasticity module, deformation energy, and fracture force. These properties can be determined by a uniaxial compression test. This test can generate a force-deformation graph, which plots the impact force against the penetration depth. When plotting the compression and restitution within the same graph, the area under the graph represents the energy absorbed by the potato. The energy absorbed by the potato is relevant as high energy absorption equates to high bruise damage [Surdilovic et al. (2018)].

It is to be noted that Ahangarnezhad et al. (2019) showed that many physical properties such as length, width, mass, and geometric mean diameter had a direct relationship to the potato size, while density had an inverse relationship. Relative density, also known as specific gravity, is one of the most important indicators of potato quality (see Waxman et al. (2018) for further reading). This is an estimate of the dry matter content of the potato, providing an indication of its water content. The water content of potatoes is relevant since, as stated by Surdilovic et al. (2018), potatoes with a higher water content experience less force yet higher deformations. Since higher levels of deformation equates to higher potato damage, possessing a high specific gravity is a desirable characteristic. This allows harvesters to move faster and exert more force on the potatoes while maintaining the same level of damage.

The specific gravity of potatoes can be influenced by a variety of factors. For example, Waxman et al. (2018) showed that the specific gravity can be influenced by harvest time and species of potato. Three potato varieties (Russet Burbank, Clearwater Russet, and Alpine Russet) were grown with harvest timings standardized based on that of Russet Burbank, a popular variety of potato used in the processing industry. There were three harvest timings used: approximately 2 weeks prior to normal harvest (early), normal Russet Burbank harvest time (normal), and approximately 2 weeks past normal harvest (late). They determined the specific gravity of the potatoes by two methods, weight-in-air and weight-in-water. A low specific gravity was indicative of an early harvest and a declining specific gravity was that of a late harvest. They also found that the species of potato had an impact on the specific gravity. Clearwater Russet exhibited the highest specific gravity in both years of the experiment.

Potatoes can be bred to have desirable characteristics such as a higher specific gravity. In Melito et al. (2017), an evaluation index is proposed to support the selection of clones with interesting trait combinations. As a result, they compared the tuber yield, specific gravity, chipping ability and earliness. They found a 48% higher productivity in clones compared to the best control. The various clone families had significantly different tuber specific gravity, with 70% of clones having a higher score than 1.080 which is the minimum required to be used in the processing industry. Potato processing contracts often contain Incentive Adjusted Prices (IAP) which provide farmers with financial incentives to produce higher quality potatoes. A common criterion in IAPs is producing potatoes over a certain specific gravity. Consequently, potatoes with a higher specific gravity are not only easier to harvest but also financially beneficial to the farmer.

3.2 The soil characteristics that impact harvesting

Applying the correct agronomic practices for a potato species can greatly improve the quality of potatoes produced. Agronomic practices and potato characteristics, such as flesh color, can impact the nutrition required to optimally grow and harvest potatoes [Vaitkevičienė et al. (2020)]. Furthermore, throughout the growth cycle, the nutritional demand and therefore availability of nutrients in the soil varies. This temporal availability of nutrients can be utilized by planting multiple species of crops in close proximity. This is called intercropping.

The goal of intercropping systems is to achieve a Land Equivalent Ratio (LER) > 1 [Dong et al. (2018)]. This would suggest that the crops are temporally or spatially cooperating and sharing resources. Conversely, an LER < 1 means the crops are in competition for resources and no benefit is gained from the intercropped system. Intercropping systems have multiple benefits such as reducing weeds and disease. Potato harvester designs should consider that there may be other crops, particularly above-ground crops, in close proximity to the potatoes. Farmers can also get similar benefits from crop rotation [Khakbazan et al. (2019)]. Reducing the load placed upon the farmer by maintaining multiple crops concurrently.

Finally, the soil type and water content can greatly impact tuber damage and loss when harvesting [Bulgakov et al. (2021); Wei et al. (2019)]. Heavy loam soil is considered particularly difficult to harvest as it is prone to compaction. This compaction leads to large soil clods getting extracted with the potatoes which in turn bruise and damage the potatoes. A low water content can also increase the probability of bruising and damaging the potato when harvesting [Wei et al. (2019)]. Soil water content can be controlled through irrigation [Tang et al. (2019)]. Irrigation can ensure that potatoes grow optimally and do not experience water stress. However, this can negatively impact the environment. As a result, the environmental impact should be minimized while also maximizing long-term yield [Tang et al. (2019)]. Table 1 displays the soil type and water content of the soil in literature. As can be seen in the table, several works discuss the soil type but fewer discuss water content. Reporting these values can help to improve the repeatability of experiments and also help identify trends that arise due to these variables.

4 Mechanical harvesters

4.1 Mechanical harvester specifications

When harvesting potatoes, a common design option is the tunable parameters. These parameters can be adjusted in the field to optimize the performance of the harvester. The characteristics of the potato and the soil can influence the optimal parameters. This review will focus on the forward and conveyor speed of the harvesters as well as the digging depth and angle. Forward speed is the velocity of the harvester as it moves along the farm when harvesting. Conveyor speed is the velocity of the conveyor belt that lifts the potatoes out of the soil and places them in a collection device or in windrows. The digging angle is the angle of the digging

TABLE 1 Potato harvesting papers from 2017–2022, the country of their experiment, and soil characteristics that impact harvesting.

Publication	Country	Soil Type	Water Content
Muneer and Dowell (2022)	Scotland	–	–
McPhee et al. (2020)	Australia	Clay loam	–
Issa et al. (2020)	China	Sandy clay	23.8
Fu et al. (2022)	China	Heavy clay	–
Wei et al. (2019)	China	Sandy, clayey	15.6
Tang et al. (2019)	China	–	–
Dong et al. (2018)	China	Orthic anthrosol	–
Bulgakov et al. (2021)	Ukraine	Heavy loam	15–25
Hrushetskyi et al. (2021)	Ukraine	Average loam	16.5
Bulgakov et al. (2017)	Ukraine	Medium loamy	11
Hrushetsky et al. (2019)	Ukraine	Loamy and sandy	–
Bulgakov et al. (2020)	Ukraine	–	11
Bulgakov et al. (2019)	Ukraine	–	–
Poppa et al. (2020)	Germany	–	–
Surdilovic et al. (2018)	Germany	–	–
Schneider et al. (2019)	Austria, Germany	–	–
Nasr et al. (2018)	Egypt	Clay loam	–
Ghebreagziabiher et al. (2022)	Eritrea	–	–
Melito et al. (2017)	Italy	–	–
Sibirev et al. (2019)	Russia	Sandy	21.5
Gulati (2019)	India	Sandy to sandy loam	–
Khakbazan et al. (2019)	Canada	Silty clay loam	–
Vaitkevičienė et al. (2020)	Lithuania	–	–
Vezirov et al. (2021)	Bulgaria	–	–
Ahangarnezhad et al. (2019)	Iran	–	–
Waxman et al. (2018)	USA	Silt loam	–

The soil characteristics are soil type and water content.
- means no data reported.

blade in the soil and the digging depth is the depth. In recent literature, forward speed has varied from 0.9–7.9km/h, conveyor speed from 0.2–2.37m/s, digger angle from 10–24°, and digging depth from 12–27cm. The full list of publications discussing one of these parameters –over the period under study– and the parameters they used are presented in Table 2.

As much as soil and potato characteristics can help indicate the optimal harvester parameters, the main criterion affecting these parameters is the farmer's optimization criterion. The farmer has to balance several objectives such as reducing tuber damage and loss while increasing their harvesting efficiency. Varying the forward speed of the harvester can result in a variety of outcomes. One outcome which is impacted by varying the forward speed is tuber damage and loss. For example, Bulgakov et al. (2021) found that increasing forward speed from 2.9–7.9km/h while increasing their rotor diameter from 0.65–1m decreased their tuber damage rate from 4.2% to 1.5%. This is in line with Bulgakov et al. (2017), which shows an increase in forward speed decreases the percentage of damaged tubers greatly, despite the percentage of tubers lost increasing. However, contradictory results have been found by Hrushetsky et al. (2019) and Issa et al. (2020), who found that increasing forward speed increased tuber damage. Additionally, Hrushetsky et al. (2019) witnessed an increase in both tuber loss and damage percentage when increasing forward speed for their design and that of the KST-1,4, which is a standard serial potato digging machine.

Another factor impacted by forward speed is separation efficiency. In Bulgakov et al. (2017); Bulgakov et al., (2021), the impact of forward speed on separation efficiency is studied. In both works, they notice that increasing forward speed up to a point can improve separation efficiency, after which increasing forward speed decreases performance. In Bulgakov et al. (2017), separation efficiency increased slowly up to 2.4km/h after which there was a slow decrease from 2.4 to 3.0km/h. As forward speed is further increased to 4.0km/h a sharp drop in separation efficiency is observed. This is confirmed in Bulgakov et al. (2021), where increasing forward velocity from 2.9–5.4 km/h while increasing the rotor diameter from 0.65–1m improved soil separation. However, when further increasing the forward speed from 5.4–7.9km/h they found that soil separation decreased.

Finally, forward speed also impacts field capacity and harvesting efficiency; Issa et al. (2020) found that in general increasing forward speed, increased actual field capacity and the power required by the harvester, while also decreasing field efficiency and the specific energy consumption of the harvester. Another observation from this paper was that increasing forward speed from 2.5–4.5km/h increased the tuber lifting percentage. Although tuber lifting percentage decreased when further increasing forward speed from 4.5–6.5km/h.

Digging angle and depth are similar as a greater digging angle equates to a greater digging depth. We can reduce tuber loss by varying the digging angle: Issa et al. (2020) found that the lifted potato percentage increased from 87.63% to 95.14% with an increase in digging angle from 12°to 22°. The total potato damage also decreased with an increase in the digging angle. However, increasing the digging angle increased the soil resistance resulting in a decreased actual field capacity and field efficiency alongside an increase in required specific energy and power.

Increasing conveyor speed can also increase tuber damage: Wei et al. (2019) acknowledges that at various stages of the potato-soil separation process, the potato will experience different levels of soil cushioning. As a result, they vary soil-potato proportions, splitting them into three groups: the primary clod-crushing stage (7.83% - 38.55%), intermediate clod-crushing stage (38.55% - 69.28%) and fine clod-crushing stage (59.04% - 69.28%). They also experiment with agitator frequency and amplitude measuring the number of impacts, impact acceleration, impact duration, and velocity change as an indicator of potato bruising and damage probability. Potato bruising was broken into 4 groups: no bruising, slight bruising, moderate bruising, and severe bruising. Varying the potato-soil

TABLE 2 Harvester specifications in papers from 2017–2022.

Publication	Forward Speed (km/h)	Digging Depth (cm)	Digger Angle (°)	Conveyor Speed (m/s)
Bulgakov et al. (2021)	2.9, 3.6, 5.4, 7.2, 7.9	27	10	1.91
Issa et al. (2020)	2.5, 4.5, 6.5	14–25	12, 17, 22	0.78, 1.11
Hrushetskyi et al. (2021)	7.92	14–25	16–24	–
Bulgakov et al. (2017)	1.9, 2.4, 3.0, 4.0	27	–	1.81–2.37
Sibirev et al. (2019)	3–5.2	12–18	–	1–1.78
Nasr et al. (2018)	1.5, 2.0, 2.5	16, 20, 24	–	–
Hrushetsky et al. (2019)	0.9, 1.8, 2.7, 3.6, 4.5	–	–	–
Gulati (2019)	2.7	–	–	–
Fu et al. (2022)	–	–	–	0.2, 0.4, 0.6, 0.8, 1.0
Poppa et al. (2020)	–	–	–	0.33, 1.00
Wei et al. (2019)	–	–	–	1.54, 1.80, 2.06

Forward speed of the harvester in km/h. Digging depth of the harvester blade in cm. Digger blade angle in °. Conveyor speed in m/s.
- means no data reported.

proportion had a large influence on the harvest quality and the impact characteristics experienced during the separation process. As the potato-soil proportion increased and the soil cushion decreased, the number of impacts and peak impact acceleration increased. A slight increase was seen between the primary and intermediate stages but a significant increase was observed between the intermediate and fine clod-crushing stages. The movement of potatoes on the conveying device also varied depending on the stage. At the primary stage, there was little potato movement, at the intermediate stage the potatoes were rolling, and at the fine stage, potatoes were jumping and rolling increasing the damage probability. As the agitator vibration intensity increased the number of impacts and peak impact acceleration also increased gradually. Consequently, vibration intensity should be selected in order to reduce bruising and mechanical damage while maximizing separation efficiency. The impact of potato-soil proportion was more obvious than that of the conveyor running speed. Although at 2.06m/s, the peak impact acceleration at intermediate and fine potato-soil separation was higher than when the conveyor speed was 1.54 and 1.80m/s. The number of impacts was slightly smaller at 2.06m/s compared to 1.80m/s. They do state that increasing speed, increases separation efficiency, and if the rod-type conveyor speed is too slow it will negatively impact harvesting efficiency. However, increasing the conveyor speed will increase the linear velocity of the potatoes as they fall into the windrows or containers which can cause damage.

An opposing discovery is presented by Bulgakov et al. (2017), who shows that the percentage of soil separation and separation intensity both decrease with an increase in conveyor speed. Finally, Issa et al. (2020) state that the actual field capacity and field efficiency increase with conveyor speed, although they conclude that varying conveyor speed had no significant impact on tuber damage. They also find that an increase in conveyor speed decreased tuber lifting percentage.

4.2 Mechanical harvester designs

There is a significant amount of research into the mechanical design of potato harvesters. These designs vary in complexity, from simple designs focused on harvester specifications such as digging depth and forward speed to more complex designs with agitators and rotary components to remove soil clods from the production pipeline.

Designing mechanised potato harvesters has proven to be a constant trade-off between efficiency and potato damage. Designs which improve efficiency while minimizing damage are highly desirable. One common design option which can be altered to optimize this goal is the sub-cultivating working parts of the harvester. These parts are important in breaking up the soil and reducing tuber damage. Done effectively, tuber damage can be reduced and efficiency increased: Hrushetsky et al. (2019) proposed to improve harvesting efficiency with a digging component that utilizes a passive blade with cutting discs and soil compactors. The design reduces the tractive resistance of the potato digger by 18% while improving the buckling rate of the potato-soil layer. Ultimately increasing productivity by 22%, achieving a yield of 13.2 t/ha and a digging completeness of 99.1% compared to the serial digger KST-1,4 which achieved 97.6%. This is similar to the work of Hrushetskyi et al. (2021) who aimed at reducing tuber mechanical damages while providing qualitative indicators of the potato heap separation process. They achieved this by mathematically modelling the movement of particles when the share-board surface of the harvester collided with the potato heap. Similar to the work done by Hrushetsky et al. (2019), they compare their theoretical and experimental results, showing their model to have a deviation within 5%. They concluded that this indicates the adequacy of their mathematical model to simulate the separation process of potato heaps.

A different approach to improving the time efficiency of potato harvesting was taken by Gulati (2019) by designing a two row combine harvester. Their harvester can reduce labor, time, and expenses by harvesting two rows of potatoes at once. The design works again by breaking the soil ridge, exposing the potatoes so they can be easily and efficiently collected. These potatoes are then lifted from the soil and conveyed to the following trolley using a rod-chain separator-conveyor and a swan-neck elevator-conveyor. Two sets of agitators are attached to the conveying system. The purpose of the rod-chain separator-conveyor system with agitators is to remove the soil, stems and debris from the collected potatoes with minimal injuries. Their prototype was able to operate with a single 40 horsepower tractor and has an effective field capacity of 0.26 ha/hr, tuber bruising of 6%, and 98.4% of the excavated potatoes made it to the trolley with a field loss of 1.6%.

Finally, Bulgakov et al. (2017; Bulgakov et al., 2019; Bulgakov et al., 2020; Bulgakov et al., 2021) published four articles during 2017–2021 related to the use of rotary components in potato harvesting. The goal of this research was to clean the potatoes and in particular remove soil clods. This was achieved by a variety of designs however the key connection was that of rotation. Their later publication from 2021, relates to the concept of breaking up the potato-soil layer and therefore will be discussed first. They designed a rotary-type potato harvester that improves soil-clod separation in heavy loam soil [Bulgakov et al. (2021)]. The rotational component was added to help break up the soil, reducing the number of soil clods lifted onto the separation tool. Their proposed design can be seen in Figure 6. They varied the translational velocity of the machine, the rotor rotation frequency, the rotor diameter, the rotor circumference and the distance between the spherical discs to determine their effects on performance. They found that the soil separation improves as the rotor diameter increases from 0.65 to 1.0m and translational velocity increases from 0.8 to 1.5m/s. However, when velocity increases from 1.5 to 2.2m/s soil separation decreases. Also, tuber damage rates decrease from 4.2 to 1.5% when rotor diameter increases from 0.65 to 1.0m and translational velocity increases from 0.8 to 2.2m/s. When the distance between the rotors' circumference and the spherical discs increases, the tuber damage rate also increases. The maximum soil separation reached was 93.5%.

Other approaches by the same authors, discuss the concept of a spiral soil separator that can be included in the conveyor system. For example, Bulgakov et al. (2017) proposed a novel design for a spiral potato heap separator. This design can be seen in Figure 7. They believe

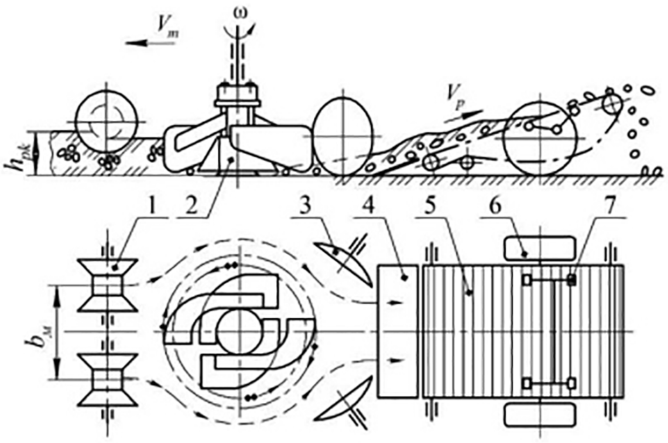


FIGURE 6
The design of a rotary-type potato harvester to improve soil clod separation. Image is taken from Bulgakov et al. (2021).

that their spiral separator in conjunction with other technical solutions such as agitators can self-clean the rollers resulting in improved soil separation. Their initial experiments corroborated this belief. They found the optimal parameters to be: a peripheral speed of rotation of 1.75–2.0m/s; an inclination angle of the separator to the horizon of 15–19°; and the installation eccentricity of the spirals as 5–10mm. The recommended forward speed was 0.6–0.8m/s (2.16–2.88km/h). Increasing the inclination angle of the separator and eccentricity of the spirals increased soil sifting and separation intensity. Conversely, increasing the peripheral speed of rotation towards 2m/s gradually decreased the percentage of sifted soil. After 2m/s a rapid decrease in the percentage of sifted soil was observed, this is due to a reduction in the contact time between the potato-soil mixture and the separator.

The concept of a spiral separator was further developed in their work, Bulgakov et al. (2019). In this paper, they discuss a theoretical design with the goal of removing soil clods and unwanted debris. They define a mathematical model for sieving potatoes on a spiral separator and use Matlab to compare the impact of different variables on the time taken to remove soil clods. They find that as the angular velocity goes from 10 to 50 rad/s the time to complete sieving goes from 0.07 to 0.025s. As the spiral's radius goes from 0.1 to 0.3m the time to complete sieving goes from 0.04 to 0.01s.

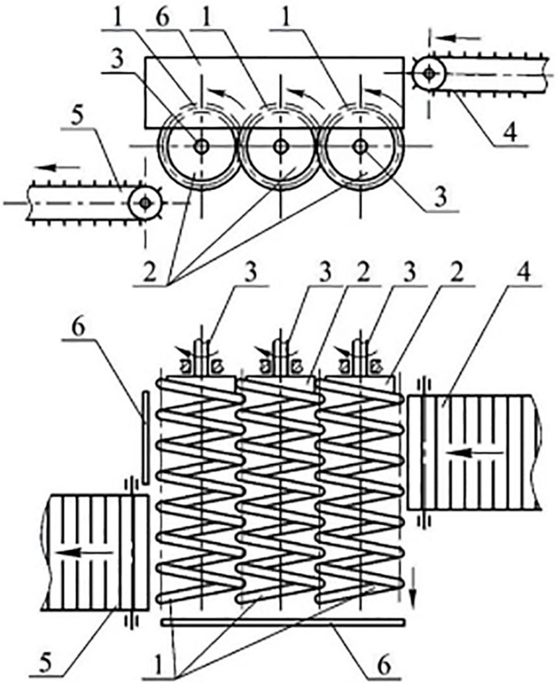


FIGURE 7
The spiral potato heap separator design. Image taken from Bulgakov et al. (2017).

Increasing the cleaning spiral's helix angle from 10°to 30°at a radius of 0.1m reduces the time to complete sieving from 0.75 to 0.026s, at a radius of 0.28m it goes from 0.28 to 0.005s. Varying the amplitude of oscillation of the spiral does not significantly impact the soil clod's residual mass.

Later in 2020, Bulgakov et al. (2020) implemented the spiral separator with the goal of removing more clods, soil, plant debris, and stones on the field so it is better environmentally. The spiral potato cleaner contained three cleaning spirals mounted as cantilevers. One end of each spiral is fixed on the hubs connected to the driving shaft. The soil mixture is dropped from a small height, which partially destroys the soil layer around the potato. Since the spirals are cantilevers the free ends make oscillatory movements in the longitudinal-vertical plane. There are gaps between the spirals which allow small soil clods and plant debris to fall through. The theoretical study of the motion and sifting of a body on the surface of the spiral-type potato cleaner is based on the basic principles of the dynamics of the motion of a body of variable mass. Their equation takes into account that the mass of the soil clod will decrease over time. Field experiments were used to determine the performance of the potato cleaner. The following indicators were used to determine the quality of the spiral-type potato cleaner: the screening ability of the cleaner, the intensity separation of admixtures, and the specific separation intensity. They then performed regression on each quality indicator. The cleaning ability of their design can be improved by altering the angular velocity, the initial angle of inclination, and the radius of the spirals. A soil clod reduction of 95% in the time range of 4.8–7.2s was achieved. Similar to conveyor speed, too fast of an angular velocity reduces the contact time between the soil clods and the spirals, reducing the potato cleaner's separation performance. Decreasing the initial angle of contact between the potato-soil layer and the spiral cleaner positively impacts the separation rate of the soil admixtures from the potato heap.

5 Trends in potato harvesting

One trend identified during the review was the use of electronic potatoes to understand the impact forces applied on the potato throughout the harvesting process. This is important not only when designing a potato harvester but also when selecting the harvester specifications. Electronic potatoes are objects designed to be as similar as possible to actual potatoes while containing sensors that can record the forces exerted on them. They have been utilized by Sibirev et al. (2019), to determine the impact forces experienced by potatoes during the full harvesting process for three different potato harvesters: AVR-Spirit-6200, Dewulf RA-3060 and Bolko. This study varied the forward speed, depth of the ploughshare in the soil, and the speed of the open-web elevator to determine their influence. However, the difficulty with electronic potatoes revolves around correctly modelling the potato in order to gain accurate measurements. One paper using the coefficient of restitution and the static modulus of elasticity to better model the impact characteristics and elasticity of potatoes is Surdilovic et al. (2018). The aim of this paper is to better understand the forces applied to potatoes when they fall. They found that all bar one of their dummies did not accurately represent real potatoes. Noting that the dummy potatoes had a higher maximum impact and acceleration with a lower deformation.

Several publications, in the period under study, look to change the status quo of potato harvesting procedures. The first of which is that farmers are currently not accurately reporting the waste generated during potato harvesting. As stated by Schneider et al. (2019), undersized potatoes that get composted should be reported as waste. Subsequently, they provide a practical approach for determining potato losses directly on the field. Their study included two farms, one in Austria, and the other in Germany. They consider two types of loss, type one, those remaining in the soil not collected by the harvester, and type two, those sorted out due to technical or quality reasons. In Austria, they used a net to catch type two, the net also helped to represent the area that needed to be excavated to find type one. In the German farm, the farmer de-haulms the potatoes prior to harvest and plants mustard plants. The roots of the mustard plant loosen the soil and elevate the potatoes. Due to this elevation, the potatoes are easier to extract from the soil which allows the harvester to drive faster. Small potatoes at the root of the plant are not economically viable for farmers to collect. As a result, they set shallower digging angles to save fuel. These smaller potatoes are often automatically filtered out by potato harvesters as they fall through gaps in the conveyor system which are intended to remove soil clods from the system. In Austria, loss two was higher than loss one while in Germany loss one was higher than loss two. The German farm on average produced larger potatoes which were cut in half by the harvester. This in conjunction with several smaller potatoes caused loss one to outnumber loss two. Overall, the loss in Germany was 1.4% compared to 9.1% in Austria. They conclude that losses during primary production are highly variable depending on region, weather, type of crop as well as cultivator and harvest method. They surmise that the harvester specifications such as digging depth and forward speed have a big impact on tuber loss. Their final proposal uses 2-4 people to determine loss, by collecting and weighing the potatoes on the field.

Another trend potentially interrupting the status quo around the world is the push to use more renewable energy. In particular, the trend towards electric vehicles, and potato harvesting is not exempt from such changes: Muneer and Dowell (2022) provides a case study on the use of renewable energy on a potato farm in Scotland, UK. In the case study, they compare the prices of different energy sources. They show that the cost of generating one kWh of energy using solar and wind power is lower than coal, gas, geothermal and nuclear. And that the cost has dropped significantly in the last 10 years as renewable technology improves. In order to prove that renewable energy is appropriate when potato harvesting they need to ensure that power is consistently supplied to the farm year-round and that the equipment used to generate the energy will not need to be replaced frequently. To measure the performance of the wind turbine they measure the average wind speed (m/s), average power (kW), and capacity factor (ratio) for wind turbines across the years of their experiment as well as across the months of 2015. They also provide the energy generated and capacity factor for solar power. Solar power generates the most energy in summer, while wind generates the most energy in the winter. In Scotland wind power generates more energy than solar power. A combination of the two can provide enough energy year-round to harvest and store potatoes. They notice that the months from March to June produce the most energy when combining both sources of energy. They also state that if maintained correctly then the output from both solar and wind energy does not deteriorate significantly in the first eight years. With some countries signing on to meet specific climate targets the pressure placed on agriculture to reduce its emissions will increase. This may lead to more farms following the blueprint provided in this article and therefore electronic tractors and potato harvesters may increase in demand.

Finally, McPhee et al. (2020) attempts to model the impact of low-mass autonomous vehicles on soil bulk density using COMPSOIL. They also look at the critical soil bulk density and what this means for harvesting two different crops, one of which is potato. They determine suitability in terms of operational capacity and what this means logistically for farming operations. They wish to determine the correct size of machine which will reduce traffic-induced soil compaction while still meeting a certain standard of productivity. They determine that a medium-sized autonomous fleet integrated into a Controlled Traffic Farming (CTF) approach would be best equipped to meet these requirements. However, CTF is not suitable for root and tuber farming as the harvester must currently drive over the top of the crops. They also state that even low-mass autonomous vehicles breach critical bulk density and therefore are not a solution for avoiding soil compaction in potato harvesting. They claim that alternative harvester designs must be created to avoid soil compaction for potato harvesting.

6 Discussion

A better understanding of potato characteristics can improve the design of the equipment involved in the harvesting and post-harvesting processes. However, publications such as Ahangarnezhad et al. (2019) need to ensure they develop upon previous work in the field so as to not waste time repeating the work of others. This paper for example only discussed one other paper which explores the physical and mechanical properties of the potato. Despite this being a common area of research, especially in the creation of electronic potatoes. They could also help to further develop the community and improve the repeatability of their experiment by providing the soil type and growing conditions for the potato they used in their experiments. The following subsections, are the main outcomes of the analysis covered in this work.

6.1 Farming land vs. population

The average population, harvested area, production and yield are used to produce Figures 8, 9. In Figure 8, the average potato production for 2017–2021 is plotted against the average harvested area for this time period. This graph shows that generally, the larger the average harvested area the higher the average potato production. The size of each circle equates to the average population of the country. Countries with a larger population tend to produce more potatoes than those with a smaller population.

The opposite relationship between potato production and harvested area is seen when comparing the average potato yield against the average harvested area for 2017–2021 (see Figure 9). As the average harvested area increases the average potato yield decreases. Again, the population size is represented by the size of the circle. However in this case there appears to be no clear relationship between the population size and yield.

6.2 Conflicting harvester specifications

Harvester specifications are specific to the field and design of the potato harvester. Therefore research can often appear to contradict one another. For example, Bulgakov et al. (2021) states that increasing forward speed decreases tuber damage while Issa et al. (2020) found that increasing forward speed increased damage. There are two important factors to discuss here. Firstly soil type, Bulgakov et al. (2021) performed their experiments in heavy loam soil which is notoriously difficult to harvest in due to the high percentage of soil

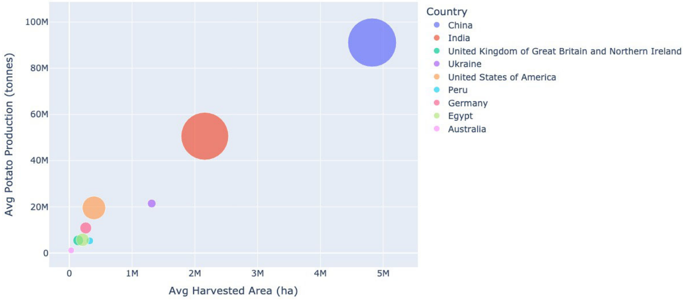


FIGURE 8
The average potato production (tonnes) between 2017–2021, against the average harvested area dedicated to growing potatoes (ha) for the same time period for each country displayed. The size of each circle equates to the size of that countries population. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

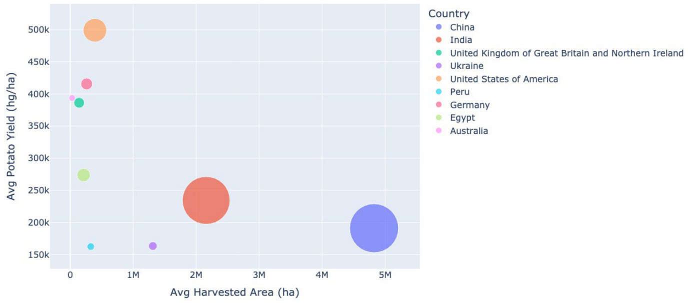


FIGURE 9
The average yield (hg/ha) between 2017–2021, against the average harvested area dedicated to growing potatoes (ha) for the same time period for each country displayed. The size of each circle equates to the size of that countries population. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

clods. In Issa et al. (2020), experiments were performed over sandy clay, which is a more preferable environment for potato harvesters. Most well-known potato combine harvesters are built to only operate in sandy soils (see Bulgakov et al. (2021)]. The second is the design of the harvester. The harvester design influences how forces are applied to the soil and potato. Therefore changing the harvesting specifications will vary their impact. Different designs will have different optimal harvester specifications.

6.3 Levels of automation

This section looks at the current level of automation present in each of the countries discussed in this review. The levels of automation described here are based loosely on those presented by the SAE International On-Road Automated Driving Committee, O (2021). The levels however differ slightly and their definitions are presented below: Level 0 equates to hand harvesting. Level 1 is a semi-mechanised harvester. Level 2 is a fully-mechanised harvester. Level 3 is partial automation of the harvesting process. Level 4 is the full automation of the harvesting process. Level 5 is the full automation of the potato farming process.

The only work reporting on automated potato harvesting was McPhee et al. (2020). However it was a hypothetical proposal, no potato harvester was actually automated. As such the highest level of automation was achieved by Fu et al. (2022), with their autonomous potato cleaner. This device was not attached to a harvester and therefore it is not considered part of the harvesting process. Since no other paper discussed automation, the top level of automation in potato harvesting is therefore Level 2. There were no potato harvesting papers produced by Peru and therefore it was not assigned a level of automation. However, based on surrounding countries, it is likely that Peru is Level 1. Table 3 summarizes the automation levels of potato harvesting in the different countries under study (over the period covered in this review).

China, India, Germany, and Australia were all assigned Level 2 due to reviewed papers from these countries discussing fully-mechanised harvesters [Fu et al. (2022); Gulati (2019); Schneider et al. (2019); McPhee et al. (2020)]. Ukraine, the USA, and the UK were also assigned Level 2, though this decision was arrived at based on additional papers not included in the survey [Bulgakov et al. (2022); Spang and Stevens (2018); Godwin et al. (1999)]. The UK and Germany are also part of NWEC-05 which as discussed by Goffart et al. (2022) has a very high level of mechanization, this confirmed their assignment as Level 2. Egypt was assigned Level 1 based on their

TABLE 3 The levels of potato harvesting automation, number of potato harvesting based journal publications between 2017–2022; as well as production and yield in 2021 for the top potato producing countries by continent.

Countries	Automation Level	Number of Publications	Potato Production (tonnes)	Yield (hg/ha)
China	2	5	**94,362,175.0**	163,179.0
Ukraine	2	**6**	21,356,320.0	166,430.0
India	2	1	54,230,000.0	241,237.0
Germany	2	3	11,312,100.0	437,944.0
UK	2	1	5,306,719.8	387,352.0
Australia	2	1	1,267,638.6	403,372.0
USA	2	1	18,582,370.0	**490,727.0**
Egypt	1	2	6,902,817.0	262,758.0
Peru	–	0	5,661,443.0	171,245.0

Bold values means larger value.

reviewed papers [Nasr et al. (2018; Nasr et al., 2019)] proposing a semi-mechanised potato harvester.

The following arguments can be made to change the automation level for China, India, and Ukraine. China and India are both primarily semi-mechanised harvesting countries [Wei et al. (2019); Gulati (2019)]. Despite this, they have both produced papers in the last five years discussing the use of fully-mechanised harvesters [Gulati (2019); Fu et al. (2022)]. As a result, both have been assigned an automation Level 2.

According to Hrushetskyi et al. (2021) and Hrushetsky et al. (2019) the majority of Ukrainian potato harvesting is carried out manually, despite previously most harvesting being mechanised. The majority of potato harvesters are imported from Russia, Belarus and Germany and are outdated. Nevertheless, since Ukrainian research papers discuss fully-mechanised approaches [Bulgakov et al. (2022)] they have been assigned an automation Level 2.

7 Conclusion and future work

Potato harvesting is a complex problem as the optimal solution varies around the world. Potato and soil characteristics contribute to the selection of an optimal harvesting technique and harvester specification. In the last five years, automation in potato harvesting has been discussed hypothetically but not implemented. Subsequently, the highest level of automation is fully mechanised harvesting (Automation Level 2). In recent literature, the design of mechanical potato harvesters has revolved around the breaking up and removal of soil clods. In addition to an improved ability to remove soil clods, future harvesters may also be electric as the need to reduce the environmental impact of farming increases. Intelligent systems such as electronic potatoes can help to reduce tuber damage and loss by understanding the forces exerted on the potato during harvesting. Nevertheless, there is a gap for intelligent systems in potato harvesting research. Introducing these intelligent systems may help to ease the strain placed on the agricultural sector caused by a shrinking workforce and an increasing population.

Author contributions

FA contributed to the conception and design of the study. CJ organized the papers and data, performed the analysis, and wrote the first draft. All authors contributed to the article and approved the submitted version.

Funding

PhD funding is provided by Engineering and Physical Sciences Research Council (EPSRC) Centre for Doctoral Training in Robotics and Autonomous Systems.

Acknowledgments

Thank you to Edinburgh Centre for Robotics at Heriot-Watt University and the University of Edinburgh for their support on this review.

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

Ahangarnezhad, N., Najafi, G., and Jahanbakhshi, A. (2019). Determination of the physical and mechanical properties of a potato (the agria variety) in order to mechanise the harvesting and post-harvesting operations. *Res. Agric. Eng.* 65, 33–39. doi: 10.17221/122/2017-RAE

Bulgakov, V., Bonchik, V. S., Holovach, I., Fedosiy, I., Volskiy, V., Melnik, V., et al. (2021). Justification of parameters for novel rotary potato harvesting machine. *Agron. Res.* 19, 994–1007. doi: 10.15159/ar.21.079

Bulgakov, V., Ivanovs, S., Adamchuk, V., and Ihnatiev, Y. (2017). Investigation of the influence of the parameters of the experimental spiral potato heap separator on the quality of work. *Agron. Res.* 15, 44–54.

Bulgakov, V., Ivanovs, S., Pascuzzi, S., Adamchuk, V., Ruzhylo, Z., Ihnatiev, Y., et al. (2022). Experimental research of quality indicators of operation of new potato harvester. *Engineering for Rural Development* 21, 701–707. doi: 10.22616/ERDev.2022.21.TF222

Bulgakov, V., Pascuzzi, S., Ivanovs, S., Ruzhylo, Z., Fedosiy, I., and Santoro, F. (2020). A new spiral potato cleaner to enhance the removal of impurities and soil clods in potato harvesting. *Sustainability* 12. doi: 10.3390/su12239788

Bulgakov, V., Pascuzzi, S., Nikolaenko, S., Santoro, F., Anifantis, A. S., and Olt, J. (2019). Theoretical study on sieving of potato heap elements in spiral separator. *Agron. Res.* 17, 33–48. doi: 10.15159/ar.19.073

Dataset FAO (2022a) *Annual population*. Available at: https://www.fao.org/faostat/en/#data/OA (Accessed 11/11/2022). License: CC BY-NC-SA 3.0 IGO.

Dataset FAO (2022b) *Crops and livestock products*. Available at: https://www.fao.org/faostat/en/#data/QCL (Accessed 11/11/2022). License: CC BY-NC-SA 3.0 IGO.

Dataset Government of Western Australia, A (2018) *Agriculture and food: potatoes*. Available at: https://www.agric.wa.gov.au/crops/horticulture/vegetables/potatoes (Accessed 23/01/2023).

Devaux, A., Goffart, J. P., Kromann, P., Andrade-Piedra, J., Polar, V., and Hareau, G. (2021). The potato of the future: opportunities and challenges in sustainable agri-food systems. *Potato Res.* 64, 681–720. doi: 10.1007/s11540-021-09501-4

Dong, N., Tang, M.-M., Zhang, W.-P., Bao, X.-G., Wang, Y., Christie, P., et al. (2018). Temporal differentiation of crop growth as one of the drivers of intercropping yield advantage. *Sci. Rep.* 8. doi: 10.1038/s41598-018-21414-w

Fu, X., Meng, Z., Wang, Z., Yin, X., and Wang, C. (2022). Dynamic potato identification and cleaning method based on rgb-d. *Engenharia AgrÃcola* 42. doi: 10.1590/1809-4430-eng.agric.v42n3e20220010/2022

Ghebreagziabiher, F. G., Griffin, D., Burke, J., and Gorman, M. (2022). Understanding the capacity of key actors and their role in the seed potato systems: the case of eritrea. *Outlook Agric.* 51, 260–269. doi: 10.1177/00307270221088330

Godwin, R., Wheeler, P., O'Dogherty, M., Watt, C., and Richards, T. (1999). Cumulative mass determination for yield maps of non-grain crops. *Comput. Electron. Agric.* 23, 85–101. doi: 10.1016/S0168-1699(99)00024-1

Goffart, J. P., Haverkort, A., Storey, M., Haase, N., Martin, M., Lebrun, P., et al. (2022). Potato production in northwestern europe (germany, france, the netherlands, united kingdom, belgium): characteristics, issues, challenges and opportunities. *Potato Res.* 65, 503–547. doi: 10.1007/s11540-021-09535-8

Gulati, S. (2019). Design and development of two row tractor operated potato combine harvester. *Potato J.* 46, 81–85.

Harahagazwe, D., Condori, B., Barreda, C., Bararyenya, A., Byarugaba, A. A., Kude, D. A., et al. (2018). How big is the potato (solanum tuberosum l.) yield gap in sub-saharan africa and why? a participatory approach. *Open Agric.* 3, 180–189. doi: 10.1515/opag-2018-0019

Hrushetsky, S., Yaropud, V., Duganets, V., Pryshliak, V., and Kurylo, V. (2019). Research of constructive and regulatory parameters of the assembly working parts for potato harvesting machines. *INMATEH Agric. Eng.* 59, 101–110. doi: 10.35633/INMATEH-59-11

Hrushetskyi, S., Yaropud, V., Kupchuk, I., and Semenyshena, R. (2021). The heap parts movement on the shareboard surface of the potato harvesting machine. *Bull. Transilvania Univ. Brasov.Forestry Wood Industry Agric. Food Engineering.* 14, 127–140. SubjectsTermNotLitGenreText - Ukraine.

Issa, I., Zhang, Z., El-kolaly, W., Yang, X., and Wang, H. (2020). Design, ansys analysis and performance evaluation of potato digger harvester. *Int. Agric. Eng. J.* 29, 60–73.

Jennings, S. A., Koehler, A.-K., Nicklin, K. J., Deva, C., Sait, S. M., and Challinor, A. J. (2020). Global potato yields increase under climate change with adaptation and co2 fertilisation. *Front. Sustain. Food Syst.* 4. doi: 10.3389/fsufs.2020.519324

Khakbazan, M., Mohr, R. M., Huang, J., Xie, R., Volkmar, K. M., Tomasiewicz, D. J., et al. (2019). Effects of crop rotation on energy use efficiency of irrigated potato with cereals, canola, and alfalfa over a 14-year period in manitoba, canada. *Soil Tillage Res.* 195, 104357. doi: 10.1016/j.still.2019.104357

McPhee, J. E., Antille, D. L., Tullberg, J. N., Doyle, R. B., and Boersma, M. (2020). Managing soil compaction–a choice of low-mass autonomous vehicles or controlled traffic? *Biosyst. Eng.* 195, 227–241. doi: 10.1016/j.biosystemseng.2020.05.006

Melito, S., D'Amelia, V., Garramone, R., Villano, C., and Carputo, D. (2017). Tuber yield and processing traits of potato advanced selections. *Adv. Hortic. Sci.* 31, 151–156. doi: 10.13128/ahs-21953

Mota-Rojas, D., Braghieri, A., Alvarez, A., Serrapica, F., RamÃrez-Bribiesca, J. E., Cruz Monterrosa, R., et al. (2020). The use of draught animals in rural labour. *Animals* 11, 2683. doi: 10.3390/ani11092683

Muneer, T., and Dowell, R. (2022). Potential for renewable energy-assisted harvesting of potatoes in Scotland. *Int. J. Low-Carbon Technol.* 17, 469–481. doi: 10.1093/ijlct/ctac012

Nasr, G., Rostom, M., Hussein, M., Farrag, A., and Morsy, M. (2018). Finite element analysis (fea) for potato crop harvester blade suitable for small holdings. *Bioscience Res.* 15, 2702–2710.

Nasr, G. E.-D. M., Rostom, M. N., Morsy, M., Hussein, M., Farrag, A. E.-F., and Morsy, M. F. A. (2019). Development of potato crop harvester suitable for smallholdings. *CIGR J.* 21.

On-Road Automated Driving Committee, O (2021). Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. doi: 10.4271/J3016\s\do5(2)02104

Poppa, L., Frerichs, L., and Niemöller, B. (2020). Validation of a particle simulation of potato tubers under harvesting-like conditions. *Landtechnik* 75, 196–205. doi: 10.15150/lt.2020.3245

Ramaswamy, N. (1998). Draught animal welfare. *Appl. Anim. Behav. Sci.* 59, 73–84. doi: 10.1016/S0168-1591(98)00122-1

Ritchie, H., Rosado, P., and Roser, M. (2023). "Agricultural production," in *Our world in data*. Available at: https://ourworldindata.org/agricultural-production.

Rosegrant, M., Sulser, T., Mason-D'Croz, D., Cenacchi, N., Pratt, A., Dunston, S., et al. (2017). Quantitative foresight modeling to inform the CGIAR research portfolio. *Tech. Rep.* 225.

Schneider, F., Part, F., Göbel, C., Langen, N., Gerhards, C., Kraus, G. F., et al. (2019). A methodological approach for the on-site quantification of food losses in primary production: Austrian and german case studies using the example of potato harvest. *Waste Manage.* 86, 106–113. doi: 10.1016/j.wasman.2019.01.020

Sibirev, A., Aksenov, A. G., Dorokhov, A. S., and Ponomarev, A. (2019). Comparative study of the force action of harvester work tools on potato tubers. *Res. Agric. Eng.* 65, 85–90. doi: 10.17221/96/2018-RAE

Soethoudt, H., and Castelein, B. (2021). Food loss-reducing intervention strategies for potato smallholders in kenya–a positive business case with reduced greenhouse gas emissions. *Agronomy* 11. doi: 10.3390/agronomy11091857

Spang, E. S., and Stevens, B. D. (2018). Estimating the blue water footprint of in-field crop losses: a case study of u.s. potato cultivation. *Sustainability* 10. doi: 10.3390/su10082854

Surdilovic, J., Praeger, U., Herold, B., Truppel, I., and Geyer, M. (2018). Impact characterization of agricultural products by fall trajectory simulation and measurement. *Comput. Electron. Agric.* 151, 460–468. doi: 10.1016/j.compag.2018.06.009

Tang, J., Wang, J., Fang, Q., Dayananda, B., Yu, Q., Zhao, P., et al. (2019). Identifying agronomic options for better potato production and conserving water resources in the agro-pastoral ecotone in north china. *Agric. For. Meteorology* 272-273, 91–101. doi: 10.1016/j.agrformet.2019.04.001

The Bureau of the Netherlands Agricultural Council in Buenos Aires, A (2008). Potato production in Argentina. *Tech. Rep.*

USDA (2022). United states department of agriculture: national agricultural statistics service, potatoes 2021 summary. *Tech. Rep.*

Vaitkevičienė, N., Jarienė, E., Kulaitienė, J., Danillčenko, H., Černiauskienė, J., Aleinikovienė, J., et al. (2020). Influence of agricultural management practices on the soil properties and mineral composition of potato tubers with different colored flesh. *Sustainability* 12. doi: 10.3390/su12219103

Vezirov, C., Atanasov, A., and Vladut, V. (2021). Calculation of field capacity and fuel consumption of mobile machinery with bunkers, tanks or other containers for agricultural goods. *INMATEH- AGRICULTURAL ENGINEERING* 63, 19–28. doi: 10.35633/inmateh-63-02

Waxman, A., Stark, J., Guenthner, J., Olsen, N., Thornton, M., and Novy, R. (2018). An economic analysis of the effects of harvest timing on yield, quality, and processing contract price for three potato varieties. *Am. J. Potato Res.* 95. doi: 10.1007/s12230-018-9663-z

Wei, Z., Li, H., Sun, C., Su, G., Wenzheng, L., and Li, X. (2019). Experiments and analysis of a conveying device for soil separation and clod-crushing for a potato harvester. *Appl. Eng. Agric.* 35, 987–996. doi: 10.13031/aea.13283

Wu, W., Yu, Q., You, L., Chen, K., Tang, H., and Liu, J. (2018). Global cropping intensity gaps: increasing food production without cropland expansion. *Land Use Policy* 76, 515–525. doi: 10.1016/j.landusepol.2018.02.032

Zhang, H., Xu, F., Wu, Y., Hu, H. H., and Dai, X. F. (2017). Progress of potato staple food research and industry development in china. *J. Integr. Agric.* 16, 2924–2932. doi: 10.1016/S2095-3119(17)61736-2

Appendix A1

APPENDIX TABLE 1 China's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	88,536,429.0	182,085.0	4,862,361.0	1,442,041,109
2018	90,321,442.0	189,722.0	4,760,724.0	1,448,928,199
2019	89,562,447.0	221,750.0	4,038,885.0	1,453,801,543
2020	92,852,722.1	198,588.0	4,675,654.0	1,456,928,486
2021	**94,362,175.0**	163,179.0	**5,782,738.0**	**1,457,934,562**
Mean	**91,127,043.0**	191,064.8	**4,824,072.4**	**1,451,926,779.8**
Std.	2,411,031.0	21,553.6	625,113.2	6,544,906.9
% Change	6.6	-10.4	18.9	1.1

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A2

APPENDIX TABLE 2 India's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	48,605,000.0	223,061.0	2,179,000.0	1,354,195,680
2018	51,310,000.0	239,542.0	2,142,000.0	1,369,003,306
2019	50,190,000.0	230,971.0	2,173,000.0	1,383,112,050
2020	48,562,000.0	236,772.0	2,051,000.0	1,396,387,127
2021	54,230,000.0	241,237.0	2,248,000.0	1,407,563,842
Mean	50,579,400.0	234,316.6	2,158,600.0	1,382,052,401.0
Std.	2,344,165.5	7,401.1	71,535.3	21,235,092.6
% Change	11.6	8.2	3.2	3.9

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A3

APPENDIX TABLE 3 Germany's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	11,720,000.0	467,864.0	250,500.0	82,624,374
2018	8,920,800.0	353,719.0	252,200.0	82,896,696
2019	10,602,200.0	390,361.0	271,600.0	83,148,141
2020	11,715,100.0	428,340.0	273,500.0	83,328,988
2021	11,312,100.0	437,944.0	258,300.0	83,408,554
Mean	10,854,040.0	415,645.6	261,220.0	83,081,350.6
Std.	1,172,814.0	44,326.5	10,762.8	**322,402.0**
% Change	-3.5	-6.4	3.1	1.0

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A4

APPENDIX TABLE 4 UK's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	6,218,000.0	425,890.0	146,000.0	66,064,804
2018	5,060,000.0	361,429.0	140,000.0	66,432,993
2019	5,307,000.0	368,542.0	144,000.0	66,778,659
2020	5,512,813.1	388,226.0	142,000.0	67,059,474
2021	5,306,719.8	387,352.0	137,000.0	67,281,039
Mean	5,480,906.6	386,287.8	141,800.0	66,723,393.8
Std.	442,174.1	25,030.5	3,492.9	486,067.2
% Change	-14.7	-9.1	-6.2	1.8

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A5

APPENDIX TABLE 5 Ukraine's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	22,208,220.0	167,837.0	1,323,200.0	44,657,257
2018	22,503,970.0	170,498.0	1,319,900.0	44,446,954
2019	20,269,190.0	154,869.0	1,308,800.0	44,211,094
2020	20,837,990.0	157,244.0	1,325,200.0	43,909,666
2021	21,356,320.0	166,430.0	1,283,200.0	43,531,422

(Continued)

APPENDIX TABLE 5 Continued

Year	Production	Yield	Area Harvested	Population
Mean	21,435,138.0	163,375.6	1,312,060.0	44,151,278.6
Std.	930,361.5	6,890.6	17,333.2	444,301.4
% Change	-3.8	-0.8	-3.0	-2.5

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A6

APPENDIX TABLE 6 USA's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	20,453,430.0	483,887.0	422,690.0	329,791,231
2018	20,421,560.0	497,274.0	410,670.0	332,140,037
2019	19,251,320.0	507,522.0	379,320.0	334,319,671
2020	19,051,790.0	**516,365.0**	368,960.0	335,942,003
2021	18,582,370.0	490,727.0	378,670.0	336,997,624
Mean	19,552,094.0	**499,155.0**	392,062.0	333,838,113.2
Std.	844,024.8	12,979.5	23,236.5	2,911,248.8
% Change	-9.2	1.4	-10.4	2.2

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A7

APPENDIX TABLE 7 Peru's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	4,776,294.0	153,875.0	310,400.0	31,605,486
2018	5,133,927.3	159,012.0	322,864.0	32,203,944
2019	5,389,231.0	162,730.0	331,177.0	32,824,861
2020	5,515,378.0	165,551.0	333,153.0	33,304,756
2021	5,661,443.0	171,245.0	330,604.0	33,715,471
Mean	5,295,254.7	162,482.6	325,639.6	32,730,903.6
Std.	348,828.9	**6,564.9**	9,377.0	844,357.6
% Change	18.5	**11.3**	6.5	6.7

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A8

APPENDIX TABLE 8 Egypt's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	4,841,040.0	277,724.0	174,311.0	101,789,386
2018	4,960,062.0	289,282.0	171,461.0	103,740,765
2019	5,200,563.0	292,876.0	177,569.0	105,618,671
2020	6,786,340.0	246,203.0	275,640.0	107,465,134
2021	6,902,817.0	262,758.0	262,706.0	109,262,178
Mean	5,738,164.4	273,768.6	212,337.4	105,575,226.8
Std.	1,019,114.6	19,381.0	52,129.5	2,952,333.2
% Change	**42.6**	-5.4	**50.7**	**7.3**

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

Appendix A9

APPENDIX TABLE 9 Australia's potato production (in tonnes), yield (in hg/ha), area harvested (in ha), and population for the years 2017–2021.

Year	Production	Yield	Area Harvested	Population
2017	1,105,194.2	389,534.0	28,372.0	24,590,334
2018	1,188,655.0	399,682.0	29,740.0	24,979,230
2019	1,225,273.6	378,022.0	32,413.0	25,357,170
2020	1,076,780.1	397,971.0	27,057.0	25,670,051
2021	1,267,638.6	403,372.0	31,426.0	25,921,089
Mean	1,172,708.3	393,716.2	29,801.6	25,303,574.8
Std.	**80,295.6**	10,133.2	**2,181.7**	532,074.5
% Change	14.7	3.6	10.8	5.4

The mean and standard deviation for each column across the four years is provided, as well as the percentage change between the years 2017 and 2021. Bold text indicates that is the largest value across all countries in the study, except standard deviation where it is the smallest value that is bold. Data extracted from Dataset FAO (2022a) and Dataset FAO (2022b).

TYPE Original Research
PUBLISHED 02 June 2023
DOI 10.3389/fpls.2023.1103276

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU), United States

REVIEWED BY
Uzair Aslam Bhatti,
Hainan University, China
Kai Huang,
Jiangsu Academy of Agricultural Sciences, China
Itsuki Kunita,
University of the Ryukyus, Japan

*CORRESPONDENCE
Hewei Meng
mhw_mac@126.com
Xiwen Luo
xwluo@scau.edu.cn

†These authors have contributed equally to this work

RECEIVED 20 November 2022
ACCEPTED 08 May 2023
PUBLISHED 02 June 2023

CITATION
Zhou X, Zou X, Tang W, Yan Z, Meng H and Luo X (2023) Unstructured road extraction and roadside fruit recognition in grape orchards based on a synchronous detection algorithm.
Front. Plant Sci. 14:1103276.
doi: 10.3389/fpls.2023.1103276

Unstructured road extraction and roadside fruit recognition in grape orchards based on a synchronous detection algorithm

Xinzhao Zhou[1,2], Xiangjun Zou[2,3], Wei Tang[2], Zhiwei Yan[2], Hewei Meng[1*†] and Xiwen Luo[1,4,5*†]

[1]College of Mechanical and Electrical Engineering, Shihezi University, Shihezi, China, [2]Foshan-Zhongke Innovation Research Institute of Intelligent Agriculture, Foshan, China, [3]Foshan Sino-tech Industrial Technology Research Institute, Foshan, China, [4]College of Engineering, South China Agricultural University, Guangzhou, China, [5]Guangdong Provincial Key Laboratory of Agricultural Artificial Intelligence (GDKL-AAI), Guangzhou, China

Accurate road extraction and recognition of roadside fruit in complex orchard environments are essential prerequisites for robotic fruit picking and walking behavioral decisions. In this study, a novel algorithm was proposed for unstructured road extraction and roadside fruit synchronous recognition, with wine grapes and nonstructural orchards as research objects. Initially, a preprocessing method tailored to field orchards was proposed to reduce the interference of adverse factors in the operating environment. The preprocessing method contained 4 parts: interception of regions of interest, bilateral filter, logarithmic space transformation and image enhancement based on the MSRCR algorithm. Subsequently, the analysis of the enhanced image enabled the optimization of the gray factor, and a road region extraction method based on dual-space fusion was proposed by color channel enhancement and gray factor optimization. Furthermore, the YOLO model suitable for grape cluster recognition in the wild environment was selected, and its parameters were optimized to enhance the recognition performance of the model for randomly distributed grapes. Finally, a fusion recognition framework was innovatively established, wherein the road extraction result was taken as input, and the optimized parameter YOLO model was utilized to identify roadside fruits, thus realizing synchronous road extraction and roadside fruit detection. Experimental results demonstrated that the proposed method based on the pretreatment could reduce the impact of interfering factors in complex orchard environments and enhance the quality of road extraction. Using the optimized YOLOv7 model, the precision, recall, mAP, and F1-score for roadside fruit cluster detection were 88.9%, 89.7%, 93.4%, and 89.3%, respectively, all of which were higher than those of the YOLOv5 model and were more suitable for roadside grape recognition. Compared to the identification results obtained by the grape detection algorithm alone, the proposed synchronous algorithm increased the number of fruit identifications by 23.84% and the detection speed by 14.33%. This research enhanced the perception ability of robots and provided a solid support for behavioral decision systems.

KEYWORDS

non-structural environment, machine vision, fruit harvesting robot, deep learning, roadside fruits detection

1 Introduction

Around the world, fruit plays an increasingly vital role in agriculture and economy. According to Food and Agriculture Organization of the United Nations (FAO), the total value of grape production has increased steadily since 1991, to more than $80 billion by 2020. Fruit harvesting is characterized by having limited work cycles and being labor intensive and time-consuming. With aging of the population and lack of rural labor force, labor costs have increased year by year (Wu et al., 2021; Li Y. J. et al., 2022). Under the influence of the COVID-19 pandemic and related policies (Aamir et al., 2021; Nawaz et al., 2021; Bhatti et al., 2022a; Bhatti et al., 2022b), the contradiction between labor demand and labor costs has become more prominent (Liang et al., 2021; Lin et al., 2022). This has had a negative impact on traditional hand-picking operations. With the deterioration of environmental issues (Bhatti et al., 2022; Galvan et al., 2022; Tang et al., 2023a), all the above factors pose a great challenge to China's fruit industry. With the rapid development of modern information technology and artificial intelligence technology, fruit harvesting robots and their related technologies have attracted extensive attention (Chen M. et al., 2020; Fu L. H. et al., 2020; Fu L. et al., 2020; Rysz and Mehta, 2021; Yang, 2021; Kang et al., 2022; Wang X. et al., 2022; Wu Z. et al., 2022).

As the basis of autonomous navigation, road detection is crucial to the precise operation of fruit harvesting robots and has become the focus of research in recent years (Ma et al., 2021; Sun et al., 2022). The main objective of road extraction is to extract the road regions from the background in a complex scene to lay the foundation for determining the navigation path. According to the characteristics of roads, they can be divided into two categories: structured roads and unstructured roads. Structured roads are standardized roads similar to urban roads and expressways, with clear lane markings, regular road edges, and distinct geometric features. Unstructured roads are those with irregular road edges, unclear road boundaries, no lane lines, and similar to orchards and rural areas. Compared to structured roads, unstructured roads have a more complex environmental background. For the most part, the surface of the unstructured road is mostly uneven, with a few random weeds. In contrast, the problem of unstructured road extraction is more complicated.

Research of road detection is usually divided into machine learning segmentation methods and traditional algorithms based on image features.

Road segmentation methods of machine learning are mainly divided into clustering (Zhang Z. Q. et al., 2022b), seed support vector machine (SVM; Liu et al., 2018), deep learning (Li et al., 2020), and other methods. Yang Z. et al. (2022) have proposed a visual navigation path extraction method based on neural network and pixel scanning. They introduced Segnet and Unet networks to improve the segmentation effect of orchard road condition information and background environment and adopted sliding filtering algorithm, a scanning method, and a weighted average method to fit the final navigation path. Lei et al. (2021) have combined improved SVM and two-dimensional lidar point cloud data to detect and identify unstructured roads. Wang E. et al. (2019) have realized road extraction of complex scenes by combining illumination invariant images and analyzing probability map and gradient information. Kim et al. (2020) have implemented automatic path detection in semi-structured orchards based on patch and CNN neural network methods. Alam et al. (2021) have implemented road extraction in structured and unstructured environments by combining multi-nearest neighbor classification and soft voting aggregation. Some scholars have also studied methods for road extraction in remote sensing based on machine learning methods (Xin et al., 2019; Chen et al., 2022; Guan et al., 2022; Yang M. et al., 2022). However, relevant research has been more on the basis of urban development analysis or traffic network monitoring and other fields, which are not applicable to picking robots. Machine learning usually does not require manual feature selection. However, this method requires specific network training and a large number of training sets and has certain limitations.

In the method based on image-feature analysis, some scholars use color, texture and other features to distinguish road and nonroad areas by establishing models and other methods. Zhou et al. (2021) have used the H component to extract the target path for the sky region. Chen J et al. (2020; 2021) have used an improved gray scale factor and the maximum interclass variance method (Otsu) method to extract gray scale images of soil and plants and realized segmentation of soil and plants in the greenhouse environment. Qi et al. (2019) have segmented the road region based on a graph-based manifold ranking approach and used binomial functions to fit the road region model, thus realizing road recognition in rural environment. Some scholars have also considered the vanishing point and other spatial structure features in the process of road extraction. Su et al. (2019) have adopted the Dijkstra method combined with single-line lidar to realize road extraction on the basis of the constraints of pre-vanishing points of illumination-invariant images. Phung et al. (2016) have realized pedestrian lane detection based on an improved vanishing point estimation method combined with geometry and color features. However, the detection of vanishing points is time-consuming and mostly applied to structured road detection (Xu et al., 2018), which is not suitable for dealing with unstructured roads.

To realize autonomous walking and precise operation of fruit harvesting robots in orchard environments and aiming at the uncertainty of random distribution of roadside fruit and road complexity, it is necessary to deeply study the problem of synchronous road extraction and fruit identification. This study enables robot perception of barrier-free road areas and roadside fruit distribution in the current environment and can provide an inferential basis for robot global operational behavior decisions in complex orchard environments. Moreover, this study can lay the foundation for the joint control and operation of navigation and picking based on visual guidance in the panoramic environment of wild orchards. However, current approaches have only focused on road extraction, without considering the roadside fruit detection. In this case, the autonomous decision-making function of the robot cannot perform reasonable picking responses and navigation path planning based on the random distribution of fruits along the road, which is detrimental to the intelligent global continuous operation of the robot.

In terms of object detection, neural networks have been widely used in the field of smart agriculture (Khaki and Wang, 2019; Tang et al., 2020; Feng et al., 2022; Fu et al., 2022), and You Only Look Once (YOLO), as one of the fastest target detection models at present, has also been rapidly developed (Ye et al., 2020; Ning et al., 2022; Wang X. Y. et al., 2022). For example, due to the excellent performance of the YOLOv5 model in terms of accuracy and running time, it has been greatly valued by scholars in the research of crop growth-morphology recognition (Lv et al., 2022; Rong et al., 2022; Wu F. et al., 2022), detection and positioning (Fang et al., 2022; Jintasuttisak et al., 2022; Li G. et al., 2022; Wang H. et al., 2022), tracking counting (Lyu et al., 2022; She et al., 2022; Zang et al., 2022), and pest recognition (Li S. et al., 2022; Qi et al., 2022; Zhang et al., 2022).

Given the importance of detecting and locating fruit for picking robots, researchers have explored various fruit detection and location methods based on neural networks (Wang C. et al., 2019; Ge et al., 2022; Jia et al., 2022; Zhou et al., 2022; Tang et al., 2023c). To improve the operational efficiency and success rate of picking robots, researchers have gradually shifted their focus to picking-path planning algorithms and picking decision systems based on fruit detection (Lin et al., 2021; Wang Y. et al., 2022). For example, Xu et al. (2022) have proposed an efficient combined multipoint picking scheme for tea buds through a greedy algorithm and ant colony algorithm, which improved picking efficiency and overall picking success rate. Ning et al. (2022) proposed a method for recognition and planning robotic picking sequences for sweet peppers based on an improved YOLOV4 model and a principle of anticollision picking within picking clusters. The method can accurately detects sweet peppers, reduces collision damage, and improves picking efficiency in high-density orchard environments. Rong et al. (2022) have proposed an obstacle avoidance method that combines end-effector grasping-pose adjustment and harvesting sequence planning based on a custom manipulator. Experiments show that the method significantly reduced the impact of collision on the picking and improved the success rate of tomato picking. Although some progress has been made in the study of local target detection and picking planning, there have been few reports on the synchronization information perception needs of picking robots to autonomously pick and walk.

To implement the behavioral decision-making function of the picker robot to walk autonomously and pick accurately throughout the entire process in a large-area orchard environment, road extraction and roadside fruit identification should first be implemented in the current working scenario. Currently, many algorithms only focus on road extraction and ignore the fruit distribution along the road, which leads to the serious problem that picking robots are not robust enough to adapt to the changing orchard environments. Therefore, a road extraction and roadside fruit synchronous recognition algorithm based on unstructured road was proposed in this study. The main contributions of this study were as follows:

(1) Currently, numerous studies have focused on extracting unstructured roads without considering the synchronous recognition of roadside fruits, which is detrimental to improving the ability of picking robots to obtain environmental information. Motivated by the need for cooperative behavioral decision-making in fruit picking robots, this study proposed a framework for unstructured road extraction and synchronous recognition of roadside fruit. This framework can effectively improve the ability of fruit-picking robots to extract crucial information from the picking environment and lay a foundation for multitask parallel processing, thereby enabling cooperative behavioral decision-making among fruit-picking robots.

(2) Due to the randomness and complexity of orchard environments, the results of road extraction directly from raw images were not very accurate and contained a large number of misidentified regions. An image preprocessing method based on image enhancement and filtering preprocessing was designed here which reduced the influence of interference existing in the complex orchard environment. Simultaneously, this approach enhanced the precision of road extraction results and was of great importance for improving the quality of road extraction.

(3) The irregular road edges of unstructured roads and various interference factors in orchards considerably impacted the stability of the road extraction results. To address this issue, analyses of orchard images were conducted to optimize the gray factor and enhance its adaptability to field orchards. A two-space fusion unstructured road extraction algorithm was proposed, which used color channel enhancement and gray factor optimization and demonstrated great adaptability to interference factors, such as shadow, uneven lighting, grapevine on the side of the road, and strong contrast between light and shade in the field complex environment.

(4) A fusion algorithm based on the road extraction algorithm and roadside fruit detection algorithm was constructed. Based on the detection requirements for roadside grapes in wide-field environments, YOLO models were compared, selected, and optimized for their parameters. Subsequently, the three functions of image preprocessing, road extraction, and roadside grape recognition were integrated to construct a synchronous recognition algorithm, allowing for the simultaneous extraction of road and other key information during the fruit-picking process. The proposed algorithm provided information for decision-making and reasoning of collaborative behavior of key parts of the robot, so as to improve robot adaptability to randomly distributed fruit.

This study will lay a foundation for the construction of robot behavior decision control system, and it is of great significance for improving the intelligence, accuracy, and stability of robot field autonomous work.

The rest of this report is organized as follows. Section 2 introduced the materials and data. Section 3 explained the structure and implementation of the algorithm. Section 4 presented the experimental results and comparative discussion. Finally, Section 5 summarized the study and plans for future work.

2 Materials and data acquisition

2.1 Experimental platform for wine grape picking and moving

This study was based on the wine grape visual mobile picking robot that was independently designed and developed. The overall layout of the test platform is shown in Figure 1A. The test platform was battery powered to operate in the orchard. The length and width of the platform were 1.065 and 0.7 m, respectively, and the maximum climbing capacity was 30°. Two cameras were installed on the end-effector of the platform as picking camera and navigation camera, separately.

The control process of the experimental platform was divided into three main parts (Figure 1B). The first part of the control system was to construct algorithms for unstructured road extraction and roadside fruit synchronization recognition based on the collected datasets A and B. Then, the industrial personal computer (IPC) implemented the algorithm-based key information acquisition, recognition, and behavioral decisions. The second part of the control system was to use the IPC to control the navigation camera for orchard road extraction and roadside fruit recognition. By recognizing the distinction between unstructured roads and chaotic backgrounds, as well as the classification and recognition of roadside grapes and grapevines, it provided a judgment basis for the IPC to distinguish the presence of roadside fruit and lay the foundation for behavioral decisions. Based on the above information, the third part of the control system extracted the navigation path of the orchard and judged the presence of fruit in the current roadside area. If there were fruit on the roadside, the controller controlled the tracked vehicle to approach the fruit area of the roadside fruit tree and fed the information to the robotic arm and another set of stereo camera for precise positioning (the picking camera for short) for picking operations. Using the picking camera, fruit could be re-identified and accurately positioned to achieve fruit picking in complex environments. The work of this study mainly implemented the first part of the control system.

2.2 Experimental subjects

Wine grapes and non-structural orchards were taken as experiment subjects in this research. Wine grape fruit are clustered in shape and usually purple at maturity, with a clear color difference from leaves. The planting mode is usually in rows with a certain row spacing. As the fruit distribution and planting patterns of wine grapes are similar to other row-grown crops, such as tomato and dragon fruit, the results of this study are expected to be extendable to other types of fruit.

2.3 Image acquisition

In August 2022, experimental images were obtained from Xinyu Winehouse (Bohu County, Bazhou, Xinjiang). The device used for dataset sampling was an OPPO R11 mobile phone with a 20-megapixel rear camera. All images were taken under natural daylight conditions without artificial light sources and saved in Joint Photographic Expert Group (jpg) format with image size 4608×2128 pixels.

The collected images were divided into datasets A and B. The original images of vineyards in dataset A included roads and vines. As the algorithm proposed in this study was intended to provide a

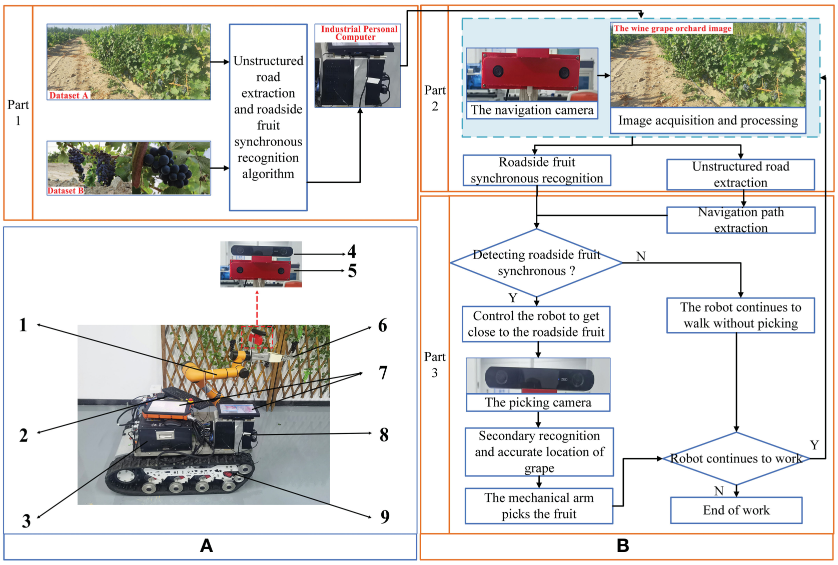

Part 1
Dataset A
Dataset B
Unstructured road extraction and roadside fruit synchronous recognition algorithm
Industrial Personal Computer
Part 2
The navigation camera
The wine grape orchard image
Image acquisition and processing
Roadside fruit synchronous recognition
Unstructured road extraction
Navigation path extraction
Part 3
Detecting roadside fruit synchronous ?
N
Y
Control the robot to get close to the roadside fruit
The robot continues to walk without picking
The picking camera
Secondary recognition and accurate location of grape
Robot continues to work
Y
N
The mechanical arm picks the fruit
End of work
1
2
3
4
5
6
7
8
9
A
B

FIGURE 1
Overall layout and control flow of test platform. **(A)** Overall layout of test platform. Mechanical arm (AUBO-i5, AUBO), 1; Battery, 2; Controller, 3; Camera for picking (HBV-1714, Huiber Vision Technology Co., Ltd), 4; Camera for navigation (ZED 2, Stereolabs), 5; End-effector, 6; Human Machine Interaction, 7; Industrial Personal Computer (IPC), 8; and Track car, 9. **(B)** Control flow of the test platform.

basis for behavioral decisions of grape-picking robots, the focus was on the region of unstructured road and distribution of fruit in a unilateral grape row. Therefore, during the collection process of dataset A, the camera observation direction was biased to the right of the road center line (Figure 2A). A total of 337 typical orchard images were selected, in which the roads in the grape orchard environment had features of shadow and irregular road edges (Figure 2B). Dataset B was composed of 1081 valid images showing wine grape clusters, including grape samples in numerous cases, with images of grapes in front and backlight (Figures 2C, D).

2.4 Image datasets

To simulate the vision system of the picking robot, valid grape and orchard image samples were collected under different conditions of illumination, weather, sampling distance, and differing severity of fruit adhesion and occlusion, forming datasets A and B.

Dataset A consisted mainly of orchard images with uneven lighting, with multiple weeds, with large shadows, in different weather conditions, and with different light and shade contrasts (Figure 3A).

The natural images of grapes (dataset B) mainly included images of single cluster grape, multiple clusters grape, slightly-adhered grape, severely-adhered grapes, front and back illumination, small string grapes, large cluster grapes, and grapes on a sunny day, on a cloudy day, and in shadow as well as grapes at different sampling distances. Their representative images are shown in Figure 3B.

Datasets A and B were challenging considering the effects of complex background, light levels, shadows, randomly distributed fruits, weeds, and different levels of fruit occlusion. Images of grapes and vineyards in a typical complex environment were contained in dataset A and B (Figure 3).

Dataset A was only employed for testing the performance of unstructured road extraction and the overall algorithm, with 100 images in this dataset randomly selected as the test set for algorithms in this study. To improve algorithm efficiency, the processing image size of the algorithm was set to 1024×473 pixels.

Dataset B was used for training and validation of the fruit model on the YOLOv7 roadside. Under LabelImg (https://github.com/tzutalin/labelImg), grapes in images were manually annotated as rectangles with the label "fruit," which then saved annotation files in "txt" format. Among them, the whole image set was randomly divided into training and validation sets with a ratio of 9 to 1.

3 Methodology and algorithm description

In this study, the algorithm content was mainly divided into two parts: First, the road in the unstructured orchard environment was extracted. Second, taking the road extraction results as input, roadside fruit were identified through YOLOv7 to realize the synchronous information extraction of the road extraction and roadside fruit detection. The algorithm process of this study is shown in Figure 4.

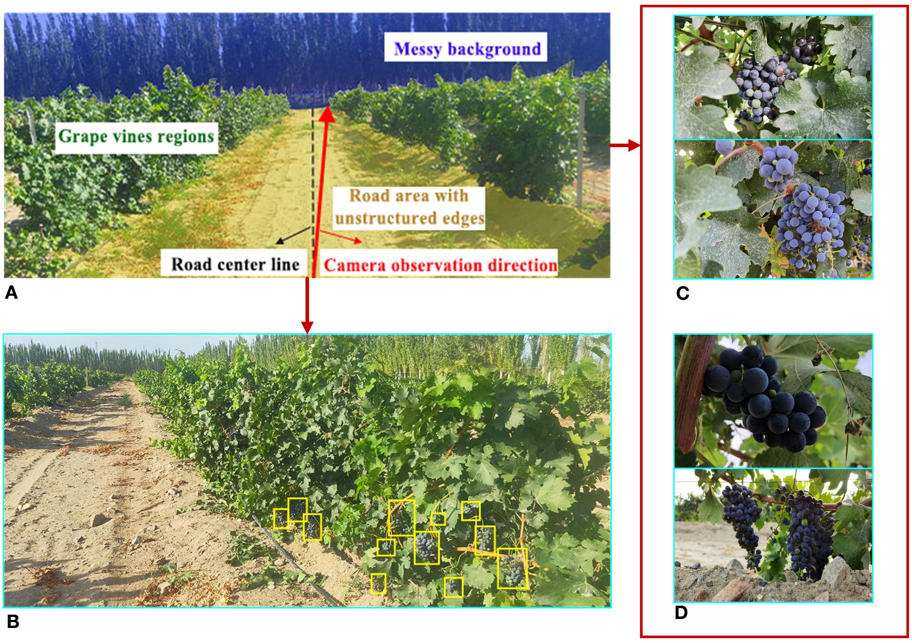

Messy background
Grape vines regions
Road area with unstructured edges
Road center line
Camera observation direction
A
B
C
D

FIGURE 2
Schematic diagram of the acquisition process of test images. **(A)** The camera observation direction. **(B)** Example image of the wine vineyard. **(C)** Examples of frontlight images. **(D)** Examples of backlight images.

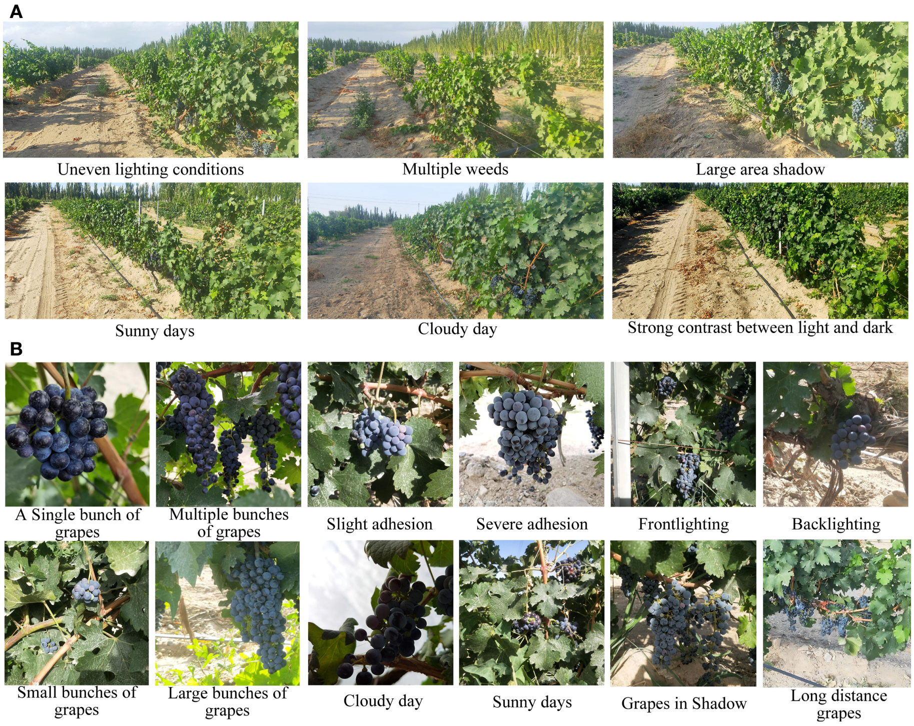


FIGURE 3
Natural images of vineyards and wine grape clusters. **(A)** Natural images of vineyards. **(B)** Natural images of wine grape clusters.

3.1 Image preprocessing

During image acquisition in the orchard, it was inevitable to be disturbed by external environmental noise, such as uneven light and dust, which made the image details unclear and led to road extraction errors. Therefore, this study preprocessed the images in dataset A, which was of great significance for improving the quality of road segmentation (Wang et al., 2018; Zhang P. et al., 2022). The image preprocessing method proposed in this study consisted of five steps, with the processing procedure and image quality enhancement results illustrated in Figure 5. Further details can be found in Sections 3.3.1 - 3.1.5.

3.1.1 Interception of regions of interest

The images in dataset A were composed of sky, road, grapes, and messy background, among which the sky and messy background were mainly distributed at the top of an image. In the image processing process, if the entire image captured by the camera was merely taken as the research object, a substantial amount of computation would be required and a significant amount of interference inevitably occurs, which will reduce road extraction accuracy. To this end, only the regions of interest (ROI) of the image was extracted for subsequent processing. After a number of experiments, it was found that the appropriate ROI was at the lower 5/6 position of the image (Figure 5B). This ROI selection not only significantly reduced the calculation volume, but also ensured the accuracy and reliability of unstructured road extraction.

3.1.2 Bilateral filter

A bilateral filter can smooth the image while maintaining edge details (Routray et al., 2020). To enhance and improve contrast between the foreground and background of the road to facilitate subsequent segmentation, a bilateral filter was used to process the present images. To reduce the influence of minor areas, such as vines, fruits, vine gaps, and cavities in subsequent segmentation, the key parameters of the bilateral filter (Liu et al., 2017) in this study were set to: diameter d of the pixel domain was 60, standard deviation of spatial domain 120, and standard deviation of intensity domain 60 (Figure 5C).

3.1.3 Logarithmic space transformation

To enhance the details in the shadowed regions and provide images with enhanced details and uniform brightness for subsequent MSRCR processing, a logarithmic transformation of the V-component in hue, saturation, and value (HSV) space was used here to expand the low gray values and compress high gray values in this channel (Figure 5D). The standard form was

$$S = c * \log(1 + L) \tag{1}$$

where S is the correction image, L the source image, and c the gain adjustment parameter, which was set to 1.

3.1.4 Image enhancement based on the MSRCR algorithm

After the above processing and observing the image under RGB color space, the altering influence of illumination was found not to

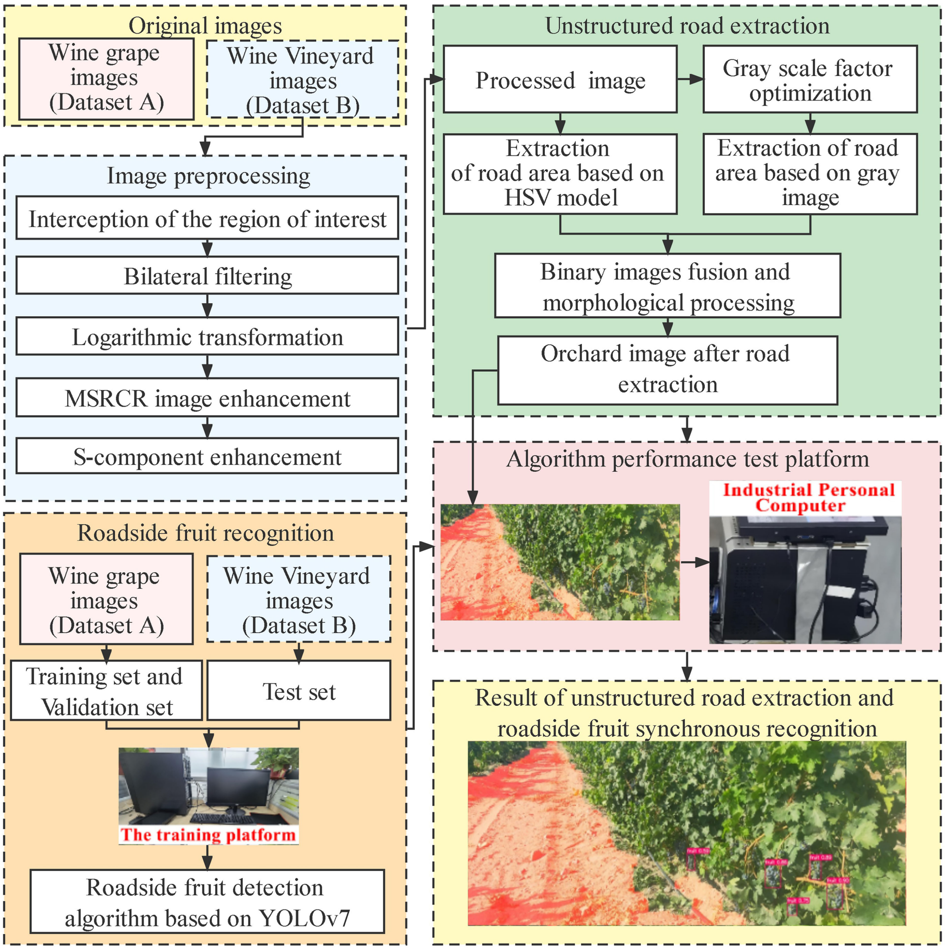

Original images
Wine grape images (Dataset A)
Wine Vineyard images (Dataset B)
Image preprocessing
Interception of the region of interest
Bilateral filtering
Logarithmic transformation
MSRCR image enhancement
S-component enhancement
Roadside fruit recognition
Wine grape images (Dataset A)
Wine Vineyard images (Dataset B)
Training set and Validation set
Test set
The training platform
Roadside fruit detection algorithm based on YOLOv7
Unstructured road extraction
Processed image
Gray scale factor optimization
Extraction of road area based on HSV model
Extraction of road area based on gray image
Binary images fusion and morphological processing
Orchard image after road extraction
Algorithm performance test platform
Industrial Personal Computer
Result of unstructured road extraction and roadside fruit synchronous recognition

FIGURE 4
Flow chart of the entire algorithm.

be entirely eliminated. Therefore, the MSRCR algorithm was selected for image correction and enhancement here to obtain realistic images with reduced illumination effects. The resulting Equations 2–4 were expressed as:

$$R_{MSRCR}(x,y) = C_i(x,y)R_{MSR}(x,y) \tag{2}$$

$$R_{MSR}(x,y) = \sum_1^N \varphi_n \left\{ \log I_i(x,y) - \sum_1^N \log[F(x,y) * I_i(x,y)] \right\} \tag{3}$$

$$C_i(x,y) = \beta\left\{ \log[\alpha I_i(x,y)] - \log\left[\sum_1^N (I_i(x,y)) \right] \right\} \tag{4}$$

The optimal functional form of MSRCR is shown in Equation 5, expressed as:

$$R_{MSRCR}(x,y) = G\left\{ C_i(x,y)\left[\log I_i(x,y) - \sum_1^N \log\left(I_i(x,y) * F(x,y)\right) \right] + b \right\} \tag{5}$$

where $I_i(x,y)$ is the color component image corresponding to each color channel, $F(x,y)$ the Gaussian filter function, and $C_i(x,y)$ the color restoration factor of the i^{th} color channel, φ_n the weight, and N the number of spectral channel, where $\sum_1^N \varphi_n = 1$, β a gain constant, and α the strength of nonlinearity, G and b the final gain and offset values, respectively. The parameters of MSRCR in this study were configured according to the reference (Jobson et al., 1997).

3.1.5 S-component enhancement

To enrich color information, this study adjusted the saturation channel S to enhance image quality, with the formulas described by Equations 6–7 (Huang et al., 2022), expressed as:

$$S_{opt} = \alpha_s * T * S_{ori} \tag{6}$$

$$T = \frac{mean(R,G,B) + Max(R,G,B) + Min(R,G,B)}{mean(R,G,B)} \tag{7}$$

where S_{opt} represents the enhanced saturation channel, S_{ori} the original saturation channel of S, $mean(R,G,B)$, $Max(R,G,B)$, and $Min(R,G,B)$ the average, maximum, and minimum values of pixels corresponding to R, G, and B color channels, respectively, and α_s and T the gain coefficients of the saturation channel, which control the enhancement degree of S channel image.

Qualitative and quantitative evaluation is significant for the evaluation of image quality. In the qualitative evaluation, the quality

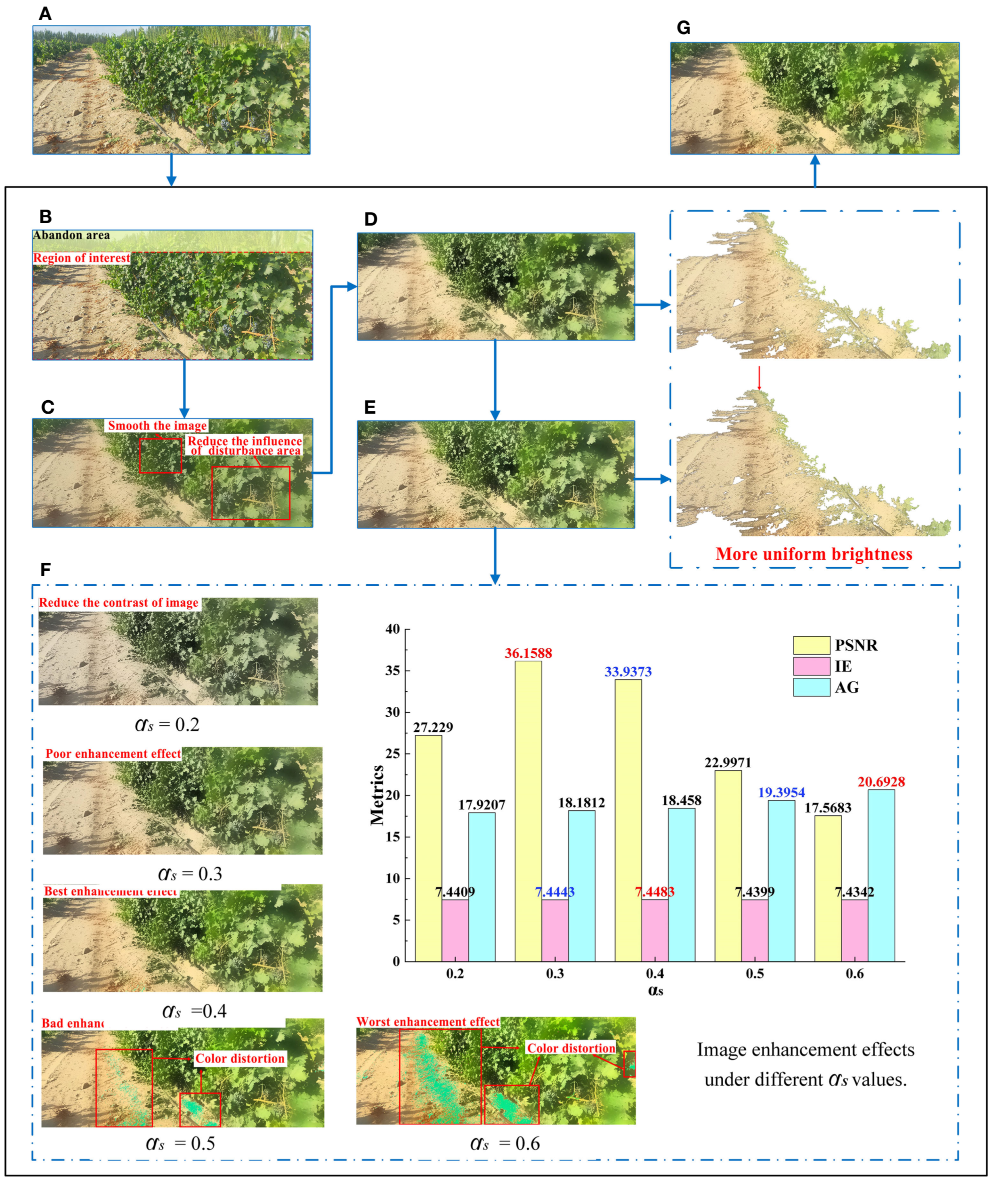


FIGURE 5
Process and results of image preprocessing algorithm. **(A)** Original image. **(B)** Region of interest. **(C)** Bilateral filtering result. **(D)** Log space transformation. **(E)** Image enhancement based on the MSRCR algorithm. **(F)** S-component enhancement. **(G)** Results of image preprocessing algorithm.

of the enhanced image was evaluated in color, contrast, and detail. By comparing the gain effect at different values, it was observed that, if the value of α_s was too high or low, the image contrast was reduced or saturation too strong, which affected the visual effect of the image. When α_s = 0.2, the contrast of the image was low, resulting in poor overall visual effect. When α_s was greater than 0.5, there was significant color distortion despite the high contrast of images, resulting in partial loss of detail in the image. When α_s = 0.3, although the tone of the image was better maintained, the enhancement effect was not obvious compared with the image without S-component enhancement. When α_s = 0.4, the contrast of the image was improved significantly without obvious color distortion and the visual effect was the best.

In the quantitative evaluation, this paper evaluated the performance the processing results by three metrics Peak signal-to-noise ratio (PSNR, He et al., 2015), information entropy value (IE, Wang et al., 2021) and average gradient (AG, Zhang X. et al., 2022). PSNR has been widely used for measuring attributes like texture details enhancement, details preservation and contrast enhancement. A higher PSNR generally indicates that the processed image is of higher quality (Gupta and Tiwari, 2019). IE is mainly an objective evaluation index that measures how much information an image contains. The enormous IE value indicates that the enhancement image contains more image information. AG represents the degree of change in the gray value of the image, and is one of the criteria for judging the processing of image details and clarity. The large AG value indicates that the enhancement image contains more gradient information and detailed texture. The image enhancement quality evaluation parameters under different values of α_s were shown in Figure 5F, where the optimal parameter values were marked in red and the second highest parameter values were highlighted in blue. Figure illustrated that the value of AG increased as the value of α_s increased, indicating that the sharpness of the image was also enhanced progressively. However, color distortion occurred when α_s was set to 0.5 or 0.6. Therefore, this paper eliminated the enhanced images with these two parameters and only discussed the image enhancement results with low α_s value(α_s< 0.5). Moreover, the highest value and the second highest value of PSNR and IE were mainly concentrated in the results of α_s =0.3 and α_s =0.4, which indicates that under the above two parameter settings, the images had a good performance in terms of image information, contrast enhancement and detail preservation. Furthermore, for α_s=0.4, both the IE and AG values were higher than those for α_s=0.3, while the PSNR was slightly lower than the latter. Therefore, based on the qualitative evaluation results and the requirements of enhanced images in terms of clarity, information content, picture details and contrast, α_s was finally set at 0.4 in this study.

3.2 Unstructured road extraction

In this section, unstructured road extraction was achieved by fusing two parts, including the segmented road region after removing green regions from the HSV space and road region based on improved gray factor.

3.2.1 Road extraction based on color enhancement and HSV color space

HSV color space is composed of hue (H), saturation (S) and luminance (V) channels. As HSV color space is more consistent with human color perception, it has been widely used in multifield research based on machine vision, such as medicine (Singh, 2020), agriculture (Liao et al., 2022), and chemical industries (Safarik et al., 2019). Therefore, the HSV color space was used here to extract road regions.

First, the enhanced and optimized RGB image was converted into an HSV image and the threshold range (H_{min}, H_{max}), (S_{min}, S_{max}), and (V_{min}, V_{max}) of each channel set to binarize the image. This completed the constraint and extraction of the green area, so as to distinguish the road area from the plant area (vines, weeds, and background trees). Based on Exploratory data analysis (EDA) and empirical values (Guo et al., 2013; Peng et al., 2013; Camizuli and Carranza, 2018), the HSV ranges were set at (35,77),(43,255),and (46,255), respectively (Figure 6A). As can be seen from the image, although the road extraction was relatively complete, the main

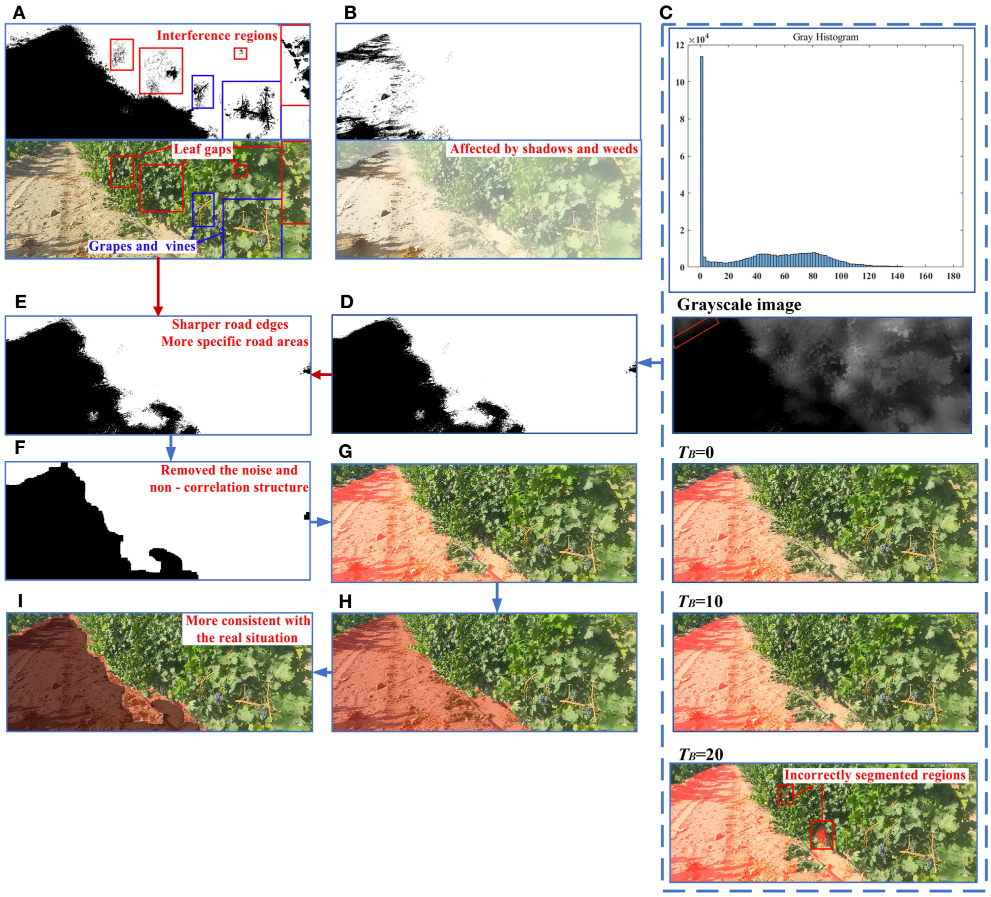


FIGURE 6
Process and results of road extraction. **(A)** Road extraction results in HSV space. **(B)** Road extraction results based on ExG Gray factor. **(C)** Road segmentation effect under different T_B. **(D)** Road extraction results based on optimized grayscale factor. **(E)** Fused binary image. **(F)** Morphological processing result. **(G)** Final extraction result. **(H)** Manual image segmentation. **(I)** Results of comparison between proposed algorithm and real situation.

constraint in the HSV space was the green region, such that there were still interference regions due to grapes and their vines, leaf gaps, and other factors in extraction results.

3.2.2 Road extraction based on gray factor optimization

Taking advantage of the significant color difference between different objects in the image, numerous researchers have realized object segmentation by examining different gray weights, such as excessive red plant index (ExR, Meyer et al., 1999), excessive green index (ExG, Woebbecke et al., 1995), and normalized difference index (NDI, Woebbecke et al., 1993). The preprocessed image mainly contained four areas: grape vines regions, soil areas, background, and shadow areas. Therefore, through manual segmentation of the above regions and obtaining the average values of R, G, and B in different regions, the gray factor was improved by a heuristic method based on the excessive green index (ExG). The optimized gray factor and its binarized image acquisition formula were expressed in Equations 8 and 9 as

$$gray(x,y) = 1.84G(x,y) - B(x,y) - R(x,y) \tag{8}$$

$$f(x,y) = \begin{cases} 0 \cdots \cdots \cdots 1.84G(x,y) - B(x,y) - R(x,y) \le T_B \\ 255 \cdots \cdots \cdots 1.84G(x,y) - B(x,y) - R(x,y) > T_B \end{cases} \tag{9}$$

where gray(x,y) is the optimized gray level factor, f(x,y) the binarized image, and G(x,y), B(x,y), and R(x,y) as the green, blue, and red components of the color range, respectively. And T_B is the binarization threshold.

Based on the optimized grayscale factor, the grayscale image and the grayscale histogram of the enhanced image after the S-component were plotted in Figure 6C. As can be seen from the gray histogram, most pixels in the image had a gray value of 0, corresponding to the majority of black road areas in the gray map. However, as shown by the red area in the grayscale image, a few pixels in the road area had gray values that were not zero. Therefore, the rationality of the binarization threshold T_B directly affected the integrity of the road segmentation. To determine the optimal binarization threshold, a comparative experiment was conducted in this paper, using the threshold value T_B as the independent variable and the road segmentation result as the dependent variable. The initial value of the binarization threshold was set to 0, and different binarization thresholds were used to segment the road. The threshold of binarization was increased by 10 for each group until the segmentation result incorrectly included the vine area on the side of the road.

When T_B= 0, the segmentation result indicated a significantly smaller road area than the actual road. With T_B set at 10, the vast majority of road area was accurately extracted from the segmentation results. However, when T_B was increased to 20, while the extracted road area was more comprehensive, there were numerous incorrectly extracted sections. Consequently, for this article, T_B was established at 10, the road extraction results were shown in Figure 6D.

The extraction method of unoptimized gray factor based on ExG was found to be affected by shadows and weeds, resulting in a large number of noise points and holes in the treatment results, and only extracted a small number of road regions (Figure 6B). Thus, the extracted area was significantly smaller than the real value. On the other hand, the improved gray factor method exhibited superior segmentation results for the grapevine area on the road and its surroundings, showing great advantages in the accuracy and integrity of road segmentation (as depicted in Figure 6D). The above results indicated that compared with the unimproved gray factor, the improved gray factor method was more adaptable to unfavorable environmental conditions such as shadows and lighting in the field.

3.2.3 Binary images fusion and morphological processing

By fusing the above two binarized images in Figures 6A, D, most of the disturbances (Figure 6E) were eliminated and road edges constrained. The fused results were more consistent with the real situation.

However, there were various tiny noises and irregularly-shaped edges in the fused binary image. Therefore, morphological processing was performed on fused binary images to remove non-correlated structures (Figures 6F, G).

The road edge extracted by this algorithm was found to be in line with the trend of the real road and fundamentally eliminated the vine area on the side of the road (Figure 6I). This reduced the interference of light, shadow, weeds, and dead branches to road extraction, with high extraction integrity and good comprehensive performance.

3.2.4 Performance evaluation indexes

In this study, the number of ROI image pixels (NRP) and the ratio between the wrongly extracted pixels and the number of ROI image pixels (RBP) were used as evaluation indices for verifying the performance of the road extraction algorithm. And the calculation equations of this evaluation index expressed in Equation 10 as

$$RBP = \frac{NWP}{NRP} \times 100\,\% \tag{10}$$

where NWP is the number of wrongly extracted pixels by the algorithm.

3.3 Roadside fruit detection based on YOLOv7

3.3.1 Characteristics of the YOLOv7 network structure

As the latest version of the YOLO series (Wang C. et al., 2022), YOLOv7 has improved the existing model in many ways. First, it offers extended efficient layer aggregation networks (E-ELAN) based on ELAN structure, which can guide different computing blocks to learn more different features and enhance the learning ability of the model on the basis of maintaining the original gradient path. Then, a compound model scaling method based on the cascade model has been proposed to ensure the initial characteristics and optimal structure of the model, which efficiently utilizes parameters and computation. Meanwhile, several trainable bag-of-freebies methods have been designed for real-time object detection, which significantly improves detection accuracy without increasing inference cost. Based on the above improvements, YOLOv7 shows great advantages in terms of speed and accuracy over other detection algorithms. Its network architecture is shown in Figure 7.

Based on the performance advantages of the YOLOv7 and YOLOv5 models, both models were adopted in this research to detect roadside fruits. The results were compared to identify the roadside grape detection model that is better suited for large-field environments. The selected model's feasibility and detection performance were then further verified for roadside fruit recognition.

3.3.2 Network training and parameter optimization

The experiment was conducted on a Windows 10 operating system, with the Python framework, YOLOv7, and YOLOv5 environments built in the Anaconda environment. The program was written in Python 3.9 and CUDA Ver. 11.7. In terms of hardware, the processor is an Intel (R) Core (R) i5-1240F CPU@ 2.5 GHz, the dominant frequency is 2.5 GHz, internal storage 32.0 GB, and graphics card an NVIDIA GeForce RTX 3060.

Due to the complex orchard environment, directly applying the default parameters of YOLO model to the roadside fruit recognition model results in poor detection results. To adapt to fruit recognition in complex field scenarios, the learning rate parameter of the YOLO model was chosen as described in this study. The initial value of the learning rate was set to 0.01 and the model was trained with different learning rates. The learning rate of each group was reduced by 0.002, respectively, until the optimal parameters were detected and chosen. By comparison, it was found that when the learning rate was larger than 0.002, the loss curves for object detection in the results suffered from severe oscillations, poor convergence or nonconvergence. Thus, the learning rate of the wine grape orchard recognition model was set to 0.002.

The training and verification sets were input into the network for training, with a batch size of 16 and 150 epochs, respectively (Table 1).

3.3.3 Model evaluation

In this study, precision (P), recall (R), F1-score, and *mAP* were used as the evaluation indices of roadside fruit detection

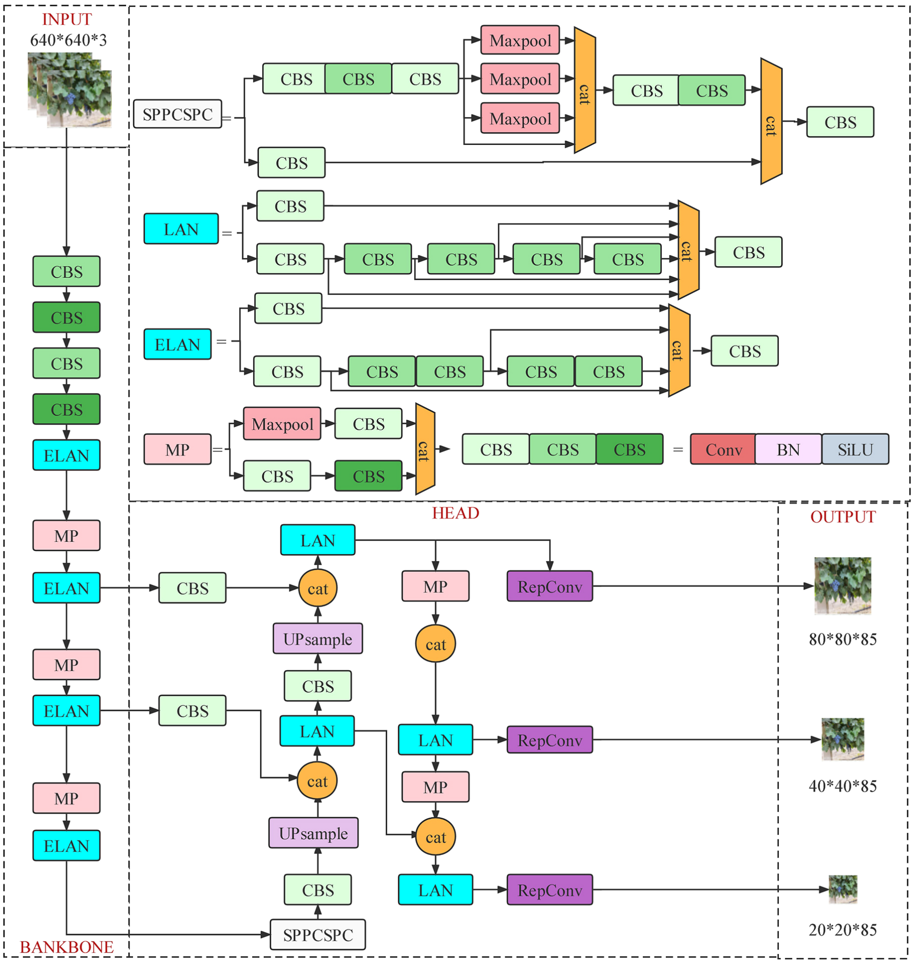


FIGURE 7
Network structure of YOLOv7.

TABLE 1 YOLO basic parameters.

Parameters of model	Value
Input image resolution	640×640
Learning rate	0.002
Momentum	0.937
Optimizer weight decay	0.0005
Warmup momentum	0.8
Batch size	16
Training epochs	150

performance and the calculation equations of each evaluation index expressed in Equations 11–14 as:

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$R = \frac{TP}{TP + FN} \tag{12}$$

$$F1 - score = \frac{2 \times P \times R}{(P + R)} \tag{13}$$

$$mAP = \frac{\sum_{1}^{c} AP(c)}{c} \tag{14}$$

where *TP*, *FP*, and *FN* correspond to true positives (there is a grape bunch in the image and the algorithm predicts it correctly), false positives (there are no grapes in the image, but the algorithm detects it), and false negatives (the algorithm failed to detect a bunch of grapes which are actually in the image), respectively, and *C* the number of detection classes. As only one kind of fruit was identified in this study, *C* = 1.

4 Experiments and discussion

By achieving synchronous recognition of road extraction and roadside fruit, this algorithm can considerably improve the ability of robots to perceive critical information in the orchard environments and lay the foundation for autonomous walking and picking decisions based on machine vision. Therefore, the performance of this algorithm was extremely critical for the robot's picking rate, navigation path extraction accuracy, and reliability of the decision system in subsequent researches. At the same time, this study served as a reference for other research in the same field.

In this section, the performance of image enhancement, road extraction, roadside fruit recognition, and overall fusion algorithm were verified and discussed.

4.1 Road extraction effects and ablation tests

4.1.1 Road extraction results and analysis

To validate the image segmentation effect of the proposed road extraction algorithm, the results obtained by fused segmentation were compared with those obtained by the conventional color image method. This study adopted two traditional algorithms: a method based on S component and Otsu and another based on the Excess Green index (ExG) and Otsu. At the same time, 25 images with pavement shadows, strong illumination variations, and grapevines with different degrees of color were selected as test samples to verify the adaptability of the above algorithms to complex environments.

The results of multiple sample images were compared, in which samples were original color images and other images obtained by segmentation methods. The comparative findings for partial sample images were illustrated in Figures 8A–F, while the comparative results for additional images could be found in the Supplementary Material. Figures 1–3 depict the image samples with the lowest NWP value in the outcomes of Methods C, D, and E, while Figures 4–6 depict the image samples with the highest NWP value in the outcomes of Methods C, D, and E, respectively.

For simplicity, the proposed algorithm was abbreviated as "Method C", the method based on S component and Otsu was abbreviated as "Method D", and the method based on EXG and Otsu was abbreviated as "Method E".

In the qualitative evaluation, the quality of different segmentation methods was assessed based on the completeness of road segmentation and the distribution of error areas. Due to the complexity of the field orchard, the primary environmental factors that influence the precision of road segmentation outcomes include the grapevine area, shadowed road area (Li et al., 2018), roadside unevenly colored area, and high contrast between light and dark areas (Tang et al., 2023b). As depicted in Figure 8A, strong lighting caused the grapevine areas on the roadside to exhibit characteristics such as uneven light and shade and varying color tones. This led to a significant contrast between light and shade in the grapevine areas on both sides of the road. Additionally, different lighting angles resulted in distinct areas of shadow on the road surface, thereby increasing the complexity involved in segmenting orchard roads.

Observationally, it was found that the extraction results of methods D and E (Figures 8D, E) suffered from problems, such as the large area errors in identification. Although the extraction results were of great completeness, the results also contained a large number of incorrect regions (Figure 8F). By comprehensive comparison, the road obtained by the Method C was found to be the closest to the real situation and had the best segmentation effect among all considered methods.

To further analyze the adaptability of the above method to complex vineyard scenarios, the extraction results of the proposed algorithm were compared with real roads (Figures 8F, G). Based on Figure 8F, it can be observed that the error areas of methods D and E were primarily concentrated in the grapevine area on the side of the road.

Method D was found to be sensitive to changes in brightness, shade, and color uniformity of the grapevine region in the image, which resulted in changes in the error area of the segmentation result (Figure 8D). Due to the unpredictable and random nature of illumination in field environments, it was difficult to guarantee the accuracy and stability of the segmentation results achieved through method D.

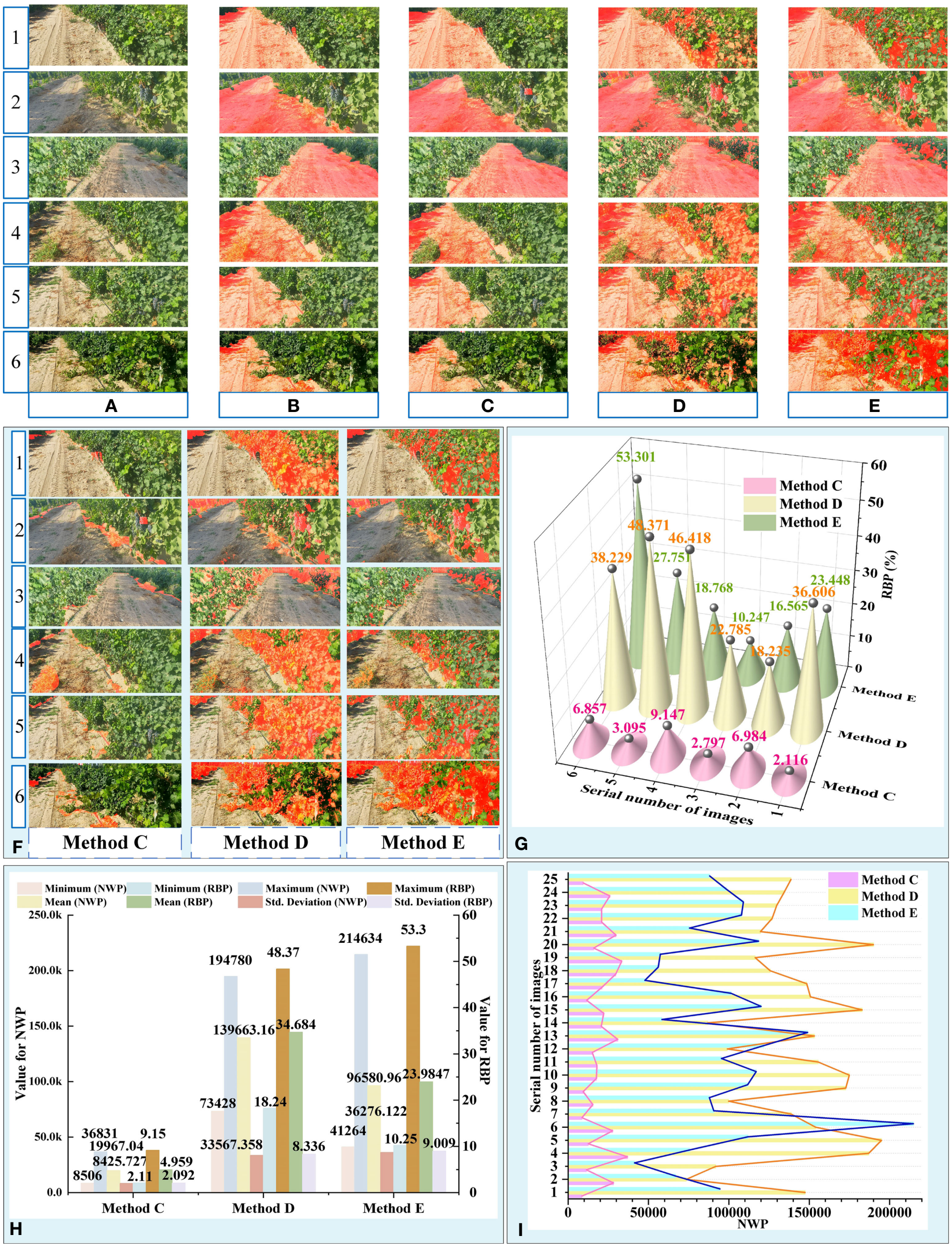


FIGURE 8
Results and analysis of different segmentation methods. **(A)** Original images. **(B)** Manual image segmentation. **(C)** Proposed algorithm. **(D)** Method based on S component and Otsu. **(E)** Method based on EXG and Otsu. **(F)** Error area results extracted by different methods. **(G)** RBP values of partial images obtained by different methods. **(H)** Descriptive Statistics for NWP and RBP. **(I)** NWP values of 25 images obtained by different methods.

The primary error source of method E was the grapevine area with strong contrast between light and shade, with the dark part of it being incorrectly identified as the road area. This greatly reduced the accuracy of the segmentation result. When the area of the dark region of the grapevine on the side of the road was small, the error rate of this algorithm decreased significantly. However, when faced with areas that had uneven colors on the side of the road, the error area of the segmentation result achieved through this method was significantly smaller than that of method D.

Conversely, Method C adapted to the aforementioned unfavorable factors, resulting in a smaller error in the segmented area, more stable road extraction performance, and the most reliable segmentation results among the three methods. Combined with the above analysis, the influence degree of unfavorable factors on the accuracy and reliability of the results obtained through different methods was comprehensively evaluated, as presented in Table 2.

To quantitatively evaluate the extraction performance of the above methods, NWP and RBP were taken as indices to achieve a road extraction performance evaluation of different algorithms, where NRP = 402,668 (Table 2; Figures 8H, I). To determine the differences in road extraction performance among the three

TABLE 2 Analysis of the influence degree of adverse factors on algorithms and extraction results.

Degree of influence of adverse factors on algorithm accuracy				
Adverse environmental factors	Impact degree			
	Method C		Method D	Method E
Grapevine area	Minor		Severity	Severity
Shadowed road area	Minor		Minor	Minor
Roadside unevenly colored area	Minor		Severity	Medium
Strong contrast between light & dark	Minor		Severity	Severity
Methods	Descriptive Statistics for NWP			
	Minimum	Maximum	Mean	Std. Deviation
Method C	8506	36831	19967.040	8425.727
Method D	73428	194780	139663.16	33567.358
Method E	41264	214634	96580.960	36276.122
Pairwise Comparisons of Methods (NWP)				
Sig	Method C vs Method D		Method C vs Method E	Method D vs Method E
	<0.001		<0.001	0.008
Methods	Descriptive Statistics for RBP/%			
	Minimum	Maximum	Mean	Std. Deviation
Method C	2.11	9.15	4.959	2.092
Method D	18.24	48.37	34.684	8.336
Method E	10.25	53.30	23.9847	9.009
Pairwise Comparisons of Methods (RBP)				
Sig	Method C vs Method D		Method C vs Method E	Method D vs Method E
	<0.001		<0.001	0.008

methods, the non-parametric Kruskal-Wallis test was conducted across the three groups using SPSS software version 27 (IBM Corporation). The significance level was set at 0.05. The null hypothesis in this test is that there is no difference between the three methods in terms of the distribution of NWP and RBP. In fact, for this test, the *Sig* values less than 0.05 indicate a significant difference between the groups.

According to the descriptive statistical table of NWP, Method C exhibited a generally low overall level of NWP value (Figure 8H). Comparing the mean value of NWP across the three methods, it was found that the mean value of NWP for Method C accounted for only 14.3% and 20.67% of the mean value of NWP for Methods D and E, respectively. Furthermore, the maximum and minimum values of NWP for Method C were one order of magnitude smaller than those of Methods D and E. Additionally, the standard deviation of NWP value for Method C was significantly lower than that of Methods D and E, indicating that the road extraction performance of Method C was more stable in the face of variable field interference factors. This observation was also validated in Figure 8I, which illustrates that the NWP of Method C exhibits a relatively mild fluctuation in comparison to the other two methods. Moreover, the Kruskal-Wallis test results showed that the NWP values of Methods C and D (*Sig*<0.01), Methods C and E (*Sig*<0.01) and Methods D and E (*Sig* = 0.008)were statistically significant difference. Furthermore, it was confirmed that there were substantial differences in the accuracy of road extraction among the three methods.

Similar results were obtained from the descriptive statistical table of RBP. Method C demonstrated favorable outcomes in terms of the maximum, minimum, mean, and standard deviation of RBP. Thereinto, Method C had an RBP of no more than 9.15%, whereas Method D had an RBP of no more than 48.37%, and Method E had a notably high RBP of 53.30%. The above data suggested that the wrongly identified pixels in the road extraction results of Method C only constituted a small portion of the current image. Compared to the other two methods, Method C was found to deliver better segmentation results for road recognition in the field environment and exhibited greater adaptability to the complex environmental interference factors in the field orchard.

4.1.2 Ablation test

To verify the improvement of the image enhancement algorithm on the overall performance of the road extraction algorithm, an ablation experiment was conducted. The comparative findings for a selection of sample images were illustrated in Figure 9, while the comparative results for additional images could be found in the Supplementary Material. For simplicity, the proposed algorithm without preprocessing was abbreviated as “Method F”.

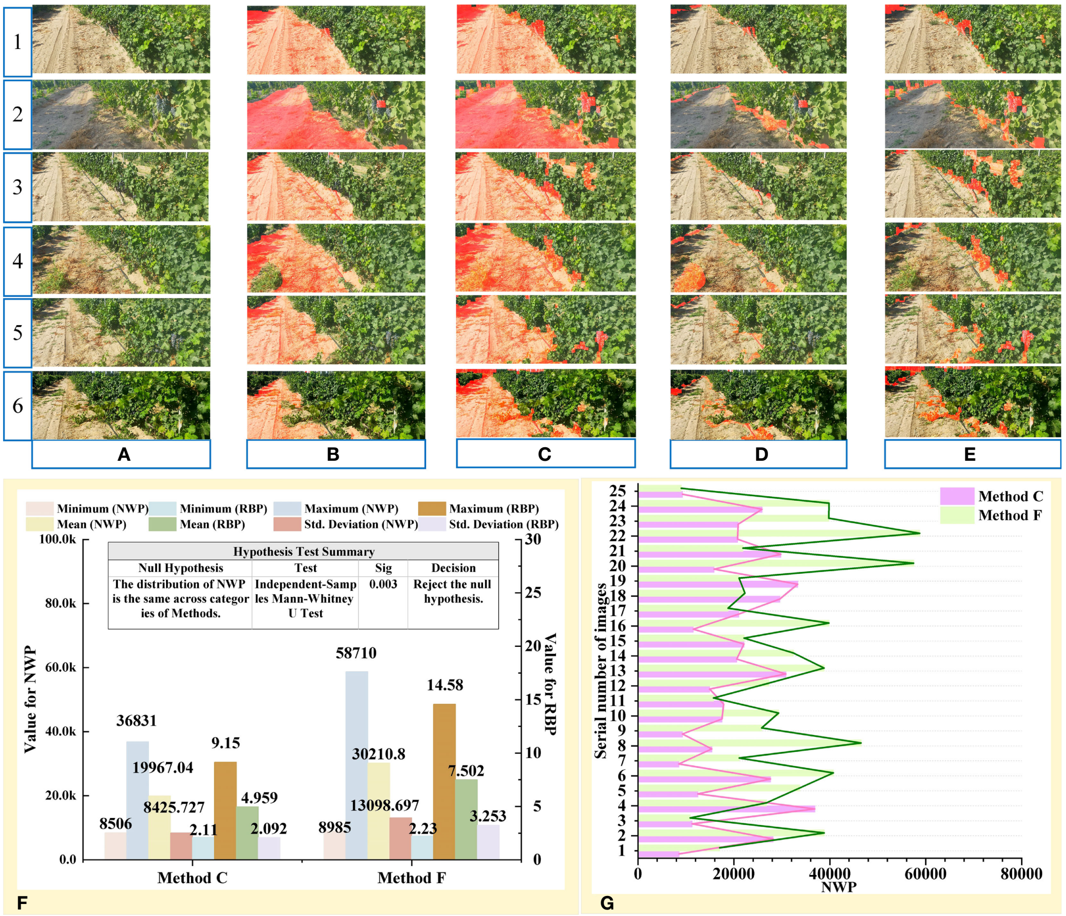
Ablation experiments were conducted on the proposed preprocessing method. The extraction results after pretreatment were shown in Figure 9B and the algorithm results without pretreatment were shown in Figure 9C. By comparing the two extraction results, the latter extraction results were found to contain a large number of error regions, such as dark grape vines area, grapes, and other objects on the roadside (Figure 9E). This phenomenon was confirmed by NWP descriptive statistics (Figure 9F).

Based on Figures 9F, G, it can be observed that the majority of segmentation results obtained using Method F had a higher NWP value compared to those obtained using Method C. However, a few image processing results showed an opposite result. The reason for this phenomenon can be attributed to the fact that after image preprocessing, the segmentation result of Method C had more stringent restrictions on green areas, resulting in the removal of a large area of weeds from the road in the segmentation results, thereby increasing the NWP value (4th row of Figure 9C).

After image preprocessing, the accuracy of the algorithm was significantly improved at the cost of a small amount of completeness, which reduced the impact of interference regions, such as road shadows, dark fruits, branches, leaves, and gaps in segmentation accuracy. Meanwhile, the Method C also suppressed the interference of noncurrent road areas on the extracted results and significantly reduced the number of misdetected pixels (3th row of Figures 9C, D).

Differences in road extraction performance between the above methods were determined using the non-parametric Mann-Whitney U test. The significance level was set at 0.05. The null hypothesis in this test is that there is no difference between the methods in terms of the distribution of NWP. And the *Sig* values less than 0.05 mean a significant difference between the groups. The Mann-Whitney U test result showed that the NWP values of Methods C and F (*Sig*= 0.03) were statistically significant difference (Figure 9F).

In conclusion, image preprocessing played a crucial role in enhancing the accuracy and reliability of road segmentation results.



Hypothesis Test Summary			
Null Hypothesis	Test	Sig	Decision
The distribution of NWP is the same across categor ies of Methods.	Independent-Samp les Mann-Whitney U Test	0.003	Reject the null hypothesis.

FIGURE 9

Ablation test results from the proposed preprocessing method. **(A)** Original images. **(B)** Proposed algorithm. **(C)** Proposed algorithm without preprocessing. **(D)** Error area results extracted by proposed algorithm. **(E)** Error area results extracted by proposed algorithm without preprocessing. **(F)** Descriptive Statistics and significance analysis result. **(G)** NWP values of 25 images obtained by methods C and F.

4.2 Comparison between YOLOv5 and YOLOv7

Target location is an important task in target detection and is normally represented by the coordinate position of the bounding box. The models in this paper used CIoU (Lv et al., 2022) loss to calculate the boundary frame position loss, which was calculated as follows:

$$L_{box} = 1 - IOU + \frac{\rho^2(A,B)}{c_d^2} + \alpha v \tag{15}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^g}{h^g} - \arctan\frac{w^p}{h^p}\right)^2 \tag{16}$$

$$\alpha = \frac{v}{(1 + IOU) + v} \tag{17}$$

Where $\rho^2(A,B)$ is the Euclidean distance of the center points between predicted box and ground truth box, c_d is the diagonal distance of the smallest rectangle containing predicted box and ground truth box, α is the weight function, and v is the function that measures the consistency of the aspect ratio. w^g and h^g are the width and height of the ground truth box, while w^p and h^p are the width and height of the prediction box.

The confidence loss function is used to measure the difference between the confidence score predicted by the model and the actual label. In this paper, the confidence loss function was calculated using a binary cross-entropy loss function (BCELoss, Zhao et al., 2023), and its formula was as follows:

$$L_{conf} = -\frac{1}{N}\sum_{n=1}^{N}[y_n \times \log x_n + (1 - y_n) \times \log(1 - x_n)] \tag{18}$$

Where y_n denotes the true category, which generally takes the value of 0 or 1, x_n denotes the prediction confidence or target probability obtained by the Sigmoid function, and N is the number of positive and negative samples.

After training, the loss function value curves for the training and validation sets of the two YOLO models were obtained, including the loss values of the detection box and detection object (Figures 10A, B). In Figure 10A, "BOX" and "Val BOX" represented the box loss of the training set and validation set, respectively. In Figure 10B, "Objectness" and "Val Objectness" represented the confidence loss of the training set and validation set, respectively. As shown in Figure 10A, B, it can be observed that the change trend of the loss curves for both models was similar. In particular, it was observed that the values of box and object detection losses for the two YOLO models decreased sharply during training batches 0 to 20, after which the rate of decline slowed down. The sample distribution ratio of model training set and verification set is shown in Figure 10D. In addition, the box and the object detection loss values of the YOLOv7 algorithm on the training set were smaller than that of the YOLOv5 algorithm after 150 training epochs. The box detection loss value of YOLOv7 finally stabilized around 0.029 and object detection loss value eventually stabilized around 0.012.

In addition, although the loss value of box detection in the validation set was slightly higher than that of YOLOv5, the loss value of object detection in the validation set of YOLOv5 showed a trend of fluctuation and rise after 50 training batches. Meanwhile, the loss value of YOLOv7 algorithm decreased steadily and finally the loss value tended to stabilize around 0.0025.

Under the same dataset B, the performance indices of YOLOv7 were better than those of YOLOv5 (Figure 10C). The P, R, *mAP*, and F1-scores of YOLOv7 were 88.9, 89.7, 93.4, and 89.3%,

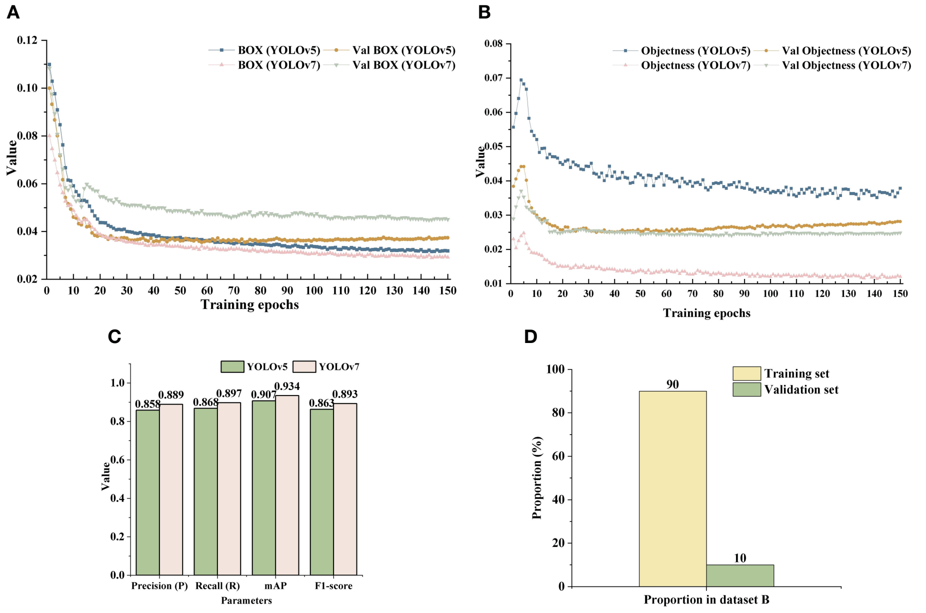


FIGURE 10
Loss curves and detection results of the two YOLO models. **(A)** Box loss value curve of YOLOv5 and YOLOv7 model. **(B)** Confidence loss function value curve of YOLOv7 model. **(C)** Detection results of YOLOv5 and YOLOv 7 on dataset B. **(D)** Training set and verification set introduction.

respectively, which were 3.1, 2.9, 2.7, and 3% higher than from YOLOv5.

Although the number of YOLOv7 targets detected in some images was less than that of YOLOv5, the overall accuracy of the former was higher than that of the latter (Figures 11A–C). Moreover, in global images, YOLOv5 showed the phenomenon of grape cluster misidentification (Figure 11, last row). Algorithm detection confidence was the main evaluation metric in this study. In summary, YOLOv7 was able to better perform the task of detecting clusters of grapes in orchards and, hence,YOLOv7 was used to identify grapes on the roadside.

The confidence level of grape clusters recognition results tested by YOLOv7 on dataset B was mostly above 0.8, while it was mostly above 0.5 on dataset A. There were two reasons for this phenomenon. The first was that the grape clusters were smaller on dataset A than those in the training set and the second that dataset A contained a large number of backgrounds, such as sky, trees, and roads, and the overall complexity of the image far greater than that of the training set.

4.3 Recognition effects of the synchronous detection algorithm

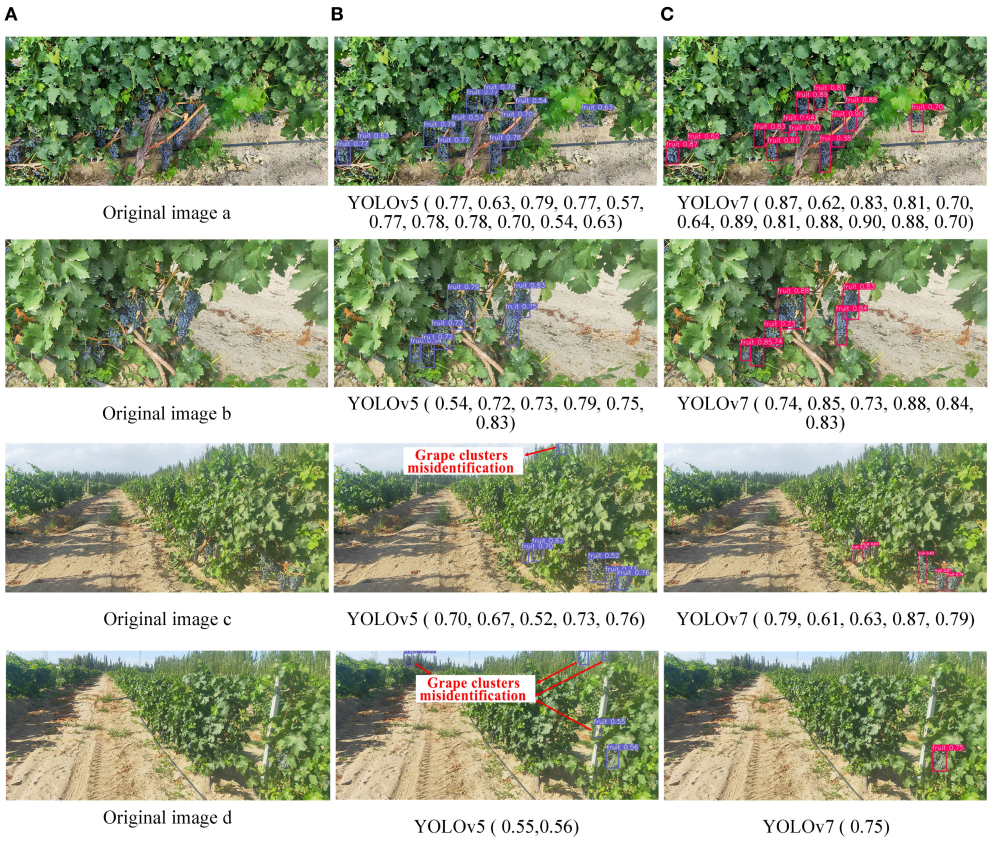
Furthermore, in order to evaluate the overall detection performance of the synchronous detection algorithm proposed in this paper (Figure 4), simultaneous recognition of the road and roadside fruit was conducted (Figure 12A).

The results demonstrate that the algorithm was able to effectively segment the road area despite the complex outdoor environment, and accurately recognize the grapes on the side of the road. This provides valuable information for the intelligent decision-making and control of the robot during subsequent walking and fruit picking operations, and enhances the robot's ability to identify crucial targets within a complex environment.

Furthermore, the synchronous recognition algorithm demonstrated better effectiveness in roadside grape recognition. To validate the positive impact of image preprocessing and road segmentation in the synchronous recognition algorithm on the recognition performance of road test grapes, the images with and without above aforementioned steps were identified using yolov7 model (Figure 12B). The results revealed that, under identical circumstances, the former approach detected more clusters of grapes on the road side.

To further demonstrate the superiority of the proposed synchronous recognition algorithm in roadside grape detection, 66 images from dataset B were used to detect grape clusters. The number of recognized fruits, recognition time and the promotion ratio (P_r) were taken as evaluation parameters. The P_r was calculated by the following formula.

$$P_r = \frac{V_w - V_n}{V_n} \tag{19}$$


A
B
C
Grape clusters misidentification
Grape clusters misidentification

Original image a

YOLOv5 (0.77, 0.63, 0.79, 0.77, 0.57, 0.77, 0.78, 0.78, 0.70, 0.54, 0.63)

YOLOv7 (0.87, 0.62, 0.83, 0.81, 0.70, 0.64, 0.89, 0.81, 0.88, 0.90, 0.88, 0.70)

Original image b

YOLOv5 (0.54, 0.72, 0.73, 0.79, 0.75, 0.83)

YOLOv7 (0.74, 0.85, 0.73, 0.88, 0.84, 0.83)

Original image c

YOLOv5 (0.70, 0.67, 0.52, 0.73, 0.76)

YOLOv7 (0.79, 0.61, 0.63, 0.87, 0.79)

Original image d

YOLOv5 (0.55,0.56)

YOLOv7 (0.75)

FIGURE 11
Comparison of partial detection results. **(A)** Original images. **(B)** Identification results of YOLOv5 model. **(C)** Identification results of YOLOv7 model.

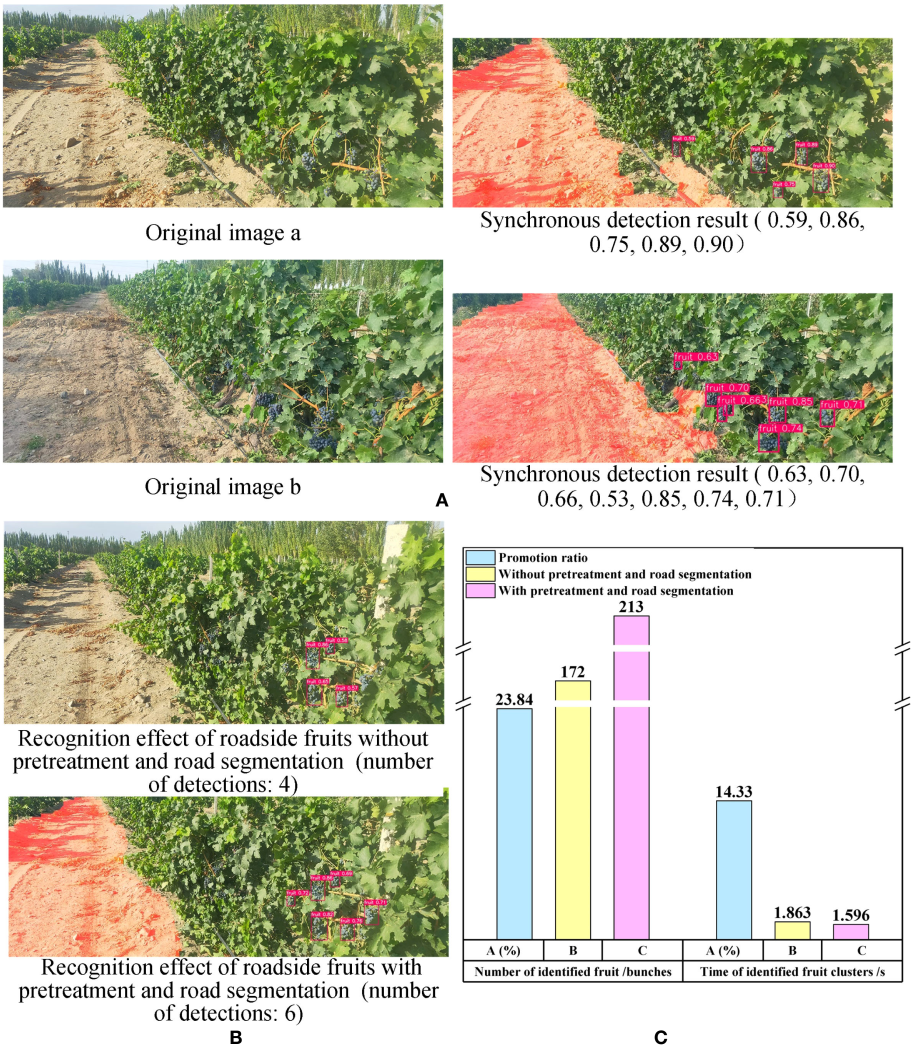


FIGURE 12
Recognition results of overall algorithm and comparison results between proposed algorithm and single YOLOv7 model. **(A)** The overall synchronous detection algorithm recognition results. **(B)** Comparison results of roadside grape clusters identification results between proposed synchronous detection algorithm and single YOLOv7 model. **(C)** Performance comparison between overall synchronous detection algorithm and the single YOLOv7 model.

Here, V_w represents the evaluation parameters obtained through image calculation based on image preprocessing and road segmentation, while V_n represents the evaluation parameters obtained without image preprocessing and road segmentation.

The number of recognized grape clusters in the former was 41 more than that in the latter, representing a 23.84% increase. Additionally, the recognition speed of the former was 0.267 seconds faster than that of the latter, resulting in a speed increase of 14.33%. The results indicated that the images with pre-processing and road segmentation were able to identify more grape clusters and at a faster detection speed compared to the images without pre-processing and road segmentation (refer to Figure 12C). This finding provided evidence that the synchronous recognition algorithm proposed in this paper outperforms using YOLOv7 alone for identifying roadside grapes under the same scenario.

The reasons for the above phenomena were as follows: First, due to the extraction and preprocessing of the ROI in the overall algorithm, a large number of backgrounds, such as sky and trees, were eliminated, which improved the proportion of grape cluster pixels in the whole image. In addition, after extracting the road in the image, the interference of the road area on fruit cluster recognition was reduced and grape features more pronounced, which was beneficial for detecting fruit clusters on the roadside.

4.4 Discussion

Although the unstructured road extraction and roadside fruit synchronous recognition algorithm proposed in this study had good performance, it also had some limitations (Figures 13A–C). First, it was difficult to distinguish the adhesive road areas between different rows during road extraction. For example, when the death of grape plants leads to a large area of vacancy on the road side, the road regions of images consisted of two parts: the road part of the robot's current row

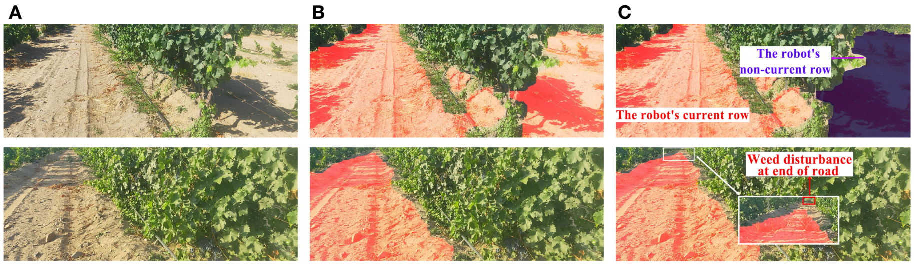


FIGURE 13
Adverse Conditions. **(A)** Original image. **(B)** Result of road extraction. **(C)** Analysis of adverse factors.

and road part of the non-current row (Figure 13C). In this case, it was difficult for the proposed algorithm to distinguish the correct region from the wrong one. At the same time, when there were a large area of weeds near the end of the road with a width of more than 1/2 of the width of the road, the completeness of the extracted results was reduced. Future research will consider optimization algorithms and add constraints to improve result accuracy.

In addition, in the process of roadside fruit string identification, there was still a situation of missing grape-cluster detection. Future research will further optimize and improve the network structure for the problems of missing fruit string detection and low confidence of some detection target results.

5 Conclusions

In this study, an algorithm for unstructured road extraction and roadside fruit synchronous recognition in a complex orchard environment was developed to address the above issues. The main conclusions could be obtained as follows:

(1) An unstructured road extraction and roadside fruit synchronous recognition framework was constructed for achieving simultaneous road extraction and roadside fruit detection, which effectively improved the ability of fruit picking robots to extract key information from the picking environment. The algorithm also provided information for decision-making and reasoning of collaborative behavior of key parts of the robot, which improved the adaptability of the robot to randomly distributed fruit.

(2) Based on the analysis of the orchard images, an image enhancement preprocessing method was proposed to reduce the interference of road shadows, dark fruits, branches, and leaves as well as gaps in segmentation results. The method also suppressed the influence of noncurrent road areas on extraction results to a certain extent, which improved result accuracy and reliability.

(3) By enhancing the color channel and optimizing the grayscale factor, the dual spatial fusion road extraction was achieved. Experimental results showed that, compared with the extraction method based on S component and Otsu and extraction method based on EXG and Otsu, the proposed algorithm showed greater adaptability to adverse conditions, such as uneven illumination and road shadows under the background of complex orchards. The proposed road extraction algorithm also largely avoided the problems of missing extraction of real road areas and identification of large area errors, which had the best segmentation effect.

(4) The YOLOv7 and YOLOv5 algorithms, optimized with grape cluster target data, were used to identify roadside grape clusters. The optimized YOLOv7 model achieved a precision of 88.9%, recall of 89.7%, mAP of 93.4%, and F1-score of 89.3%, all of which were higher than those obtained from the YOLOv5 model. Based on this comparison, the YOLOv7 with optimized parameters was found to be more suitable for roadside grape recognition in wide-field views.

(5) The proposed fusion algorithm took the road extraction results as input and then identified fruit strings on the road side. The performance of the proposed fusion algorithm was superior to only using the YOLOv7 model. Compared with the single YOLOv7 model, the number of grape string detections and detection speed of the fusion algorithm were increased by 23.84% and 14.33%.

Although the new algorithm has achieved satisfactory results, there remains some room for progress. First, due to the similarity between different lines of the roads, the algorithm in this case had difficulty in segmenting the cohesive road area between different lines. At the same time, the completeness of the extraction results was reduced when there were a large area of weeds with a width ratio of 1/2 near the end of the road.

Future work will focus on network structure optimization to improve the accuracy and speed of road extraction and roadside fruit detection algorithms. Constraints between road zones will also been studied to enable the identification and segmentation of road zones between different lines. Furthermore, environment-aware robot behavioral decision control systems will be developed to enable collaborative decision planning and response control of picking and walking operations in complex orchard environments.

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

XJZ designed the experiments. WT and ZY carried out the experiments. XZZ designed the study, analyzed the data, and wrote the manuscript. HM and XL supervised and revised the manuscript. All authors contributed to the article and approved the submitted version.

Funding

This study was supported by the scientific and technological breakthrough plan in key fields of Xinjiang Production and Construction Corps, China (Approval No.2022DB004), Foshan Science and Technology Innovation Project, China (Approval No.2120001008424) and Guangdong Basic and Applied Basic Research Foundation (Approval No.2022A1515140162 and No.2022A1515140013).

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.2023.1103276/full#supplementary-material

References

Aamir, M., Li, Z., Bazai, S., Wagan, R. A., Bhatti, U. A., Nizamani, M. M., et al. (2021). Spatiotemporal change of air-quality patterns in hubei province-a pre- to post-COVID-19 analysis using path analysis and regression. *ATMOSPHERE* 12 (10), 1338. doi: 10.3390/atmos12101338

Alam, A., Singh, L., Jaffery, Z., Verma, Y., and Diwakar, M. (2021). Distance-based confidence generation and aggregation of classifier for unstructured road detection. *J. King Saud Univ. - Comput. Inf. Sci* 34 (10, Part A), 8727–8738. doi: 10.1016/j.jksuci.2021.09.020

Bhatti, U. A., Nizamani, M. M., and Huang, M. X. (2022). Climate change threatens pakistan's snow leopards. *Science* 377 (6606), 585–586. doi: 10.1126/science.add9065

Bhatti, U. A., Wu, G., Bazai, S. U., Nawaz, S. A., Baryalai, M., Bhatti, M. A., et al. (2022a). A pre- to post-COVID-19 change of air quality patterns in anhui province using path analysis and regression. *Polish J. Of Environ. Stud.* 31 (5), 4029–4042. doi: 10.15244/pjoes/148065

Bhatti, U. A., Zeeshan, Z., Nizamani, M. M., Bazai, S., Yu, Z., and Yuan, L. (2022b). Assessing the change of ambient air quality patterns in jiangsu province of China pre-to post-COVID-19. *Chemosphere* 288, 132569. doi: 10.1016/j.chemosphere.2021.132569

Camizuli, E., and Carranza, E. J. (2018). "Exploratory data analysis (EDA)," in *The encyclopedia of archaeological sciences*. Ed. S. L. López Varela (John Wiley & Sons Inc), 1–7. doi: 10.1002/9781119188230.saseas0271

Chen, J., Qiang, H., Wu, J., Xu, G., and Wang, Z. (2021). Navigation path extraction for greenhouse cucumber-picking robots using the prediction-point hough transform. *Comput. Electron. Agriculture.* 180, 105911. doi: 10.1016/j.compag.2020.105911

Chen, J., Qiang, H., Wu, J., Xu, G., Wang, Z., and Liu, X. (2020). Extracting the navigation path of a tomato-cucumber greenhouse robot based on a median point hough transform. *Comput. Electron. Agriculture.* 174, 105472. doi: 10.1016/j.compag.2020.105472

Chen, X., Sun, Q., Guo, W., Qiu, C., and Yu, A. (2022). GA-net: a geometry prior assisted neural network for road extraction. *Int. J. Appl. Earth Observation Geoinformation.* 114, 103004. doi: 10.1016/j.jag.2022.103004

Chen, M., Tang, Y., Zou, X., Huang, K., Huang, Z., Zhou, H., et al. (2020). Three-dimensional perception of orchard banana central stock enhanced by adaptive multi-vision technology. *Comput. Electron. Agriculture.* 174, 105508. doi: 10.1016/j.compag.2020.105508

Fang, J., Meng, J., Liu, X., Li, Y., Qi, P., and Wei, C. (2022). Single-target detection of oncomelania hupensis based on improved YOLOv5s. *Front. Bioeng. Biotechnol.* 10. doi: 10.3389/fbioe.2022.861079

Feng, X., Zhao, C., Wang, C., Wu, H., Miao, Y., and Zhang, J. (2022). A vegetable leaf disease identification model based on image-text cross-modal feature fusion. *Front. In Plant Sci.* 13. doi: 10.3389/fpls.2022.918940

Fu, L. H., Duan, J., Zou, X., Lin, J., Zhao, L., Li, J., et al. (2020). Fast and accurate detection of banana fruits in complex background orchards. *IEEE Access.* 8, 196835–196846. doi: 10.1109/ACCESS.2020.3029215

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. Engineering.* 197, 245–256. doi: 10.1016/j.biosystemseng.2020.07.007

Fu, L., Wu, F., Zou, X., Jiang, Y., Lin, J., Yang, Z., et al. (2022). Fast detection of banana bunches and stalks in the natural environment based on deep learning. *Comput. Electron. Agriculture.* 194, 106800. doi: 10.1016/j.compag.2022.106800

Galvan, L. P. C., Bhatti, U. A., Campo, C. C., and Trujillo, R. A. S. (2022). The nexus between CO2 emission, economic growth, trade openness: evidences from middle-income trap countries. *Front. In Environ. Sci.* 10. doi: 10.3389/fenvs.2022.938776

Ge, Y., Xiong, Y., and From, P. J. (2022). Three-dimensional location methods for the vision system of strawberry-harvesting robots: development and comparison. *Precis. Agric* 24 (2), 764-782. doi: 10.1007/s11119-022-09974-4

Guan, H., Lei, X., Yu, Y., Zhao, H., Peng, D., Marcato, J.Jr., et al. (2022). Road marking extraction in UAV imagery using attentive capsule feature pyramid network. *Int. J. Appl. Earth Observation Geoinformation.* 107, 102677. doi: 10.1016/j.jag.2022.102677

Guo, A., Zou, X., Zhu, M., Chen, Y., Xiong, J., and Chen, L. (2013). Color feature analysis and recognition for litchi fruits and their main fruit bearing branch based on exploratory analysis. *Trans. Chin. Soc. Agric. Engineering.* 29 (4), 191–198. doi: 10.3969/j.issn.1002-6819.2013.04.024

Gupta, B., and Tiwari, M. (2019). Color retinal image enhancement using luminosity and quantile based contrast enhancement. *Multidimensional Syst. Signal Processing.* 30 (4), 1829–1837. doi: 10.1007/s11045-019-00630-1

He, N., Wang, J.-B., Zhang, L.-L., and Lu, K. (2015). An improved fractional-order differentiation model for image denoising. *Signal Processing.* 112, 180–188. doi: 10.1016/j.sigpro.2014.08.025

Huang, D., Liu, J., Zhou, S., and Tang, W. (2022). Deep unsupervised endoscopic image enhancement based on multi-image fusion. *Comput. Methods Programs Biomedicine.* 221, 106800. doi: 10.1016/j.cmpb.2022.106800

Jia, W., Wei, J., Zhang, Q., Pan, N., Niu, Y., Yin, X., et al. (2022). Accurate segmentation of green fruit based on optimized mask RCNN application in complex orchard. *Front. Plant Sci.* 13, 955256. doi: 10.3389/fpls.2022.955256

Jintasuttisak, T., Edirisinghe, E., and Elbattay, A. (2022). Deep neural network based date palm tree detection in drone imagery. *Comput. Electron. Agriculture.* 192, 106560. doi: 10.1016/j.compag.2021.106560

Jobson, D., Rahman, Z. U., and Woodell, G. (1997). A multi - scale retinex for bridging the gap between color images and the human observation of scenes. *Image Processing IEEE Trans. on.* 6, 965–976. doi: 10.1109/83.597272

Kang, H., Wang, X., Zhou, H., Au, W., and Chen, C. (2022). Geometry-aware fruit grasping estimation for robotic harvesting in apple orchards. *Comput. Electron. Agric.* 193, 106716. doi: 10.1016/j.compag.2022.106716

Khaki, S., and Wang, L. (2019). Crop yield prediction using deep neural networks. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.00621

Kim, W.-S., Lee, D.-H., Kim, Y.-J., Kim, T., Hwang, R.-Y., and Lee, H.-J. (2020). Path detection for autonomous traveling in orchards using patch-based CNN. *Comput. Electron. Agriculture.* 175, 105620. doi: 10.1016/j.compag.2020.105620

Lei, G., Yao, R., Zhao, Y., and Zheng, Y. (2021). Detection and modeling of unstructured roads in forest areas based on visual-2D lidar data fusion. *Forests* 12, 820. doi: 10.3390/f12070820

Li, Y. J., Feng, Q. C., Li, T., Xie, F., Liu, C., and Xiong, Z. C. (2022). Advance of target visual information acquisition technology for fresh fruit robotic harvesting: a review. *Agronomy-Basel* 12 (6), 1336. doi: 10.3390/agronomy12061336

Li, G., Fu, L., Gao, C., Fang, W., Zhao, G., Shi, F., et al. (2022). Multi-class detection of kiwifruit flower and its distribution identification in orchard based on YOLOv5l and euclidean distance. *Comput. Electron. Agriculture.* 201, 107342. doi: 10.1016/j.compag.2022.107342

Li, Y., Hong, Z., Cai, D., Huang, Y., Gong, L., and Liu, C. (2020). A SVM and SLIC based detection method for paddy field boundary line. *Sensors* 20 (9), 2610. doi: 10.3390/s20092610

Li, S., Li, K., and Qiao and L. Zhang, Y. (2022). A multi-scale cucumber disease detection method in natural scenes based on YOLOv5. *Comput. And Electron. In Agric.* 202, 107363. doi: 10.1016/j.compag.2022.107363

Li, Y., Tong, G., Sun, A., and Ding, W. (2018). Road extraction algorithm based on intrinsic image and vanishing point for unstructured road image. *Robotics Autonomous Systems.* 109, 86–96. doi: 10.1016/j.robot.2018.08.011

Liang, L., Qin, K., Jiang, S., Wang, X., and Shi, Y. (2021). Impact of epidemic-affected labor shortage on food safety: a Chinese scenario analysis using the CGE model. *FOODS* 10 (11), 2679. doi: 10.3390/foods10112679

Liao, J., Babiker, I., Xie, W.-f., Li, W., and Cao, L. (2022). Dandelion segmentation with background transfer learning and RGB-attention module. *Comput. Electron. Agric.* 202, 107355. doi: 10.1016/j.compag.2022.107355

Lin, X., Qi, L., Pan, H., and Sharp, B. (2022). COVID-19 pandemic, technological progress and food security based on a dynamic CGE model. *Sustainability* 14 (3), 1842. doi: 10.3390/su14031842

Lin, G., Zhu, L., Li, J., Zou, X., and Tang, Y. (2021). Collision-free path planning for a guava-harvesting robot based on recurrent deep reinforcement learning. *Comput. Electron. Agriculture.* 188, 106350. doi: 10.1016/j.compag.2021.106350

Liu, Y., Xu, W., Dobaie, A. M., and Zhuang, Y. (2018). Autonomous road detection and modeling for UGVs using vision-laser data fusion. *Neurocomputing* 275, 2752–2761. doi: 10.1016/j.neucom.2017.11.042

Liu, N., Yang, C., and Cao, H. (2017). Noise suppression of the reconstruction of infrared digital holography based on pyramid-based bilateral filter. *Infrared Phys. Technol.* 85, 352–358. doi: 10.1016/j.infrared.2017.07.023

Lv, J., Xu, H., Han, Y., Lu, W., Xu, L., Rong, H., et al. (2022). A visual identification method for the apple growth forms in the orchard. *Comput. Electron. Agriculture.* 197, 106954. doi: 10.1016/j.compag.2022.106954

Lyu, S., Li, R., Zhao, Y., Li, Z., Fan, R., and Liu, S. (2022). Green citrus detection and counting in orchards based on YOLOv5-CS and AI edge system. *Sensors* 22, 576. doi: 10.3390/s22020576

Ma, Y., Zhang, W., Qureshi, W. S., Gao, C., Zhang, C., and Li, W. (2021). Autonomous navigation for a wolfberry picking robot using visual cues and fuzzy control. *Inf. Process. Agriculture.* 8 (1), 15–26. doi: 10.1016/j.inpa.2020.04.005

Meyer, G., Hindman, T., and Laksmi, K. (1999). Machine vision detection parameters for plant species identification. *Proc. SPIE - Int. Soc. Optical Eng.* 3543, 327–335. doi: 10.1117/12.336896

Nawaz, S. A., Li, J. B., Bhatti, U. A., Bazai, S. U., Zafar, A., Bhatti, M. A., et al. (2021). A hybrid approach to forecast the COVID-19 epidemic trend. *PloS One* 16 (10), e0256971. doi: 10.1371/journal.pone.0256971

Ning, Z., Luo, L., Ding, X., Dong, Z., Yang, B., Cai, J., et al. (2022). Recognition of sweet peppers and planning the robotic picking sequence in high-density orchards. *Comput. Electron. Agriculture.* 196, 106878. doi: 10.1016/j.compag.2022.106878

Peng, H., Zou, X., Guo, A., Xiong, J., and Chen, Y. (2013). Color model analysis and Recognition for parts of citrus based on exploratory data analysis. *Trans. Chin. Soc. Agric. Machinery* 44 (S1), 253–259. doi: 10.6041/j.issn.1000-1298.2013.S1.045

Phung, S., Le, M., and Bouzerdoum, A. (2016). Pedestrian lane detection in unstructured scenes for assistive navigation. *Comput. Vision Image Understanding.* 149, 186–196. doi: 10.1016/j.cviu.2016.01.011

Qi, J., Liu, X., Liu, K., Xu, F., Guo, H., Tian, X., et al. (2022). An improved YOLOv5 model based on visual attention mechanism: application to recognition of tomato virus disease. *Comput. Electron. Agriculture.* 194, 106780. doi: 10.1016/j.compag.2022.106780

Qi, N., Yang, X., Chuanxiang, L., Lu, R., He, C., and Cao, L. (2019). Unstructured road detection *via* combining the model-based and feature-based methods. *IET Intelligent Transport Syst.* 13, 1533–1544. doi: 10.1049/iet-its.2018.5576

Rong, J., Wang, P., Wang, T., Hu, L., and Yuan, T. (2022). Fruit pose recognition and directional orderly grasping strategies for tomato harvesting robots. *Comput. Electron. Agriculture.* 202, 107430. doi: 10.1016/j.compag.2022.107430

Routray, S., Malla, P. P., Sharma, S., Panda, S. K., and Palai, G. (2020). A new image denoising framework using bilateral filtering based non-subsampled shearlet transform. *Optik* 216, 164903. doi: 10.1016/j.ijleo.2020.164903

Rysz, M. W., and Mehta, S. S. (2021). A risk-averse optimization approach to human-robot collaboration in robotic fruit harvesting. *Comput. Electron. Agriculture.* 182, 106018. doi: 10.1016/j.compag.2021.106018

Safarik, I., Baldikova, E., Prochazkova, J., and Pospiskova, K. (2019). Smartphone-based image analysis for evaluation of magnetic textile solid phase extraction of colored compounds. *Heliyon* 5 (12), e02995. doi: 10.1016/j.heliyon.2019.e02995

She, J., Zhan, W., Hong, S., Min, C., Dong, T., Huang, H., et al. (2022). A method for automatic real-time detection and counting of fruit fly pests in orchards by trap bottles *via* convolutional neural network with attention mechanism added. *Ecol. Informatics.* 70, 101690. doi: 10.1016/j.ecoinf.2022.101690

Singh, P. (2020). A neutrosophic-entropy based clustering algorithm (NEBCA) with HSV color system: a special application in segmentation of parkinson's disease (PD) MR images. *Comput. Methods Programs Biomedicine.* 189, 105317. doi: 10.1016/j.cmpb.2020.105317

Su, Y., Gao, Y., Zhang, Y., Alvarez, J. M., Yang, J., and Kong, H. (2019). An illumination-invariant nonparametric model for urban road detection. *IEEE Trans. Intell. Veh.* 4 (1), 14-23. doi: 10.1109/TIV.2018.2886689

Sun, Q., Zhang, R., Chen, L., Zhang, L., Zhang, H., and Zhao, C. (2022). Semantic segmentation and path planning for orchards based on UAV images. *Comput. Electron. Agriculture.* 200, 107222. doi: 10.1016/j.compag.2022.107222

Tang, Y., Chen, C., Leite, A. C., and Xiong, Y. (2023a). Editorial: precision control technology and application in agricultural pest and disease control. *Front. Plant Sci.* 14, 1163839. doi: 10.3389/fpls.2023.1163839

Tang, Y., Chen, M., Wang, C., Luo, L., Li, J., Lian, G., et al. (2020). Recognition and localization methods for vision-based fruit picking robots: a review. *Front. Plant Science.* 11. doi: 10.3389/fpls.2020.00510

Tang, Y., Qiu, J., Zhang, Y., Wu, D., Cao, Y., Zhao, K., et al. (2023b). Optimization strategies of fruit detection to overcome the challenge of unstructured background in field orchard environment: a review. *Precis. Agric.* doi: 10.1007/s11119-023-10009-9

Tang, Y., Zhou, H., Wang, H., and Zhang, Y. (2023c). Fruit detection and positioning technology for a camellia oleifera c. Abel orchard based on improved YOLOv4-tiny model and binocular stereo vision. *Expert Syst. WITH APPLICATIONS.* 211, 118573. doi: 10.1016/j.eswa.2022.118573

Wang, C.-Y., Bochkovskiy, A., and Liao, H.-y. (2022). YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *Arxiv* doi: 10.48550/arXiv.2207.02696

Wang, C., Lee, W. S., Zou, X., Choi, D., Gan, H., and Diamond, J. (2018). Detection and counting of immature green citrus fruit based on the local binary patterns (LBP) feature using illumination-normalized images. *Precis. Agric.* 19, 1062–1083 doi: 10.1007/s11119-018-9574-5

Wang, E., Li, Y., Sun, A., Huashuai, G., Yang, J., and Fang, Z. (2019). Road detection based on illuminant invariance and quadratic estimation. *Optik* 185, 672–684. doi: 10.1016/j.ijleo.2019.04.026

Wang, Y., Liu, D., Zhao, H., Li, Y., Song, W., Liu, M., et al. (2022). Rapid citrus harvesting motion planning with pre-harvesting point and quad-tree. *Comput. Electron. Agriculture.* 202, 107348. doi: 10.1016/j.compag.2022.107348

Wang, C., Luo, T., Zhao, L., Tang, Y., and Zou, X. (2019). Window zooming–based localization algorithm of fruit and vegetable for harvesting robot. *IEEE Access.* 7, 103639–103649. doi: 10.1109/ACCESS.2019.2925812

Wang, P., Wang, Z., Lv, D., Zhang, C., and Wang, Y. (2021). Low illumination color image enhancement based on gabor filtering and retinex theory. *Multimedia Tools Applications.* 80 (12), 17705–17719. doi: 10.1007/s11042-021-10607-7

Wang, X., Yang, W., Lv, Q., Huang, C., Liang, X., Chen, G., et al. (2022). Field rice panicle detection and counting based on deep learning. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.966495

Wang, H., Zhang, S., Zhao, S., Lu, J., Wang, Y., Li, D., et al. (2022). Fast detection of cannibalism behavior of juvenile fish based on deep learning. *Comput. Electron. Agriculture.* 198, 107033. doi: 10.1016/j.compag.2022.107033

Woebbecke, D., Meyer, G., Bargen, K., and Mortensen, D. (1993). Plant species identification, size, and enumeration using machine vision techniques on near-binary images. *SPIE Optics Agric. Forestry* 1836, 208-219. doi: 10.1117/12.144030

Woebbecke, D., Meyer, G., Bargen, K., and Mortensen, D. (1995). Color indices for weed identification under various soil, residue, and lighting conditions. *Trans. ASAE.* 38, 259–269. doi: 10.13031/2013.27838

Wu, F., Duan, J., Ai, P., Chen, Z., Yang, Z., and Zou, X. (2022). Rachis detection and three-dimensional localization of cut off point for vision-based banana robot. *Comput. Electron. Agriculture.* 198, 107079. doi: 10.1016/j.compag.2022.107079

Wu, F. Y., Duan, J. L., Chen, S. Y., Ye, Y. X., Ai, P. Y., and Yang, Z. (2021). Multi-target recognition of bananas and automatic positioning for the inflorescence axis cutting point. *Front. In Plant Sci.* 12. doi: 10.3389/fpls.2021.705021

Wu, Z., Li, G., Yang, R., Fu, L., Li, R., and Wang, S. (2022). Coefficient of restitution of kiwifruit without external interference. *J. Food Engineering.* 327, 111060. doi: 10.1016/j.jfoodeng.2022.111060

Xin, J., Zhang, X., Zhang, Z., and Fang, W. (2019). Road extraction of high-resolution remote sensing images derived from DenseUNet. *Remote Sens.* 11 (21), 2499. doi: 10.3390/rs11212499

Xu, F., Hu, B., Chen, L., Wang, H., Xia, Q., Sehdev, P., et al. (2018). An illumination robust road detection method based on color names and geometric information. *Cogn. Syst. Res.* 52, 240–250. doi: 10.1016/j.cogsys.2018.06.019

Xu, L., Xie, Y., Chen, X., Chen, Y., Kang, Z., Huang, P., et al. (2022). Design of an efficient combined multipoint picking scheme for tea buds. *Front. Plant Sci.* 13, 1042035. doi: 10.3389/fpls.2022.1042035

Yang, Z. (2021). Review of smart robots for fruit and vegetable picking in agriculture. *Int. J. Agric. Biol. Engineering.* 14, 33–54. doi: 10.25165/j.ijabe.20221501.7232

Yang, Z., Ouyang, L., Zhang, Z., Duan, J., Yu, J., and Wang, H. (2022). Visual navigation path extraction of orchard hard pavement based on scanning method and neural network. *Comput. Electron. Agriculture.* 197, 106964. doi: 10.1016/j.compag.2022.106964

Yang, M., Yuan, Y., and Liu, G. (2022). SDUNet: road extraction *via* spatial enhanced and densely connected UNet. *Pattern Recognit.* 126, 108549. doi: 10.1016/j.patcog.2022.108549

Ye, C.-w., Yu, Z.-w., Kang, R., Yousaf, K., Qi, C., Chen, K.-j., et al. (2020). An experimental study of stunned state detection for broiler chickens using an improved convolution neural network algorithm. *Comput. Electron. Agric.* 170, 105284. doi: 10.1016/j.compag.2020.105284

Zang, H., Wang, Y., Ru, L., Zhou, M., Chen, D., Zhao, Q., et al. (2022). Detection method of wheat spike improved YOLOv5s based on the attention mechanism. *Front. Plant Sci.* 13, 993244. doi: 10.3389/fpls.2022.993244

Zhang, P., Liu, X., Yuan, J., and Liu, C. (2022). YOLO5-spear: a robust and real-time spear tips locator by improving image augmentation and lightweight network for selective harvesting robot of white asparagus. *Biosyst. Engineering.* 218, 43–61. doi: 10.1016/j.biosystemseng.2022.04.006

Zhang, Z., Qiao, Y., Guo, Y., and He, D. (2022). Deep learning based automatic grape downy mildew detection. *Front. Plant Sci.* 13, 872107. doi: 10.3389/fpls.2022.872107

Zhang, X., Sun, X., and Lu, L. (2022). Research on multi-image and multi-parameter fusion algorithm based on detail restoration. *Multimedia Tools Applications.* 81 (12), 16589–16600. doi: 10.1007/s11042-022-12682-w

Zhang, Z. Q., Zhang, X., Cao, R., Zhang, M., Li, H., Yin, Y., et al. (2022). Cut-edge detection method for wheat harvesting based on stereo vision. *Comput. Electron. Agriculture.* 197, 106910. doi: 10.1016/j.compag.2022.106910

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

Zhou, Y., Tang, Y., Zou, X., Wu, M., Tang, W., Meng, F., et al. (2022). Adaptive active positioning of camellia oleifera fruit picking points: classical image processing and YOLOv7 fusion algorithm. *Appl. Sci.* 12 (24), 12959. doi: 10.3390/app122412959

Zhou, M., Xia, J., Yang, F., Zheng, K., Hu, M., Li, D., et al. (2021). Design and experiment of visual navigated UGV for orchard based on hough matrix and RANSAC. *Int. J. Agric. Biol. Engineering.* 14, 176–184. doi: 10.25165/j.ijabe.20211406.5953

TYPE Review
PUBLISHED 02 June 2023
DOI 10.3389/fpls.2023.1139232

Check for updates

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU), United States

REVIEWED BY
Yongxin Liu,
Embry–Riddle Aeronautical University, United States
Costas A Varotsos,
National and Kapodistrian University of Athens, Greece
Yanqiu Yang,
The Pennsylvania State University (PSU), United States

*CORRESPONDENCE
Fernando Auat Cheein
✉ fernando.auat@usm.cl

RECEIVED 06 January 2023
ACCEPTED 08 May 2023
PUBLISHED 02 June 2023

CITATION
Estrada JS, Fuentes A, Reszka P and Auat Cheein F (2023) Machine learning assisted remote forestry health assessment: a comprehensive state of the art review. *Front. Plant Sci.* 14:1139232.
doi: 10.3389/fpls.2023.1139232

Machine learning assisted remote forestry health assessment: a comprehensive state of the art review

Juan Sebastián Estrada[1], Andrés Fuentes[2], Pedro Reszka[3] and Fernando Auat Cheein[1]*

[1]Department of Electronic Engineering, Universidad Tecnica Federico, Santamaria, Valparaíso, Chile, [2]Department of Industrial Engeneering, Universidad Tecnica Federica, Santamaria, Valparaíso, Chile, [3]Faculty on Engineering and Science, Universidad Adolfo Ibáñez, Santiago, Chile

Forests are suffering water stress due to climate change; in some parts of the globe, forests are being exposed to the highest temperatures historically recorded. Machine learning techniques combined with robotic platforms and artificial vision systems have been used to provide remote monitoring of the health of the forest, including moisture content, chlorophyll, and nitrogen estimation, forest canopy, and forest degradation, among others. However, artificial intelligence techniques evolve fast associated with the computational resources; data acquisition, and processing change accordingly. This article is aimed at gathering the latest developments in remote monitoring of the health of the forests, with special emphasis on the most important vegetation parameters (structural and morphological), using machine learning techniques. The analysis presented here gathered 108 articles from the last 5 years, and we conclude by showing the newest developments in AI tools that might be used in the near future.

KEYWORDS

forestry health assessment, remote sensing, machine learning, vision system, spectral information

1 Introduction

Climate change has increased the frequency and duration of droughts around the world (Cook et al., 2014). This has a special impact on ecosystems, where it is estimated by the United Nations Convention to Combat Desertification (UNCCD) that in the last 40 years the percentage of vegetated areas affected by droughts has doubled, and around 12 million hectares of agricultural land have been lost due to desertification (UNCCD, 2022). Another issue caused by intense droughts is the increase in wildfires. According to UNCCD (2022) more than 84% of terrestrial ecosystems are in danger due to more frequent and intensive fires. Forests are particularly affected by longer droughts due to water stress; the relationship between forestry health and posterior forest recovery is still being studied (Xu et al., 2018).

Forest management plays a fundamental role in the analysis of forest health. Its main target is to reduce risks or negative impacts derived from external disturbances (Migliavacca et al., 2021) including wildfires (Hillman et al., 2021; Reilly et al., 2021; Rodríguez et al., 2021; Wells et al., 2021; Trencanová et al., 2022), atmospheric pollution, forest stress (Cężkowski et al., 2020; Huo et al., 2021), pests (Huo et al., 2021), climate change, and forest diseases (Lin et al., 2018; Sapes et al., 2022). The scientific community has established the use of forest indicators to ease forest health assessment (Trumbore et al., 2015; Cai et al., 2021; Kopacková-Strnadová et al., 2021; Migliavacca et al., 2021; Neuville et al., 2021). These indicators comprise in their nucleus, a previous examination of factors associated with the physical and chemical forest attributes, such as greenness of the leaves, nitrogen content, tree height, canopy height, diameter at breast height, and others. Their importance lies in the study of water absorption, drought response, moisture content, changes in vegetation, and detection of tree diseases (Abdollahnejad and Panagiotidis, 2020; Raddi et al., 2021; Malabad et al., 2022; Zhuo et al., 2022).

Technological developments have allowed researchers to process massive data and obtain measurements of large portions of land. Unmanned aerial vehicles have been used in recent years as mechanisms to gather massive information about various ecosystems (Eugenio et al., 2020; Osco et al., 2021; Sangjan and Sankaran, 2021). Coupling UAVs with computer vision systems (RGB, multi-spectral, hyper-spectral and thermal cameras) and other sensors as LiDAR has allowed researchers to estimate forest parameters like height, canopy cover, DBH, vegetation indexes (Abdollahnejad and Panagiotidis, 2020; Kopacková-Strnadová et al., 2021; Raddi et al., 2021; Malabad et al., 2022; Zhuo et al., 2022). The promising use of UAVs in the assessment of forest health allows the experimentation with larger-scale satellite monitoring systems, particularly LANDSAT, SENTINEL, and even Google Earth (Ahmad et al., 2021).

Likewise, the use of remotely sensed imagery has contributed to the study of vegetation indices (Becker et al., 2018; Gallardo-Salazar et al., 2021; Rodríguez et al., 2021; Zhang Y. et al., 2021; Fakhri et al., 2022; Qiu et al., 2022; Talavera et al., 2022; Xu et al., 2022; Yang et al., 2022), forest mapping (Lin Y. Z. et al., 2021; Onishi and Ise, 2021; Fakhri et al., 2022; Li et al., 2022; Nasiri et al., 2022; Trencanová et al., 2022; Xu et al., 2022), evaluation and detection of diseased forests (Lin et al., 2018; Sapes et al., 2022), canopy characterization(Furukawa et al., 2021; Ribas Costa et al., 2022), tree species classification (Liu et al., 2021; Mäyrä et al., 2021; Onishi and Ise, 2021; Zhang C. et al., 2021; Hell et al., 2022; Yang and Kan, 2022), identification of fire-prone ecosystems (Trencanová et al., 2022), prediction of chlorophyll and nitrogen content (Yao et al., 2021; Narmilan et al., 2022; Wan et al., 2022), recognition of intrinsic forest factors (Xu et al., 2019; Dainelli et al., 2021), wildfire prevention (Trencanová et al., 2022), and so on. The analysis of these applications guarantees a comprehensive assessment of woodland features which determines the current forest health status and allows for better forest management.

In accordance with the data gathered by the different robotic platforms and sensors, it is essential to know how to treat the information. Although traditional methods such as statistical analysis are a viable option for post-processing data, currently the use of machine learning techniques has been chosen in order to generalize models, increase the accuracy of parameters estimation, and provide better feature prediction to the ecosystems variability and forest species involved (Corte et al., 2020; Wells et al., 2021; Zhang Y. et al., 2021; Ilniyaz et al., 2022; Narmilan et al., 2022; Nasiri et al., 2022; Qiu et al., 2022). In addition, some works have considered the use of deep learning strategies to further improve forest health monitoring capabilities and obtain more detailed individual tree features (Mäyrä et al., 2021; Onishi and Ise, 2021; Zhang C. et al., 2021; Hell et al., 2022; Li et al., 2022; Trencanová et al., 2022).

Machine learning (ML) models have been used as both classifiers and predictors. Forest structure parameters and tree phenotypic features are predicted using machine learning techniques with input data gathered from LiDAR, RGB, and Multi-spectral cameras (Shin et al., 2018; McClelland et al., 2019; Puliti et al., 2019; Abdollahnejad and Panagiotidis, 2020; Fan et al., 2020; Imangholiloo et al., 2020; Ahmad et al., 2021; Cai et al., 2021; Neuville et al., 2021; Sangjan and Sankaran, 2021; Yu et al., 2021). Predictions of leaf moisture, chlorophyll, and nitrogen content, have been achieved using machine learning methods (Watt et al., 2020; Lou et al., 2021; Raddi et al., 2021; Raj et al., 2021; Narmilan et al., 2022; Zhuo et al., 2022). The most common predictor is linear regression, but other common ones are support vector machine regression, random forest regression, and gradient boost machines (McClelland et al., 2019; Blanco-Sacristán et al., 2021; Fraser and Congalton, 2021b; Yu et al., 2021; Torre-Tojal et al., 2022). Another task that can be accomplished using ML methods is tree classification, which is important for forest inventory and mapping. The most common classifiers are random forests, support vector machines, and artificial neural networks (Feng et al., 2020; Guo et al., 2021; Hologa et al., 2021). Another use for classifiers in forestry health assessment is the identification of live trees and snags, the ratio between these two is an important parameter to evaluate forest health (Shovon et al., 2022).

The use of high-resolution cameras has allowed researchers to couple them with deep convolutional neural networks (Osco et al., 2021). Using deep learning structures alongside high-resolution aerial images has had good results in individual tree crown segmentation (Lin and Chuang, 2021; Onishi and Ise, 2021; Li et al., 2022). Other applications of deep convolutional neural networks are tree identification from aerial RGB and multi-spectral images, using temporal information has also been explored by researchers with the aid of recurrent convolutional neural networks (Feng et al., 2020). The most common deep learning back-bones used to perform feature extraction are, VGG19, RES-NET and Seg-Net (Pulido et al., 2020; Lin and Chuang, 2021; Hao Z. et al., 2022). Other structures used in semantic segmentation processes are U-NET and Mask-RCNN (Pulido et al., 2020).

This work presents a systematic review of scientific articles from the last five years (2017-2022) focused on forest health assessment assisted by remote sensing and machine learning techniques. For our analysis, we used Scopus (www.scopus.com) scientific database. We intend to determine which forest properties are considered to assess forest health, and how remote sensing in conjunction with machine learning strategies are used to estimate such features. Other review works related to remote sensing for forestry applications do not include information about the novel machine learning algorithms to relate the data gathered by various sensors and the expected metrics that are needed to evaluate forest health. For example, (Torres et al., 2021) describes various applications of remote sensing in the assessment of forest status and health including stress factors, plagues, tree mortality, tree decline, and tree health. However, there is no in-depth discussion about how the data is processed in those studies. A similar case is the work presented by (Guimarães et al., 2020), which covers other areas for forest management including tree classification and mapping, and tree parameter estimation; however, the processing techniques are not addressed. In Eugenio et al. (2020) it is presented a similar approach but focused on remotely piloted systems, and not considering satellite platforms that are used for the assessment of forests. A complete review of deep learning algorithms for forestry was presented in Diez et al. (2021), focusing directly on the images processing; however, such work does not present information about machine learning for regression problems. A more complete review including sensors and methods is discussed in Pérez-Cabello et al. (2021); but it is limitedto the assessment of post-fire vegetation recovery. To the best of our knowledge, our work is the only one that offers a more in-depth discussion about machine learning methods (including deep learning) and how they are implemented alongside remote sensing techniques for the assessment of forest health. Table 1 contains a comparison between our work and previous reviews during the five-year period under study.

This paper is organized as follows: Section 2 presents the main issues and forestry problems studied using both remote sensing techniques and machine learning methods. Section 3 presents the hardware used in the assessment of forestry health, it includes both sensors and platforms. Section 4 deals with the machine learning techniques that are used to process data. Section 5 includes the discussion and the challenges that arise in the assessment of forestry health using remote sensing aided by machine learning.

2 Vegetative problems

This section discusses the vegetative issues that are currently being studied for forestry health assessment. In a broad sense, Figure 1 shows the distribution of the prevalent issues that have been studied the most in the reviewed articles; these were: tree classification and identification, tree structure identification, biomass estimation chlorophyll estimation, crown fuel estimation, and water and moisture content prediction.

The first subsection is dedicated to Vegetation Indices since they are one of the most important features that help researchers predict forest and individual features from the

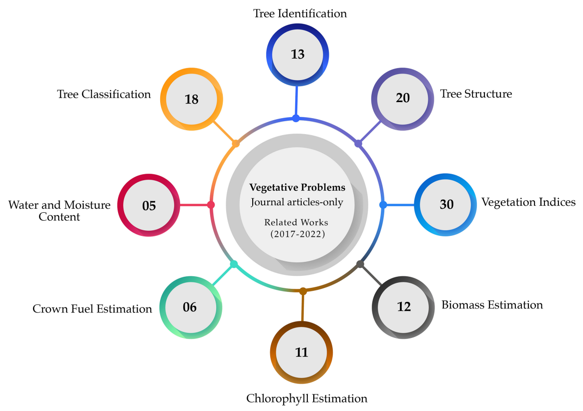


FIGURE 1
Distribution of the main vegetative problems that were studied in the reviewed articles.

TABLE 1 Comparison between the present work and other similar reviews related to remote sensing in forestry applications.

Article	Years	Forest issue	Sensors	Platforms	Methods
Our Work	2017-2022	Vegetation indices, Biomass estimation, Tree structure parameters, Tree identification, Tree recognition, Water and moisture content, Chlorophyll estimation	Cameras (RGB, Hyperspectral, Multispectral, Thermal); LiDAR; TerrestrialLaser Scanning, Spectrometer	UAV, Satellite	Linear regression, Random forest, SVM, K-nearest neighbors, Deep learning
Pérez-Cabello et al. (2021)	N/A	Post-fire vegetation recovery	Cameras (RGB, Hyperspectral, Multispectral, Thermal), LiDAR, Terrestrial Laser Scanning, Spectrometer	UAV, Satellite	Not Specified
Eugenio et al. (2020)	2000-2019	Forest parameter estimation, Fire monitoring, Pest and disease detection, Natural conservation	Cameras (RGB, Hyperspectral, Multispectral, Thermal), LiDAR	UAV	Not Specified
Torres et al. (2021)	2015-2020	Forest plague detection, Forest current health, Forest health decline and mortality	Cameras (RGB, Hyperspectral, Multispectral, Thermal), LiDAR	UAV, Satellite	Random forest, SVM, K-nearest neighbors, Neural networks
(Guimarães et al., 2020)	N/A	Forest parameter estimation, Tree classification and mapping, Forest health monitoring	Cameras (RGB, Hyperspectral, Multispectral, Thermal), LiDAR	UAV	Not specified
(Diez et al., 2021)	2017-2021	Forest parameter estimation, Tree classification and mapping, Forest health monitoring	Cameras (RGB, Multispectral)	UAV	Deep learning

reflected electromagnetic spectrum. The following subsection discusses tree classification and identification, tree structure parameters, biomass estimation, chlorophyll estimation, crown fuel estimation, and water and moisture content prediction.

2.1 Vegetation indices

A vegetation index is a mathematical transformation of two or more spectral bands that are designed to enhance a specific property or characteristic of the vegetation (Munnaf et al., 2020).

Recently, these indices have been used as input data for prediction and classification purposes alike, the spectrum of tree canopies can be considered a distinctive feature of the specific vegetation, thus making VIs useful for both vegetation identification in aerial photographs and for tree classification (Abdollahnejad and Panagiotidis, 2020; Imangholiloo et al., 2020; Yang and Kan, 2020; Guo et al., 2021; Arevalo-Ramirez et al., 2022; Cabrera-Ariza et al., 2022; Shovon et al., 2022). Photosynthetic pigments have a distinctive reflectance in some bands, thus the prediction of chlorophyll content and other pigments is suitable with the appropriate VI (Watt et al., 2020; Kopacková-Strnadová et al., 2021; Lou et al., 2021; Lu et al., 2021; Raddi et al., 2021; Raj et al., 2021; Zhuo et al., 2022). Another application using VIs is the prediction of biomass in different and (Morgan et al., 2021; Torre-Tojal et al., 2022; Yan et al., 2022).

Tables 2A–D contain the main VIs used in different studies regarding forest health, and their application; where R, G, B, NIR, and RE denote the reflectance in the Red, Green, Blue, Near Infrared, and Red Edge multi-spectral bands. Researchers focus on these five bands since most of the reviewed works use commercial infrared cameras that capture the radiation at these wavelengths. Other indices take advantage of the full spectrum and not only on specific bands but these indices are also obtained with the aid of a hyper-spectral camera or by a laboratory or hand-held spectrometer (Abdollahnejad and Panagiotidis, 2020; Watt et al., 2020; Yang and Kan, 2020; de Almeida et al., 2021; Raj et al., 2021; Villacrés and Cheein, 2022; Wan et al., 2022; Yang and Kan, 2022). Li et al. (2021) uses spectral indices to estimate the leaf water content. The authors specify five different indices: Simple Ratio, Simple Difference, normalized difference, double difference index, and difference ratio. Other indices are used to estimate the content

TABLE 2A Common VIs used in the reviewed articles.

Vegetation Index	Formula	Application	Reference
Normalized Difference Vegetation Index (*NVDI*)	$\frac{NIR - R}{NIR + R}$	Predict forest vertical structure. Tree Recognition. Chlorophyll Content Estimation. Fuel Content Prediction.	Ahmed et al. (2021a); Raddi et al. (2021); Yu et al. (2021); Arevalo-Ramirez et al. (2022); Qiao et al. (2022); Villacrés and Cheein (2022); Zhuo et al. (2022)
Green normalized difference vegetation index (*GNDVI*)	$\frac{NIR - G}{NIR + G}$	Predict forest vertical structure. Soil Moisture Content Prediction. Chlorophyll Content estimation. Fuel Content Prediction	Yu et al. (2021); Raddi et al. (2021); Cheng et al. (2022); Arevalo-Ramirez et al. (2022); Villacrés and Cheein (2022)
Normalized difference red edge index (*NDRE*)	$\frac{NIR - RE}{NIR + RE}$	Predict forest vertical structure	Yu et al. (2021)
Structure insensitive pigment index (*SIPI*)	$\frac{NIR - B}{NIR - R}$	Predict forest vertical structure. Soil Moisture Content Prediction. Chlorophyll Content Prediction	Yu et al. (2021); Cheng et al. (2022)
Normalized green blue difference index (*NGBDI*)	$\frac{G - B}{G + B}$	Tree Classification	Guo et al. (2021)
Normalized green red difference index (*NGRDI*)	$\frac{G - R}{G + R}$	Tree Classification	Guo et al. (2021); Cabrera-Ariza et al. (2022)
Green red difference index (*GRDI*)	$G - R$	Tree Classification	Guo et al. (2021)
Normalized blue green vegetation index (*NBGVI*)	$\frac{B - G}{B + G}$	Tree Classification	Guo et al. (2021)
Normalized excessive green index (*NEGI*)	$\frac{2G - R - B}{2G + R + B}$	Tree Classification	Guo et al. (2021)
Modified Green Blue Vegetation Index (*MGRVI*)	$\frac{G^2 - R^2}{G^2 + R^2}$	Biomass Prediction	Morgan et al. (2021)
Modified Visible Atmospheric Resistant Index (*MVARI*)	$\frac{G - B}{G + R - B}$	Biomass Prediction	Morgan et al. (2021)
Red-Green-Blue Vegetation Index (*RGBVI*)	$\frac{G^2 - B{\star}R}{G^2 - B{\star}R}$	Biomass Prediction	Morgan et al. (2021)

TABLE 2B Common VIs used in the reviewed articles.

Vegetation Index	Formula	Application	Reference
Triangular Greenness Index (*TGI*)	$G - 0.39R - 0.61B$	Biomass Prediction	Morgan et al. (2021)
Visible atmospheric resistant index (*VARI*)	$\frac{G-R}{G+R-B}$	Tree Structure. Biomass Prediction. Leaf Nitrogen Concentration	Lu et al. (2021); Morgan et al. (2021); Qiao et al. (2022)
Green red ration index (*GRRI*)	$\frac{G}{R}$	Leaf Nitrogen Concentration	Lu et al. (2021)
Normalized redness intensity (*NRI*)	$\frac{R}{R+G+B}$	Leaf Nitrogen Concentration	Lu et al. (2021)
Green Red Vegetation Index (*GRVI*)	$\frac{G-R}{G+R}$	Leaf Nitrogen Concentration. Biomass Prediction	K.C. et al. (2021); Lu et al. (2021)
Atmospherical Resistant Vegetation Index (*ARVI*)	$\frac{G-R}{G+R-B}$	Leaf Nitrogen Concentration	Lu et al. (2021)
Simple Ratio (*SR*)	$\frac{NIR}{R}$	Tree Classification. Chlorophyll Content Estimation	Abdollahnejad and Panagiotidis (2020); Zhuo et al. (2022)
Soil Adjusted Vegetation Index (*SAVI*)	$1.5\frac{NIR-R}{NIR+R+0.5}$	Tree Classification.Soil Moisture Content Prediction	Abdollahnejad and Panagiotidis (2020); Cheng et al. (2022)
Chlorophyll index (*CI*)	$\frac{NIR}{RE}-1$	Tree Classification	Abdollahnejad and Panagiotidis (2020)
Plant Sense Reflectance Index (*PSRI*)	$\frac{R-G}{RE}$	Tree Classification	Abdollahnejad and Panagiotidis (2020)
Modified canopy chlorophyll content index (*M3CL*)	$\frac{NIR+R+RE}{NIR-RED+RE}$	Tree Classification	Abdollahnejad and Panagiotidis (2020)
Shadow Index (*SI*)	$\frac{R+G+B}{3}$	Biomass Prediction	K.C. et al. (2021)
Modified Simple Ratio Index (*MSR*)	$\frac{NIR/R-1}{(NIR/R+1)^{\frac{1}{2}}}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Optimized Soil Adjusted Vegetation Index (*OSAVI*)	$\frac{1.16(NIR-R)}{NIR+R+0.16}$	Soil Moisture Content Prediction. Forest Structure	Arevalo-Ramirez et al. (2022); Cheng et al. (2022)
Ratio Vegetation Index (*RVI*)	$\frac{NIR}{R}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Ratio Vegetation Index 2 (RVI_2)	$\frac{NIR}{G}$	Soil Moisture Content Prediction	Cheng et al. (2022)

of phosphorus and nitrogen, which is related to photosynthetic efficiency (Watt et al., 2020; Raj et al., 2021), the information gathered by hyperspectral indices, allows the processing data models to make more accurate predictions.

Comparisons between hyper-spectral information and multi-spectral indices have been performed to evaluate drought responses in various ecosystems (Raddi et al., 2021). Other studies show that there is the possibility to recreate indices from hyper-spectral bands with the information gathered from multi-spectral indices (Villacrés and Cheein, 2022).

This section includes only a few of the most common VIs, however, more extensive articles and reviews are available, and the reader is encouraged to see (Tran et al., 2022).

2.2 Biomass estimation

From an ecological standpoint, biomass is defined as the mass of living organisms in a determined area or ecosystem. Biomass depending on the environment has multiple functions, for example, to know about carbon sinks and it is important in water exchange with the atmosphere. However, ecosystems are constantly changing due to climate change has strengthened environmental stressors for various ecosystems, changing the natural composition of biomass; thus estimating its value is a strong indicator of how an ecosystem responds to external changes. Biomass is also an indicator of biological fuel present in environments (Morgan et al., 2021).

2.3 Chlorophyll estimation

Chlorophyll concentration (CC) indicates the physiological and structural basis by which leaves drive photosynthesis (Narmilan et al., 2022) and its relationship to soil respiration (Yao et al., 2021). Likewise, studies evidence a strong connection with nitrogen content. As a matter of fact, a deficiency in nitrogen content implies a reduction in CC which improves leaf transmittance at visible wavelengths. Several findings have demonstrated that this pigment has diverse spectrum behavior with particular absorption properties at different wavelengths, thus the electromagnetic leaf reflection is an indicator of chlorophyll

TABLE 2C Common VIs used in the reviewed articles.

Vegetation Index	Formula	Application	Reference
Triangular Vegetation Index (*TVI*)	$60(NIR - G) - 100(G - R)$	Soil Moisture Content Prediction	Cheng et al. (2022)
Enhanced Vegetation Index (*EVI*)	$2.5\frac{NIR - R}{NIR + 6R - 7.5B + 1}$	Soil Moisture Content Prediction. Forest Structure	Arevalo-Ramirez et al. (2022); Cheng et al. (2022)
Green Index (*GI*)	$\frac{G}{R}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Transformed Chlorophyll Absorption in reflectance Index (*TCARI*)	$3[(RE - R) - 0.2(RE - G)\frac{RE}{R}]$	Soil Moisture Content Prediction	Cheng et al. (2022)
Simple Ratio Pigment Index (*SRPI*)	$\frac{B}{R}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Normalized Pigment Chlorophyll Index (*NPCI*)	$\frac{R - B}{R + B}$	Soil Moisture Content Prediction. Chlorophyll Content Estimation	Cheng et al. (2022); Zhuo et al. (2022)
Normalized Difference Vegetation Index 2 ($NDVI_{GB}$)	$\frac{G - B}{G + B}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Plant Senescence reflectance Index 2 (*PSRI*)	$\frac{B - R}{G}$	Soil Moisture Content Prediction	Cheng et al. (2022)
Color Index of vegetation extraction (*CIVE*)	$0.44R - 0.81G + 0.39B + 18.79$	Soil Moisture Content Prediction	Cheng et al. (2022)
Near Infrared Reflectance of Vegetation (NIR_V)	$NIR{\star}NDVI$	Chlorophyll Content Estimation	Raddi et al. (2021)
Difference Vegetation Index (*DVI*)	$NIR - R$	Fuel Estimation	Villacrés and Cheein (2022)
Modified Soil Adjusted Vegetation Index (*MSAVI*)	$[2NIR + 1 - \sqrt{2NIR + 1} - 8(NIR - R)]/2$	Forest Structure	Arevalo-Ramirez et al. (2022)
Chlorophyll Absorption Reflectance Index (*CARI*)	$RE - R - 0.2(RE - G)$	Forest Structure	Arevalo-Ramirez et al. (2022)

TABLE 2D Common VIs used in the reviewed articles.

Vegetation Index	Formula	Application	Reference
Red Edge Modified Simple Ratio (*REMSR*)	$\frac{NIR/RE - 1}{\sqrt{NIR}/RE + 1}$	Forest Structure	Arevalo-Ramirez et al. (2022)
Red Edge Normalized Difference Vegetation Index (*RENDVI*)	$\frac{NIR - RE}{NIR + RE}$	Forest Structure	Arevalo-Ramirez et al. (2022)
Leaf Chlorophyll Index (*LCI*)	$\frac{NIR - RE}{NIR + R}$	Fuel Estimation	Villacrés and Cheein (2022)
Normalized Difference Red Edge (*NDRE*)	$\frac{NIR - RE}{NIR + RE}$	Fuel Estimation	Villacrés and Cheein (2022)
Red Edge Modified Simple Ratio (*REMSR*)	$\frac{NIR/RE - 1}{\sqrt{NIR}/RE + 1}$	Forest Structure	Arevalo-Ramirez et al. (2022)
Red Edge Normalized Difference Vegetation Index (*RENDVI*)	$\frac{NIR - RE}{NIR + RE}$	Forest Structure	Arevalo-Ramirez et al. (2022)
Leaf Chlorophyll Index (*LCI*)	$\frac{NIR - RE}{NIR + R}$	Fuel Estimation	Villacrés and Cheein (2022)
Normalized Difference Red Edge (*NDRE*)	$\frac{NIR - RE}{NIR + RE}$	Fuel Estimation	Villacrés and Cheein (2022)

content. CC can be altered by natural or man-made noxious agents as well as stress factors. Additionally, an accurate measurement of CC involves a good examination of plant health, regulation of fertilizer application, and so on. CC ground measurements are used as an indicator of fertilizer status (Narmilan et al., 2022). Due to its importance in the agriculture field, current remote sensing efforts contemplate the blending of vegetation indices and machine learning techniques in order to find a well-established model that accurately defines CC (Yao et al., 2021; Narmilan et al., 2022).

2.4 Water and moisture content

Water and moisture content (WMC) is affected by tree species type (Yao et al., 2021) and canopy cover attributes (Gale et al., 2021). It is also a factor of soil respiration. In addition, WMC is associated with the production of CO_2 in soil and the transportation of CO_2 from soil to the atmosphere, so continuous or unexpected changes in WMC can affect soil respiration behaviors (Yao et al., 2021). Likewise, WMC is commonly used to assess wildfire risk in forested areas, (Barmpoutis et al., 2020; Gale et al., 2021) and knowledge of its behavior are necessary for land management decision-making (Barber et al., 2021).

Parameters such as moisture of forest canopy are used jointly with the moisture of the soil-litter layer and forest temperature for the early detection of forest fires. Therefore the development and usage of aerial remote sensing platforms including radiometer sensors, which is useful for determining and classifying areas of forests that are prone to wildfires (Varotsos et al., 2020).

The WMC is highly dependent on temperature changes, so predictive models to estimate WMC are altered by meteorological conditions (Gale et al., 2021). Current efforts are mainly focused on establishing more accurate and affordable measure systems; the most remarkable developments which have enabled effective estimation of WMC are related to the improvement of processing software/techniques and computational power and the availability of aerial imagery from satellite data, airplanes, or unmanned aerial vehicles (UAVs) (Forbes et al., 2022). Furthermore, recent studies have shown that reflectance data in a variety of wavelengths is a promising alternative for WMC estimation (Barber et al., 2021).

2.5 Tree recognition

The tree identification problem is to identify each individual tree from an aerial image. Its importance relies on the fact that tree recognition is a key factor when evaluating biodiversity evolution due to external factors such as climate change and natural disasters (Hologa et al., 2021). Another important application for tree identification is to evaluate the survival rate of seedlings, which is vital to assess the efforts of afforestation, identifying seedlings across several seasons is a difficult task, given the fact that each individual tree crown needs to be identified in a complex vegetation environment (Guo et al., 2021). Forest inventory and mapping are crucial for forest managers, to ensure the preservation of the different habitats (Neuville et al., 2021).

2.6 Tree structure

Tree and forest structure is related to forest biodiversity and productivity (Bohn and Huth, 2017). Tree structure identification is related to the measurement of parameters that help to characterize both individual trees and forests alike. The most common parameters used to characterize tree structure are tree height, diameter at breast height, basal area, total stem volume, crown cover, crown height, and crown area (Lin and Herold, 2016; Shin et al., 2018; Fraser and Congalton, 2021b; Guo et al., 2021; Hologa et al., 2021; Neuville et al., 2021; Terryn et al., 2022). These parameters are strong indicators of forest vigor and forest health when facing stress due to climate change (Fraser and Congalton, 2021b). Tree structure is essential in studies such as forest meteorology, botany, and ecology (Lin and Herold, 2016; Terryn et al., 2022). There is also a correlation between tree structure and the exchange of energy, carbon, and water between the forest canopy and its environment. Figure 2 indicates the most common parameters that are used to assess forest structure

2.7 Tree classification

In the assessment of forested areas, tree species present distinctive traits such as textural characteristics and a specific spectral reflectance; these traits allow researchers to identify each tree species (Zhang C. et al., 2021). One of the purposes of tree classification is to know which tree species are able to regulate temperature and relative humidity in a certain environment, a fact that helps to better understand forested ecosystems (Liu et al., 2021; Zhang C. et al., 2021). Tree species classification is a crucial research topic for effective forest management (Onishi and Ise, 2021). Nevertheless, the most predominant factors that prevent a well-performed tree classification procedure are due to the diversity of tree species and the complexity of land (Zhang C. et al., 2021). Thus, gathering this data usually requires carrying out *in situ* measurements

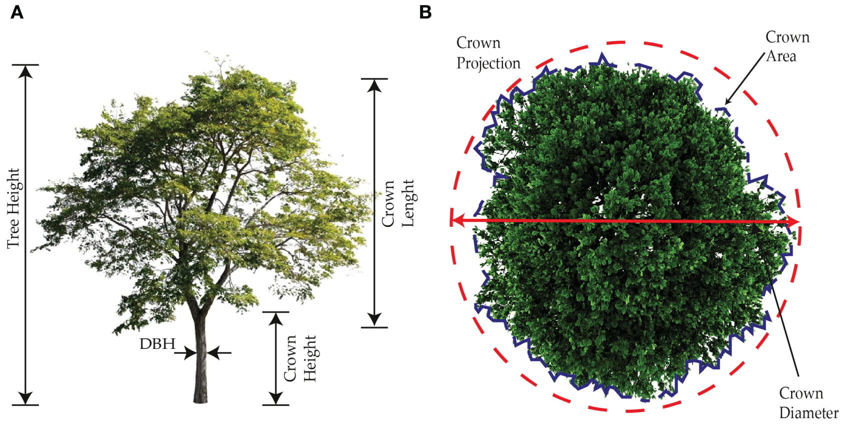


FIGURE 2
Tree structure parameters used to assess forestry health. In **(A)** are shown the parameters from a frontal view, **(B)** shows the parameters from an aerial point of view, focusing on the tree crown.

from sample plots and extrapolating to larger scales (Hell et al., 2022). Overall, this shallow or deep mapping is processed by hand-crafted features or specialized methods (Mäyrä et al., 2021). Currently, there are some new developments in this field, where researchers have introduced novel techniques related to computing various vegetation indices and textural features (Mäyrä et al., 2021), machine learning-based models, deep learning methods to extract tree features (Liu et al., 2021; Onishi and Ise, 2021) and the full use of forest spectral information (Zhang C. et al., 2021). Moreover, sensors and platforms used for this task, have become more specialized in order to capture enough information to accurately assess the type of tree (Mäyrä et al., 2021; Onishi and Ise, 2021; Zhang C. et al., 2021).

2.8 Crown fuel estimation

Several forest fire prediction studies rely on empirical models (Barber et al., 2021) using site-specific information on climate, topography, and fuels (Arkin et al., 2021). This information is strongly important for fire-prone countries in order to predict the impact of fire in certain scenarios. Fuel management programs (Wells et al., 2021) have been considered to reduce fire risk. The behavior of wildfires can be predicted by Crown Fuel Estimation (CFE). CFE is the assessment of fuel hazard layers. CFE is the assessment of fuel hazard due to the spatial arrangement of vegetation elements (branches, leaves, etc.); thus CFE helps researchers assess the severity of wildfires (Hillman et al., 2021), this task plays a key role since canopy fuels are the primary fuel layer of initiation and spread of crown fire (Arkin et al., 2021). It is worth mentioning that an accurate CFE can infer in the total or partial wildfire mitigation (Hillman et al., 2021; Wells et al., 2021). However, to completely assess the risk of wildfire; models including not only CFE but other tree structure parameters are needed; for example, the measure of live crown base height is critical this metric helps to estimate the likelihood of fire propagating from the surface into tree crowns (Arkin et al., 2021).

3 Hardware for remote sensing applications

In remote sensing applications, hardware fulfills vital roles in the data acquisition process, and choosing the correct sensors is critical to the success of the desired task (Müllerová et al., 2021). This section describes the different sensors, imaging systems, and platforms used in the reviewed articles.

3.1 Sensors

Remote sensing platforms include various kinds of sensors for gathering information about the environment. The most common sensors for forestry health assessment include the following: Visible Light Cameras (RGB Cameras), multi-spectral cameras, hyper-spectral cameras, thermal cameras, Laser imaging Detection and Ranging (LiDAR) systems, terrestrial laser scanning systems (TLS), and other common sensors. This section will discuss the working principle of the most common sensors in remote sensing for forestry health assessment. Figure 3, contains a visual representation of the most common sensors used for forestry health assessment.

3.1.1 RGB cameras

RGB cameras capture spectral information in visible light (400-700 nm), which is the same spectrum perceived by the human eye (Idrissi et al., 2022), the working principle of this kind of camera is

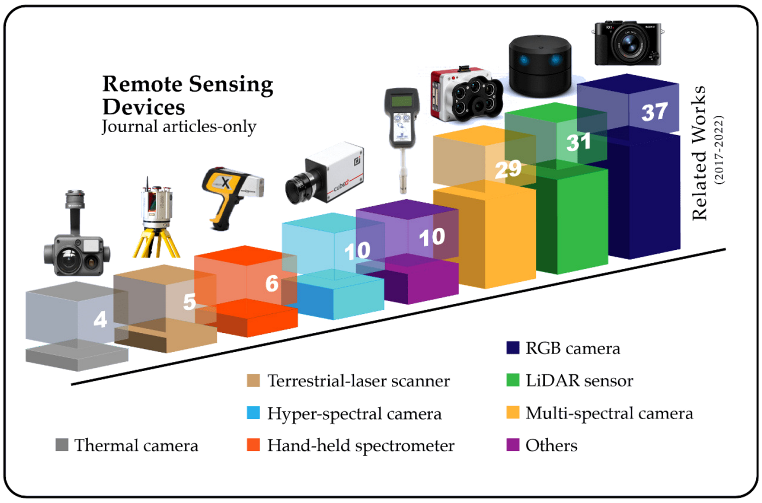


FIGURE 3
Most common sensors used for forestry health assessment, each column represents the number of articles that used each sensor in the data collecting process.

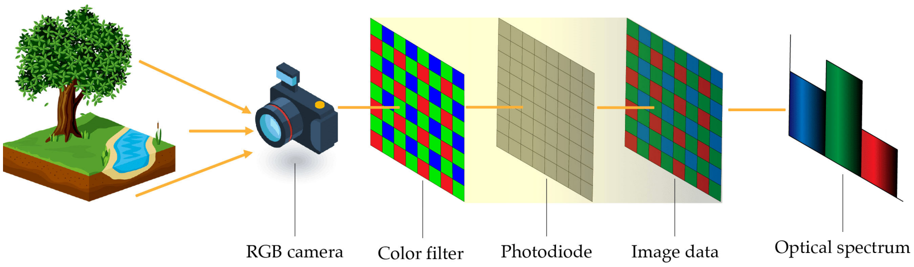


FIGURE 4
RGB camera working principle: a typical image processing system.

visualized in Figure 4. These cameras are designed to represent the real colors of objects and nature using trichromatic red (620 - 750 nm), green (495-570 nm), and blue (450 -495 nm) wavelengths. Overall, RGB cameras provide two-dimensional images (Lin et al., 2022), and their performance tends to decrease in the presence of adverse atmospheric conditions (fog, haze, heat waves, etc.). The quality of an RGB camera is expressed in megapixels, which determine the number of pixels (i.e. length x height) of a static photo (Linhares et al., 2020). RGB cameras have been used for the study of vegetation indices based on RGB information (Ilniyaz et al., 2022; Talavera et al., 2022; Yang et al., 2022), forest canopy mapping and modeling (Nasiri et al., 2022; Suwardhi et al., 2022; Trencanová et al., 2022), tree identification and characterization(Onishi and Ise, 2021), and among others.

3.1.2 Multi-spectral cameras

Multi-spectral cameras collect color data and spectral monitoring. They capture two or more bands in the visible and invisible spectrum (Akkoyun, 2022). These cameras are able to cover parts of the infrared and ultraviolet regions. The most common wavelengths for these cameras are the Near-infrared wavelength (NIR) and red-edge wavelength from the infrared spectrum. Likewise, multi-spectral cameras hold a sensitive area detector used in conjunction with a series of specific waveband filters or a waveband tunable light source (Ramirez et al., 2022). The working principle of a multi-spectral camera is shown in Figure 5, with a visual representation of an image expected from this camera. In forestry health assessment, multi-spectral cameras have been used to obtain spectral indices and the derived applications as seen in previous sections.

3.1.3 Hyper-spectral cameras

Hyperspectral sensors capture the radiation emitted by bodies in many bands, that go from hundreds up to thousands of wavelength bands, with narrower bandwidths than multi-spectral cameras (from 5 to 20 nm). Other sensors like RGB or Near-infrared (NIR) cameras only capture a minor number of bands (three in the case of RGB) (Adão et al., 2017). A comparison of multi-spectral and hyper-spectral cameras is shown in Figure 6, the main difference is that the hyper-spectral captures a continuous representation of the light spectrum, given the fact that it collects the reflectance in narrow bands; but the multi-spectral cameras only capture the reflectance in a selected number of bands.

Hyperspectral cameras have been used in forestry, to obtain new VIs to predict vegetation features such as leaf nitrogen content (Raj et al., 2021), chlorophyll, and other photosynthetic plant traits (Watt et al., 2020). Mapping forest hyperspectral characteristics have been performed as well (Weinstein et al., 2021). The main advantage of using hyperspectral cameras is the increased number of wavelengths, thus more information is gathered about the environment, however, the models created using this information might be overfitted and thus not usable in general cases (Lee et al., 2004).

3.1.4 Infrared cameras

Infrared cameras are a specific type of sensor that captures the infrared radiation that is emitted by all bodies with a temperature above absolute zero. The range of wavelengths that is captured by these sensors depends on the nature of each one, but common wavelengths are Short-wave Infrared (SWIR) that ranges from 700 to 1400 *nm*,

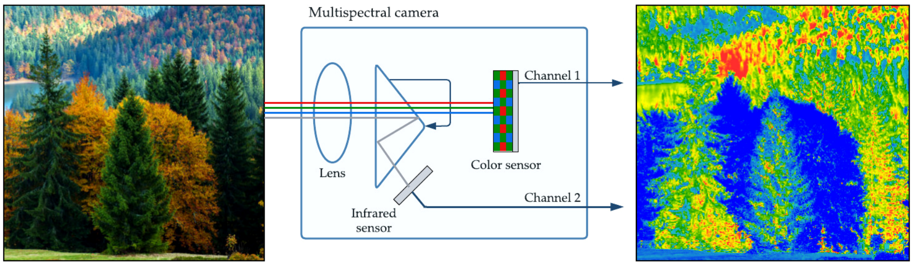


FIGURE 5
Multi-spectral imaging: camera structure and a sample of spectral forestry images.

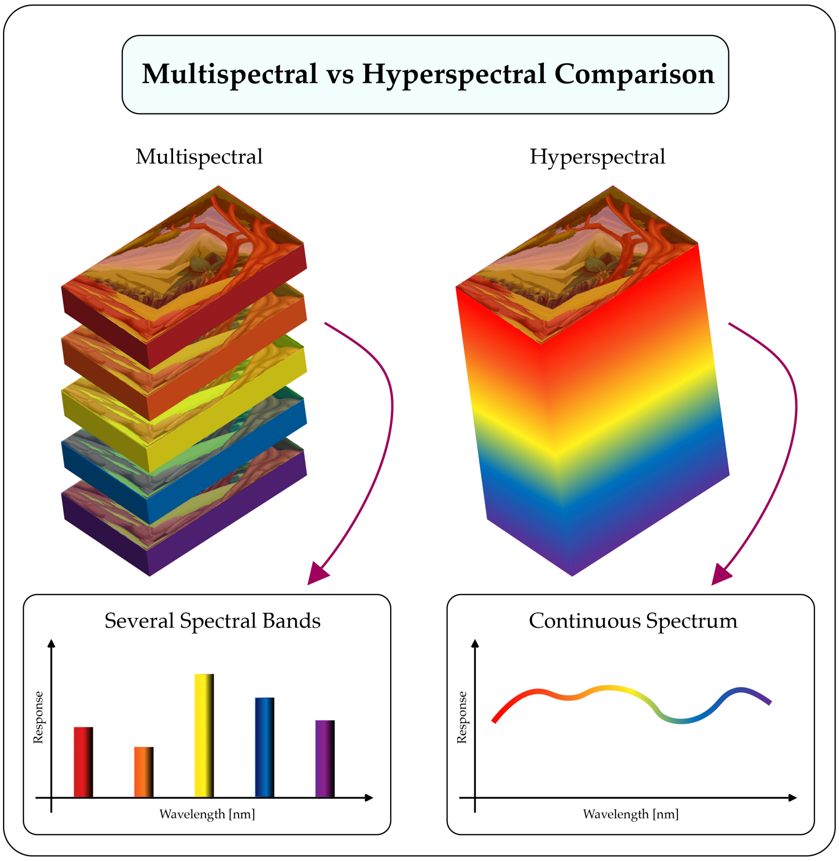


FIGURE 6
Comparison between Multi-spectral and Hyper-spectral camera operation. The multi-spectral camera presents a discrete and reduced number of bands, however, the hyper-spectral camera presents a continuous spectrum that ranges from wavelengths of 5 to 20 nm.

Mid-wave, infrared (MWIR) from 3000 to 5000 *nm*, and Long Wave infrared (LWIR) that ranges from 8000 to 14000 *nm* (Gade and Moeslund, 2013), these sensors are also known as thermalcameras in the reviewed studies (Xu et al., 2018; Cheng et al., 2022).

Figure 7 shows the common structure of a thermal camera used in remote sensing applications. These sensors have been used in forestry health assessment to create thermal mappings that are coincident with RGB mapping information (Webster et al., 2018). Other applications include the use of thermal indices to predict soil moisture (Cheng et al., 2022) and for phenotyping (Xu et al., 2018).

3.1.5 LiDAR sensor

LiDAR (Light Detection And Ranging or Laser Imaging Detection and Ranging) sensor is a device widely used for remote sensing. It is considered an active device due to its light emission and detection (See Figure 8 for comparison with passive sensors). Moreover, this sensor has two key elements to gather and analyze data: photodetector and optics. The principle of LiDAR is to emit laser light towards an object on the Earth's surface and compute how long it takes to return to the LiDAR emitter, this definition holds for an airborne-based LiDAR system (Khairul and Bhuiyan, 2017).

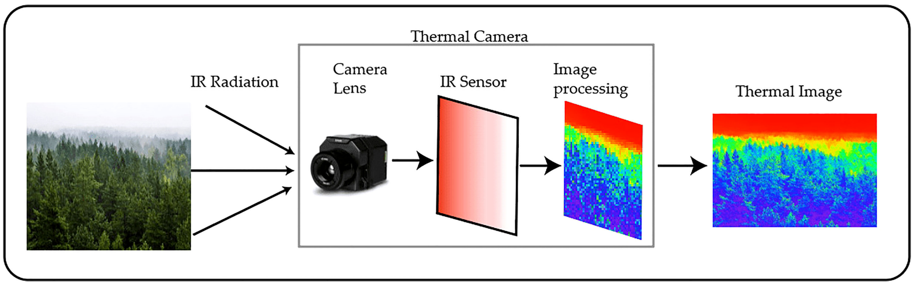


FIGURE 7
Internal Structure and expected forestry image from a thermal camera.

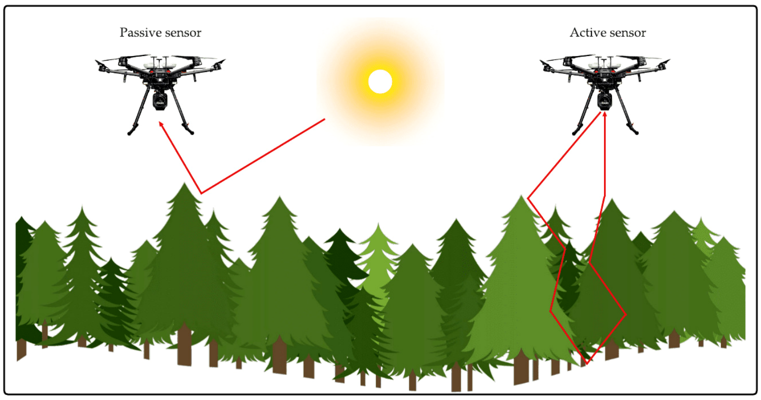


FIGURE 8
Differences between Passive sensors and Active sensors.

The LiDAR point cloud is useful for obtaining physical information about the surveyed area, the 3D measurements can be used for generating terrain models, then by processing the LiDAR point cloud information digital terrain models and digital elevation models can be retrieved by thresholding the altitude of each point and discerning which point can be considered from terrain or from the top tree crowns. With this information, elevation models are easily obtained by subtracting the digital elevation modelsand digital surface models surface models (Hologa et al., 2021). LiDAR point clouds are also useful for obtaining geometric features of vegetation as slopes or texture information; these metrics are then used as input data for machine learning models with various tasks for example (Hologa et al., 2021), uses geometric descriptions of vegetation obtained from a point cloud to perform tree classification, a similar approach is done in (Hell et al., 2022). Due to the resolution that the LiDAR point cloud is capable of generating, individual trees can be identified, and thus tree metrics can be directly computed. In (Vizireanu et al., 2020; Neuville et al., 2021), DBH is estimated based only on LiDAR retrieved data, other forest attributes estimated by LiDAR cloud points are canopy cover (Cai et al., 2021), which can be derived through the density of vegetation points, this metric is also used to predict biomass near rivers (Resop et al., 2021), and with the purpose of determining crown fuels (Suwardhi et al., 2022). Morphological features derived from LiDAR point cloud can be key factors to determine and differentiate between alive trees and snags or deciduous and evergreen trees, this study is done by Stitt et al. (2022). The use of LiDAR has helped researchers investigate the following: tree modeling (Suwardhi et al., 2022), biomass estimation (Torre-Tojal et al., 2022), and tree classification (Hell et al., 2022) among others.

3.1.6 Terrestrial laser scanning systems

Terrestrial laser Scanning Systems (TLS) are instruments used to obtain three dimension observation of the surface of objects. It uses LiDAR sensing to obtain the distance from the surface to the sensor, and precise angular measurements to obtain 3D information from the objects. TLS systems are capable of reconstructing an area with high precision in the order of millimeters (Liang et al., 2016). A representation of the TLS and its measurements are shown in Figure 9. In forest health assessment, TLS systems are used to determine tree features and structure (Miraki and Sohrabi, 2021; Terryn et al., 2022; Yang et al., 2022), and to estimate crown fuel and fuel hazard (Hillman et al., 2021).

3.1.7 Handheld spectrometer

A handheld spectrometer is a device that is capable to retrieve the spectrum emitted by a body in many wavelength bands, the same as a hyper-spectral camera, but this one is portable and operated by hand. Another difference is that a hyperspectral camera captures many pixels, and the spectrometer only captures a single point. The main application for this device is to obtain samples of an object that will serve as ground truth for mass data obtained with a camera or by other means. Handheld spectrometers have been used to gather information to estimate leaf water content (Li et al., 2021), to monitor the chlorophyll response to droughts (Raddi et al., 2021), and to perform tree recognition based on hyper-spectral features (Yang and Kan, 2022).

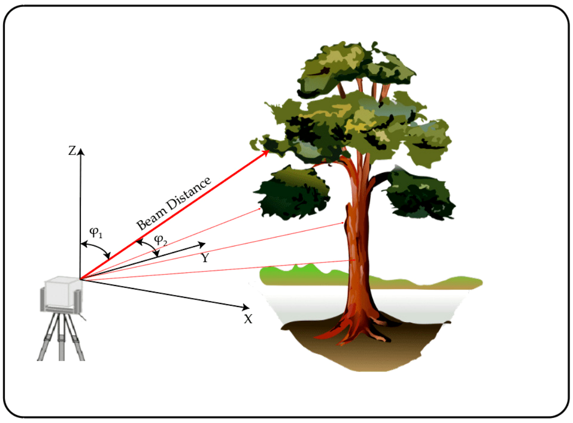


FIGURE 9
TLS sensor variables needed for obtaining 3D cloud points.

3.1.8 Others

There are other kinds of sensors used for forestry health assessment. For instance, an ANAFI camera (Ribas Costa et al., 2022), wireless sensors (Yang et al., 2022), a thermocouple (Yao et al., 2021), and a SPAD-502 meter (chlorophyll meter) (Yao et al., 2021; Narmilan et al., 2022). These sensors are used for very specific scenarios, such as measuring chlorophyll in a single leaf, and thus are not considered for further revision in this review.

3.2 Remote sensing platforms

This section presents a brief review of the most common remote sensing platforms; highlighting their advantages, disadvantages, and applications; for a most extensive review on the topic, see (Omasa et al., 2006; Ashraf et al., 2011; Pajares, 2015; Toth and Jó´zków, 2016; Zhang K. et al., 2020; Chamola et al., 2021; Zhao et al., 2022).

Remote sensing platforms are understood as the platforms that physically carry the different cameras and sensors used for the assessment of forestry health. There are two major groups of platforms that are identified: Unmanned Aerial Vehicles (UAVs) and satellites. Figure 10, summarizes the number of appearances that the different remote sensing platforms have in the reviewed articles. Figure 11 shows a remote sensing platform using a UAV.

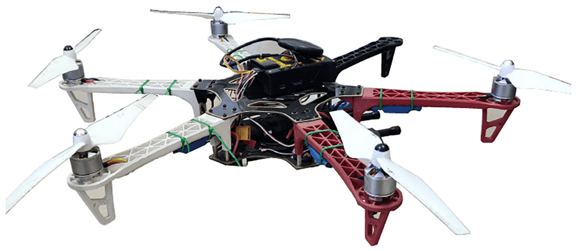


FIGURE 11
A remote sensing platform mounted on a hexacopter.

3.2.1 Satellites

Satellites are commonly used for remote sensing purposes (Zhao et al., 2022). These devices are aimed at gathering data from Earth using imaging sensors. Satellites tend to capture electromagnetic radiation in the microwave, ultraviolet, and visible wavelengths reflected by the Earth's surface (Ashraf et al., 2011). Overall, a remote-sensing satellite is able to take 4-5 photos with different types of color filters, evidently, these color filters help to better assess vegetation features such as soil, leaves, stems, tree crowns, under/over the canopy, and so on.

Satellites carry onboard high-resolution microsatellite cameras (HR-250 and Raptor imagers) with advanced electronic detectors known as CCDs (Charge-Coupled Devices). These devices not only allow them to be more sensitive than a film but also convert the multispectral photographs into electronic signals for further study (Zhang K. et al., 2020).

According to the literature reviewed, Sentinel 1 and 2 (Huo et al., 2021; Nasiri et al., 2022), Landsat-8 (Rodríguez et al., 2021), Worldview-2 (Becker et al., 2018), Triplesat (Fakhri et al., 2022) are the most prominent satellite platforms used to assess forestry health.

3.2.2 UAVs

Unmanned Aerial Vehicles are the most common platforms in remote sensing applications for forestry health assessment. The typical UAV for remote sensing is an electric-propelled air vehicle, with a navigation system and communication system, and a sensor for remote sensing (Toth and Józków, 2016). The navigation and flight control systems are composed of various onboard sensors in

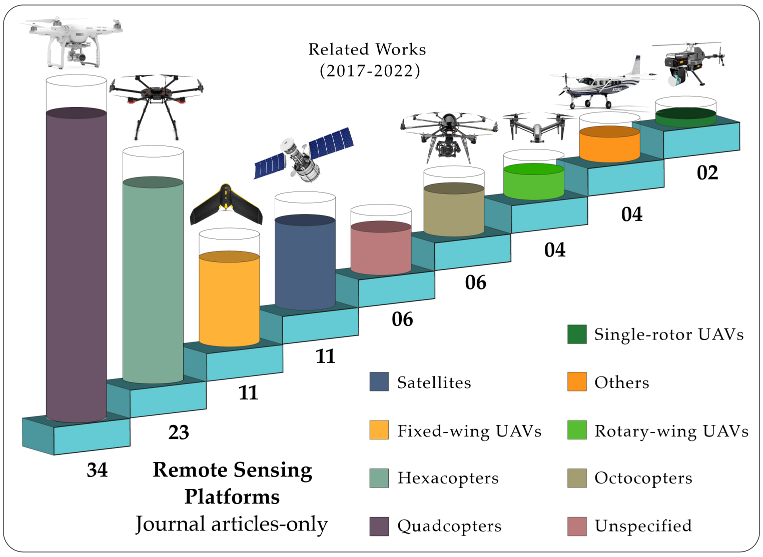


FIGURE 10
Distribution of the most common UAV platforms in the reviewed journal articles.

the UAV, the main ones are: Global Positioning System (GPS), an Inertial Measurement System (IMU), and Micro-Electromechanical System (MEMS) (Toth and Jóźków, 2016). The other components of the remote sensing platform are the sensors needed for the data acquisition process, the most common sensors in remote sensing applications are the ones mentioned in section 3.1.

There are different kinds of UAVs, and according to their configuration, they offer different features such as higher payload capability, longer flight capacity, and better maneuverability among others. We have identified the following classes:

3.2.2.1 Single-rotor

Single-rotor UAVs are formed by a single rotary wing, they are a minority compared to other remote sensing platforms. Since they only present a single rotor they present a much higher power efficiency compared to multi-rotor UAVs, they are also used for carrying heavy payloads (Chamola et al., 2021).

3.2.2.2 Multi-rotor

Multi-rotor UAVs are the most versatile and have been used in a wide range of operations. This group includes quadcopters, hexacopters, and octocopters. The main advantages of using these UAVs are their commercial availability and affordability, the ease of maneuverability, they don't need a platform to take off, meaning that they can take off and land on any surface; so they are preferred for research purposes. The arrangement of multiple rotors provides the UAV with better stability making them ideal for imaging purposes (Toth and Jóźków, 2016; Chamola et al., 2021).

3.2.2.3 Fixed-wing UAV

These UAVs present a stationary wing, similar to a plane, the advantage of using a fixed-wing is that lift forces are lower compared to rotary wing UAVs. Since they are similar to a plane they need some area for the takeoff and eventual landing. The main advantage of fixed-wing drones is that they can fly for longer periods of time, cover larger areas, and can carry heavier payloads (Chamola et al., 2021).

3.2.2.4 Aircraft

Forestry studies have evoked their efforts to incorporate remote sensing aircraft into the dynamics of forest surveys and data collection. Aircraft remote sensing platforms rely heavily on onboard sensors to leverage their advantages associated with flexible use and high spatial resolution. In addition, images captured from the aerial inspection can be used for rapid analysis in different seasons of the year (Omasa et al., 2006).

4 Machine learning techniques used in forestry health assessment

Machine learning is a set of algorithms that require the computer or machine to infer and extract patterns from raw data (Goodfellow et al., 2016); the effectiveness of machine learning heavily depends on the representation of the data fed to the model. These algorithms can be used for regression tasks, which implies predicting a number from a set of input data; classification problems can also be accomplished by machine learning, in this case, the algorithm predicts that the data representing a feature belongs to a predefined class.

Learning is a key concept in machine learning, it can be performed in these ways:

4.1 Supervised learning

In supervised Learning algorithms, the dataset containing features also contains a number or a label that is the expected output from the input features. In this case, the machine learning algorithm needs to infer which is the relation between the set of features and the expected output, then apply these found relations in a set of testing data (Goodfellow et al., 2016).

4.2 Unsupervised learning

In these algorithms, the dataset contains a set of features and the algorithm learns properties about how the data is structured, a common task performed by unsupervised learning is to recreate the probability distribution that generated the dataset; another common function is to group data into clusters with similar characteristics (Goodfellow et al., 2016).

4.3 Metrics

It is important to measure how the machine learning algorithm is performing its task, thus it is important to describe the most common metrics to quantitatively evaluate the algorithm's performance. The following are the most used metrics for classification purposes:

4.3.1 Accuracy

It can be defined as the ratio between the number of correct predictions and the number of total predictions made by the model (Flach, 2019), it can be calculated with Eq. (1)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Where TP, TN, FP, and FN stand for True Positive, True Negative, False Positive, and False Negative respectively.

4.3.2 Precision

It is the ratio between correct positive predictions and total prediction, it indicates the proportion of how many correct predictions the model yields, it is calculated with Eq. (2)

$$Prec = \frac{TP}{TP + FP} \tag{2}$$

4.3.3 Recall

It measures the ratio of correct positive predictions and the total predictions, it is obtained with Eq. (3)

$$Rec = \frac{TP}{TP + FN} \tag{3}$$

4.3.4 F1 Score

It is a metric that combines both Precision and Recall, it is useful when the classes in a dataset are unbalanced, and it is computed with Eq. (4)

$$F_1 = 2 \cdot \frac{Prec \cdot Rec}{Prec + Rec} \tag{4}$$

4.3.5 Root mean square error

It is a measure of the error between the predicted output of the model and the real output of the model. This metric is used for evaluating regression models. It is computed with Eq. (5)

$$RMSE = \sqrt{\frac{-_{i=1}^{N} ||y(i) - \hat{y}(i)||^2}{N}} \tag{5}$$

4.3.6 Correlation factor (R^2)

It is a number that indicates if there is a correlation between two variables, in regression models it is a metric that helps to understand if the output of the model is correlated with the input. It ranges from 0 to 1, where 0 indicates that there is no correlation between the variables and 1 that there is a high correlation.

With the previous remarks, the section continues describing the most common machine-learning techniques used in the reviewed articles for the assessment of forestry health and the most critical results supported by quantitative metrics, the discussed algorithms in the section are: Linear Regression, Random Forests, Support Vector Machines, K-Nearest Neighbours, deep learning approaches and other not common machine learning techniques. Figure 12 shows the most common ML algorithms used in forestry health assessment in articles from the last five years. Figure 13 shows a visual representation of how three of the most common ML methods divide the search space for classification purposes.

4.4 Linear regression

Linear Regression is one of the most common algorithms in machine learning, for predicting results. Using an optimization process,linear regression determines the appropriate equation that maps the input features with the expected output (Goodfellow et al., 2016). Linear regression has had a wide range of applications. It has been used to find the correlation between the data derived from TLS and airborne LiDAR; the study presented by Hillman et al. (2021) demonstrated that estimations of canopy volume have a strong correlation between the data from LiDAR and TLS which achieved a value of 0.96, herein the ground-truth is the value obtained from the TLS sensors, however other tree structure parameters such as canopy base height achieved only a correlation of 0.794. In other studies canopy height volume reached a correlation of only 0.394, thus it is not suitable for predicting crown fuel (Shin et al., 2018), similar experiments were conducted by Arkin et al. (2021). For predicting the moisture of leaf fuels, multi-spectral VIs were used as input data for regression models, however, the correlation factor reached 0.435, thus more studies are needed for practical implementations for this model (Barber et al., 2021).

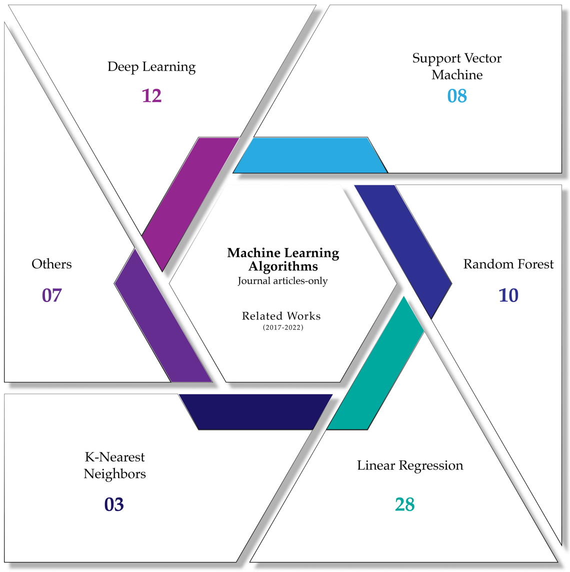


FIGURE 12
Distribution of the most common machine learning algorithms for forestry health assessment in the reviewed journal articles.

Other vegetative problems are investigated using linear regression models. Resop et al. (2021) studied the correlation between vegetation metrics, the distance from water sources, and seasonal variation; the results show that there is no correlation between the distance to the water stream and canopy height and vegetation density. Using multi-spectral VIs, regression models have been used to predict biomass in the tidal marsh; the best VI was ExG however the correlation index only reached 0.376 (Morgan et al., 2021). In coastal wetlands, the correlation between above-ground biomass and flood depth was studied, and the regression models follow a Gaussian distribution with a correlation factor of 0.54 (Yan et al., 2022). Xu et al. (2022) studied the correlation between tree diversity and spectral indices. The correlation value was 0.6; thus VIs could be used for tree classification purposes.

Estimating the correlation between tree features and point cloud LiDAR data information, in the work presented by Fraser and Congalton (2021a) RGB and LiDAR-derived metrics of DBH and crown radius were studied in a coniferous forest. The results show a correlation of 0.392 and RMSE which equates to 30% of the total error. Fan et al. (2020) created tree models derived from LiDAR point clouds, and then structure metrics were calculated, the predictions were correlated with the ground truth collected *in situ*, and the linear models achieved a correlation of more than 0.9 for DBH, tree height, and crown volume. In the article by Imangholiloo et al. (2020), tree height was estimated using LiDAR

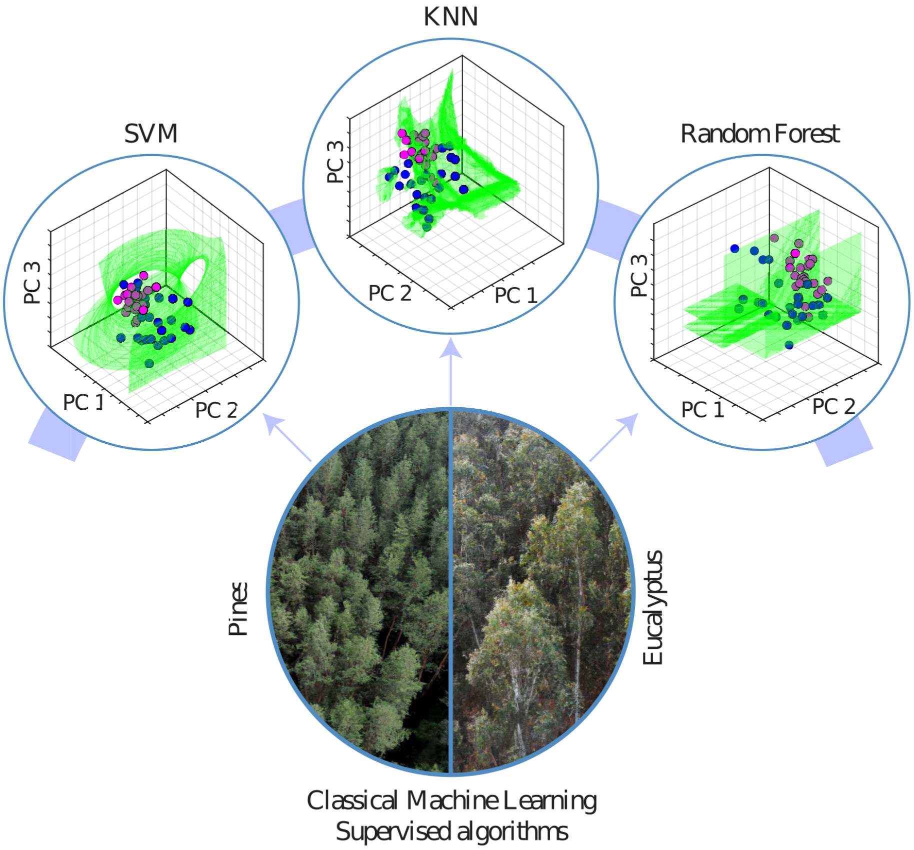


FIGURE 13
A comparison between the most common machine learning methods, and how the space is divided to generate different classes.

metrics, such as point density, in leaf-on and leaf-off seasons, and the correlation factor achieved 0.98. A similar study conducted by Puliti et al. (2019), compared tree height, stem volume, and basal area; from data obtained via different aerial methods (UAV, and manned aircraft); the results show correlation values in the range of 0.64 and 0.73. Another study combined RGB images and LiDAR metrics to predict tree height and DBH in a eucalyptus forest, combining both metrics as input data for the model achieved a correlation of 0.94 (Liao et al., 2022). Xu et al. (2021) developed a remote sensing platform and the method of validating its data was to find the correlation of tree structure parameters with the ground truth found in the field, this study also contemplated the creation of thermal and multi-spectral VIs.

Leaf area index (LAI) is another parameter that can be predicted using LiDAR metrics and linear regression models. In the work by Tesfamichael et al. (2018) the highest correlation value was 0.83; however, this model used several metrics as input data; a simple model using only two metrics achieved a correlation of 0.63 but the simplicity of the model was considered an advantage. A similar study using RGB point clouds for calculating LAI was conducted by Lin L. et al. (2021), and the models achieved a correlation of 0.92. Miraki and Sohrabi (2021) estimated LAI from RGB images and terrain model descriptors as input data, but the correlation was only 0.42, in the same study canopy height was also estimated, and using linear regression models the correlation achieved was 0.84. The study presented by (Qiao et al., 2022) also considered morphological features from the soil and the vegetation to improve the prediction of LAI, achieving correlation values of 0.93 but it depends on the growth stage of the vegetation. Water and transpiration models are also associated with LAI and canopy volume; Aboutalebi et al. (2019) estimated these parameters using information derived from airborne LiDAR and multi-spectral cameras; the LAI derived by machine learning achieved correlations of 0.7.

Predicting the chlorophyll changes in response to environmental changes has been explored with the aid of regression models. In the study presented by Raddi et al. (2021), using hyper-spectral indices and multi-spectral indices; leaf chlorophyll content in textit Quercus Robur, Quercus Pubescens, and *Quercus ilex* was estimated with the aid of linear regression models; using both kinds of indices achieved a correlation of 0.97 in both cases, thus providing an excellent alternative to assess drought responses using the change of chlorophyll content as an indicator. Zhuo et al. (2022), conducted a similar study to predict chlorophyll content, however, it considered the effect of mixed vegetation in wetlands for the computation of the spectral indices, in this case, the model reached a correlation of 0.82. (Kopacková-Strnadová et al., 2021) presented a study aimed to predict photosynthetic pigments in coniferous Spruce forests, using multi-spectral VIs; however, the researchers showed that information from the growth stage of the forests is needed since the spectrum from two years' leaves was the only VI that reached a correlation factor of 0.52 in a linear regression model. Watt et al. (2020) conducted a similar procedure but with the purpose of predicting nitrogen and phosphorus. Using hyperspectral VIs, regression models were trained and the predictor for both P and N achieved correlations of 0.75 and 0.83 respectively. Other studies predicting chlorophyll in different ecosystems are done by Narmilan et al. (2022) and Yao et al. (2021), with the purpose of evaluating soil respiration; estimating nitrogen can be achieved with regression models and RGB indices (Lu et al., 2021).

Problems related to moisture content, in general, can be performed using linear regression. In the work presented by Li et al. (2021), leaf water content estimation was performed using hyper-spectral VIs in various growth stages of vegetation reaching a correlation factor of 0.9 with the appropriate VI. Regression models were also used to assess water evaporation models and trace element uptake by trees growing on red gypsum landfill (Malabad et al., 2022). Cężkowski et al. (2020) used thermal indices used to predict various indicators of water stress in wetlands (soil moisture, chlorophyll content, and photosynthetic active radiation (fAPAR)), the correlation factors for soil moisture and fAPAR were of 0.62 and 0.70 respectively, thus the index could be an indicator of water stress.

4.5 Random forest

Random Forest is a machine learning method that combines multiple tree classifiers. Each tree is tested with a random input vector, which leads to selecting the most significant features from the input data. Random Forest can be used for classification and regression problems (Breiman, 2001).

For classification purposes random forest has been used in conjunction with information derived from LiDAR point cloud and with multi-spectral indices derived from spectral imagery; this approach presented by Hologa et al. (2021) was used to perform individual tree classification in a mixed forested area, the trained random forest achieved an accuracy of 96% over eleven different tree species when combining both inputs from LiDAR and multi-spectral imagery. A similar approach was done by Fraser and Congalton (2021a), but in this case, due to the nature of the forest, the classification task using random forest achieved an accuracy of 85%, but the authors highlight the capability of random forest over traditional methods for tree delineation. Imangholiloo et al. (2020) used random forest for classification between coniferous and deciduous trees from information obtained by LiDAR. In the work presented by Miyoshi et al. (2020), the input data included hyperspectral multi-temporal imaging data to perform tree classification in a diverse tropical forest, even though the accuracy only reached 50%, the use of multi-temporal imaging improved previous approaches using random forests as classifiers, leaving the door open to future researches in the same field.

Fraser et al. Fraser and Congalton (2021b), performed a classification of forest stands in three different categories: healthy, stressed, and degraded trees; for this purpose, VIs from multi-spectral imagery were derived and they were used to train the RF model; the accuracy achieved a maximum of 71%, due to the fact that there is a high variation in the characteristics of each healthy tree.

Classification tasks are not only needed to differentiate between tree species. Another important task is to classify between live trees and dead trees, the reason being this ratio is important for assessing the response of the ecosystem to external disturbances; Stitt et al. (2022) used information derived from a LiDAR point cloud to classify different kinds of snags, the model achieved an accuracy of 77%, signifying that only LiDAR information is not enough to identify some characteristics of snags. In the work by (Shovon et al., 2022), the RF algorithm was trained to segment between alive and dead trees in forest stands with an accuracy of 89.4%, using as input variables tree height derived from LiDAR point clouds and RGB spectral indices.

Identifying forest structure can be achieved by using random forest, Yu et al. (2021) explored the feasibility of using multi-seasonal data from LiDAR and multi-spectral images to perform vertical forest structure classification. The results show that adding information from different seasons as input variables to the models increases its performance and its capability of reliably identifying the forest structure, even though the random forest was not the best algorithm according to the metrics presented.

Individual tree recognition can be accomplished by random forest. Guo et al. (2021), with the purpose of assessing afforestation models, trained random forests methods to recognize areas of interest that could potentially be identified as tree crowns, for this purpose several VIs were computed from RGB images and they were used as training data for the random forest algorithm; the individual crown recognition task achieved an accuracy of 92%, when using more than two input variables to train the model.

Random Forests were also used for regression purposes. In the work presented by Lou et al. (2021), the feasibility of predicting canopy chlorophyll content in marsh vegetation was evaluated using multispectral images from UAVs, and from satellite platforms including Landsat-8 and Sentinel-2. The predicted canopy from the random forest was validated with the real value through a linear regression achieving a correlation value of 0, 92. Villacrés and Cheein (2022) used random forests to retrieve spectral VIs from multispectral imagery essential for mapping moisture content, however, the results were unsatisfactory, and other regression methods were needed.

Biomass prediction using Random Forest was explored by Torre-Tojal et al. (2022), for this purpose, a LiDAR point cloud was obtained using a UAV; subsequently, digital terrain models and canopy models were reproduced. Some of the metrics obtained were height distribution, canopy cover, and canopy height. An analysis of the importance of those metrics was performed resulting in that the metrics related to the height of the trees were the most significant when describing biomass; using these variables the RF was trained, and the predicted result of the model achieved a correlation value of 0.7, improving previous estimations. Indices and aerial images from satellite platforms are also promising sources of data for prediction purposes, Nasiri et al. (2022) used Sentinel-2 derived Vegetation indices with the purpose of mapping canopy cover in forested areas using Random Forest Regression to predict the percentage of canopy according to the indices, the trained model achieved a correlation of 0.69, showing the potential of combining satellite platforms and random forest for mapping purposes. Sentinel-2 imagery was used to predict the biomass of fine fuels in dryland ecosystems, and the training of the random forest yielded a correlation factor of 0.63 over a six-year period, highlighting the potential of machine learning techniques for mass land estimation of fine fuels (Wells et al., 2021).

4.6 Support vector machines

A support vector machine is a method mainly used for classification purposes, the objective of the SVM is to find a hyperplane that divides in the "best way" two different classes of data. The "best way" refers to the fact that the distance between the hyperplane and each class is maximum (Goodfellow et al., 2016). The main advantage of SVM is that it uses a kernel function that assigns the input data to a higher dimensional space, where it is easier to find the hyperplane that separates two classes.

In forestry health assessment SVMs are used to perform classification and regression tasks. In (Mäyrä et al., 2021), SVMs are used to perform the identification of tree species, using as input vectors point clouds from LiDAR and images from hyperspectral cameras from the SWIR region with 288 bands. From the point clouds, individual tree segmentation was performed and the SVMs were trained. This study shows that there are no major errors in tree classification processes using SVM, achieving an accuracy of 82%; although this method is outperformed by deep learning approaches (Mäyrä et al., 2021), which achieved an accuracy of 87%.

The work by Blanco-Sacristán et al. (2021) uses SVM to perform segmentation in images based on RGB and multi-spectral images. Images were segmented based on their level of dryness, it is important for monitoring possible fire-prone lands. The accuracy reached 80% in most cases.

Tree structure classification has also been studied with the aid of SVM (Yu et al., 2021), predicting the tree structure in a densely forested area, for this purpose the authors used LiDAR and Multi-spectral point clouds to generate height models which were used as inputs to the SVM, in this case, the classification from the SVM was outperformed by other methods. SVMs are used to evaluate carbon models from tree parameters such as canopy height and DBH (McClelland et al., 2019).

The segmentation of ground points based on VIs can be considered as a classification algorithm, in this context Zhang Y. et al. (2021) used vegetation indices as input data for SVM with the purpose of classifying ground points and vegetation points in aerial images; this method achieved an accuracy of 94% using only two VIs as input.

As a regression technique, Support Vector Regressor (SVR) was used to predict tree structure parameters such as DBH, tree height, and volume using as input data high-density LiDAR point clouds (Corte et al., 2020). The results show that the errors in the prediction were lower when using SVR, compared to other algorithms such as RF or neural networks. Nasiri et al. (2022) processed VIs derived from Sentinel-2 information to model canopy cover, achieving significant correlation values of 0.64. A similar task was performed by Abdollahnejad and Panagiotidis (2020), but the tree classification was performed with inputs from multi-spectral VIs.

4.7 K-nearest neighbors

K-nearest Neighbors is a non-parametric machine learning technique, which means that the training does not generate the optimum parameters for a mapping function or plane. It simply is a function of the training data, in its simplest form, KNN computes the expected output value from a new input, by averaging the output from the K nearest neighbors in the training data of this new entry (Goodfellow et al., 2016).

The KNN algorithm was used to perform tree classification from hyper-spectral information. In the work presented by Yang and Kan (2020), the input vectors were information from hyper-spectral imaging, in this case, the KNN algorithm was the least effective algorithm. Tuominen et al. (2017) used KNN to estimate tree structures from the information gathered manually in plots and predict them in aerial photos, the results show that the error is below 30 percent. Another use of KNN algorithm is presented by Zhang Y. et al. (2021), the model was used to segment ground points from vegetation points, however, this model was outperformed by SVM.

4.8 Deep learning

Deep learning (DL) refers to techniques that rely on multiple layers of units (called neurons). Each neuron is a function that maps the input data to the desired output. In the training process, the network is capable of learning the parameters of such mappings. Figure 14 shows the scheme of a network with two hidden layers. The name "deep" refers to the number of layers employed in these kinds of models (Goodfellow et al., 2016). The key feature of a deep learning model is its capability to make representations of unstructured data such as images or raw text (Osco et al., 2021).

Likewise, DL models are used in conjunction with RGB, multi-spectral, and hyper-spectral images, to perform different tasks concerning the assessment of forest health. Lin and Chuang (2021) used deep convolutional neural networks ResNet50, VGG19, and SegNet to extract features from aerial RGB pictures to perform tree classification. However the initial results showed poor performance based on accuracy; thus the authors proposed a simplification of the images using Principal Component Analysis, selecting only the most important features of the images. With this approach, SegNet reached an accuracy of 95%. The same task was performed by Onishi and Ise (2021), from aerial RGB images individual tree crowns were segmented, and each individual tree crown was used as the input data for the deep learning model, which was capable of categorizing seven different tree species and achieved an accuracy over 90%. Here the deep learning architectures were AlexNet, VGG16, Resnet18, and Resnet152, these were used for fine-tuning the model. A similar approach was done by Zhang C.

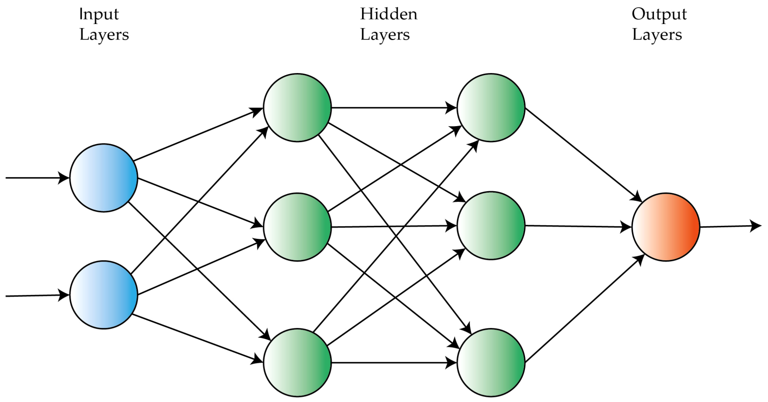


FIGURE 14
Visual representation of a neural network with two hidden layers.

et al. (2020), where a model using ResNet50 achieved an accuracy of 92.6%. In the work presented by (Feng et al., 2020), the authors investigated the results of using multi-temporal information in a recurrent convolutional neural network, for mapping vegetation using multiple-seasons aerial images. Hell et al. (2022) used PointCNN and 3DmFV-NET to perform the classification of coniferous, deciduous, and dead trees; from a LiDAR 3D cloud point, the results show that both networks are capable of differentiating between coniferous and dead trees, and it can reach an overall accuracy of more than 80%.

Pulido et al. (2020) used segmentation networks DetectNET, Faster R-CNN, and Single Shot Multibox Detector (SSD) to perform tree recognition from multi-spectral images in a forested area. The results show that, while traditional methods are capable of identifying trees, DL models outperform them and show improved metrics in areas where trees are clustered together. A similar task was performed by Hao Z. et al. (2022), herein the authors used Mask region-bases convolutional neural networks (Mask R-CNN) and evaluated the effect of reducing the number for training. The results show that by randomizing the training dataset, thus training the model with dissimilar samples each time, the metrics of the model are not as affected; therefore the training images can be reduced.

The creation of segmented images of fire-prone vegetation areas can be achieved with the use of deep learning techniques, Trencanová et al. (2022), trained U-NET network to identify these areas from RGB images, and the results show an F1 score of 0.7 in the validation dataset; however, due to the complex labeling process, the authors suggest that further improvements are needed to enhance this technique of identifying areas in landscapes.

Liu et al. (2021) proposed a 3D deep learning structure called LayerNet to perform tree classification tasks, the network used as input individual tree point clouds obtained from a LiDAR point cloud, the advantage of the network is that it can be trained from disorganized 3D point clouds. Compared to other algorithms such as random forest or KNN, this method achieved an accuracy of 88%, greatly outperforming the other two more common methods, which also need to pre-process the information to reduce the dimensions of the data, thus reducing potentially valuable traits.

Deep learning can be used to determine canopy cover in a densely forested area. Li et al. (2022) use a deep learning approach to distinguish background vegetation points from over-story canopy points, to produce canopy maps from forests' 3D imagery.The results show that the deep learning approach outperforms traditional canopy mapping methods, therefore it is an accurate and robust method for creating canopy maps under different illuminations and terrain conditions.

Regression tasks can be performed using deep neural networks, Babaeian et al. (2021) used several machine learning methods and compared them to neural networks with two or three depth layers; the input data were multi-spectral VIs, and texture measurements from the soil and the expected output was soil moisture content; the results indicate an error below 5% and a high correlation value between the machine learning models and the predicted output.

4.9 Other algorithms

Other machine learning algorithms have been sparsely applied in different tasks. For example, gradient boosting machines (GBM) have been used to estimate soil moisture content in vegetated areas. Babaeian et al. (2021) tested several ML algorithms to predict soil moisture content including GBM. The results yielded that Neural Networks outperformed the other algorithms based on prediction error and the correlation factor. In the study presented by Villacrés and Cheein (2022), boosting gradient machines were used to reconstruct vegetation indices. Another task accomplished by GBM is the prediction of leaf nitrogen content based on hyperspectral indices, this is done by Raj et al. (2021), where the model achieved a correlation factor of 0.63, in areas with water-stressed vegetation; however, the model didn't achieve the same results in well-irrigated areas.

A more optimized version of gradient boosting is Extreme Gradient Boosting machine (XGB), this approach was used by Yu et al. (2021), to determine the forest structure and it was compared to random forest and support vector machines algorithms, in this studyit was determined that XGB was the best algorithm for this task achieving an F1 score of 0.91.

For classification purposes, Yang and Kan (2020) studied the use of Extreme learning machine (ELM) which is based on neural networks; and a Linear Bayes Normal Classifier (LBNC); the authors compared both algorithms with KNN; in this study ELM and LBNC achieved an accuracy of 97.55% and 96.53% respectively, both outperforming KNN in tree classification task.

The generation of digital terrain models was explored with the aid of machine learning (Arevalo-Ramirez et al., 2022), using conditional random field (CRF) to extract ground points; this approach generated smoother terrain models than other approaches not based on machine learning methods.

5 Discussion

There is a clear relationship between the discussed vegetative or forest issues, the sensors, and the machine learning algorithms selected to accomplish the research objectives. For tasks such as tree recognition and classification, deep learning and other classification algorithms prevail, and the selected sensors for this task are mainly imaging systems, RGB, or multi-spectral. Other tasks corresponding to determining and predicting phenotype features of forests such as chlorophyll, water, and moisture content often use regression algorithms, where input data are the VIs gathered from RGB, multi-spectral, and hyperspectral cameras. In the case of physical modeling of forests and determining its parameters, sensors such as LiDAR or terrestrial laser scanning systems are more suitable, due to their capability of creating 3d models from point clouds. Figure 15 illustrates the relationship between the vegetative issues, the sensors, and the data processing algorithms.

In general, all the reviewed works follow a somewhat similar workflow described by Müllerová et al. (2021): a problem in forestry health assessment is identified (chlorophyll prediction, water content estimation, biomass estimation, forest structure parametersestimation, tree classification, crown fuel estimation). Then the suitable sensors are selected depending on the needs of the problem, for example, if the problem is related to the geometric features of forests, a LiDAR sensor could fulfill the requirements. RGB, multi-spectral, and hyper-spectral cameras are more suited when spectral information is required and VIs are needed for example in chlorophyll estimation. The specific spectral response can also be used as an indicator of a specific tree speciesthus VIs are ideal to perform tree segmentation. Once the sensors are chosen, the data acquisition process is conducted. One of the most difficult parts of assessing forest health is the information processing phase. There is no clear pathway that leads to a correct decision when deciding which algorithm is the best to process the information according to the needs; as shown in the previous section, machine learning algorithms are a powerful alternative to process data and reach meaningful results.

5.1 Sensors used in remote sensing for forestry health assessment

Forestry health assessment aided by machine learning and remote sensing platforms is a promising trend in recent years. With the evolution of technology and machine learning techniques, better results in predictions of factors that affect forestry health have been accomplished. It is now possible to determine features from hyperspectral and multi-spectral imaging technologies, the use of UAVs helps the survey of great areas in short time, contrasted with a visual inspection from experts.

The use of LiDAR technology allows precise 3D reconstruction of environments in the range of centimeters (Hologa et al., 2021), allowing a complete geometrical characterization of forests, and the retrieval of tree and forest structure parameters. Efforts of mapping are important for forestry health assessment and to test algorithms;

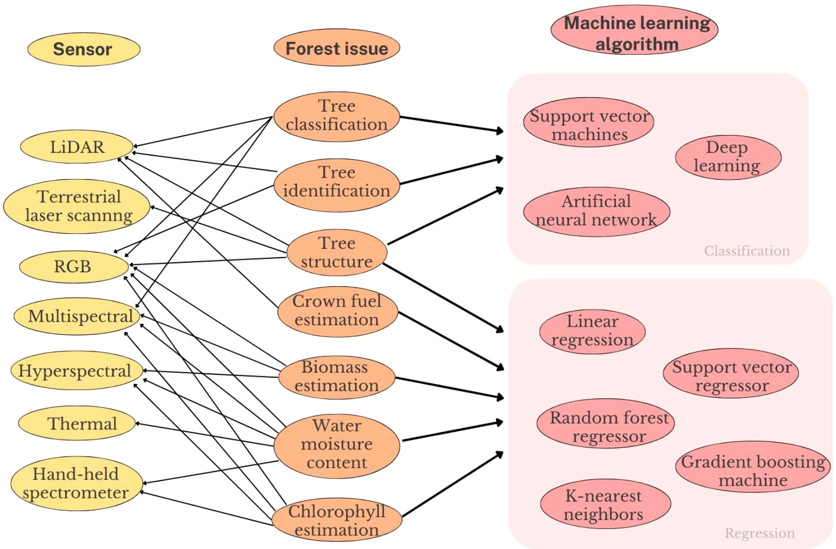


FIGURE 15
Relation between the vegetative or forest issue studied, the sensors and the machine learning algorithm chosen to do the investigation.

(Webster et al., 2018) performed thermal characterization of forest canopies in a large survey area, the study also made a coincident RGB mapping of the area, facilitating the access to public data to the scientific community.

The use of multi-spectral and hyper-spectral cameras to detect leaf reflectance and to compute different VIs has allowed an improvement in prediction techniques with the aid of machine learning algorithms. However, the information that can be gathered from spectral imaging methods is vast, and most of it will not have any correlation with the desired measurement, thus it is a current challenge to discover which bands and VIs are suitable for the different tasks in forestry health assessment. One way of reducing the dimensionality of input data for machine learning algorithms is the use of statistical methods to determine which information is more valuable and will provide better insight into the process, a common practice to reduce the dimensionality is to perform principal component analysis (PCA). Shovon et al. (2022) performed PCA in multi-spectral images, then a new VI with the four principal components, which was useful for identifying trees from snags. In the work presented by Kopacková-Strnadová et al. (2021), PCA was performed to reduce four spectral bands to three (three principal components), and with the selected bands, a VI was computed to predict photosynthetic pigments (i.e Chlorophyll). A similar process was performed by Barber et al. (2021), where the authors reduced the number of bands to predict fuel moisture in grasslands, again Ahmed et al. (2021b), reduced the number of multi-spectral bands to three principal components that represented the 86% variability of the images to generate VIs for tree identification. There is a greater issue when using hyperspectral imaging cameras since they can provide up to hundreds of bands; Yang and Kan (2020) retrieved 114 bands from a hyper-spectral camera, using a reduction process 14 bands were selected as principal feature bands, greatly reducing the dimension of the data.

5.2 Machine learning in forestry applications

The current trend in remote sensing for forestry health assessment is to use machine learning methods to process the information and find the desired correlations. These novel techniques currently outperform other methods that do not involve a training process, for example in the tree classification task Shovon et al. (2022) presented a thresholding algorithm to perform tree classification task, and even though the results were considered satisfactory, they are greatly outperformed by deep learning methods using convolutional layers. The accuracy is near a 90% (Onishi and Ise, 2021) on the training dataset with seven different tree species, whereas (Shovon et al., 2022) reported an accuracy of 80%.

The studies in classification tasks highlight that the use of deep learning techniques greatly outperforms other classification techniques (Onishi and Ise, 2021; Hell et al., 2022), and other studies present the advantage that the data does not need pre-processing (Liu et al., 2021). Hao Y. et al. (2022) performed individual tree detection without using machine learning models, and even though the proposed method improves the detection accuracy, reaching 90% in some scenarios; it is outperformed by the deep learning algorithm conducted proposed by Hao Z. et al. (2022).

The information needed as input data for deep learning and machine learning techniques is not clear either; in some cases, data extracted from UAV flights in a particular season of the year is insufficient for regression and classification purposes; thus recent articles investigate the use of multi-temporal data, for example, the results presented by Kopacková-Strnadová et al. (2021) suggest that temporal data is needed for predicting photosynthetic pigments in trees, given the fact that VIs from leaves of a certain age yielded the stronger correlated models. Other studies (Imangholiloo et al., 2020), explored the option of using data from different seasons for characterizing seedlings. Feng et al. (2020) used multi-temporal data to train DL networks, improving the accuracy of the model by more than 20% compared to the model using mono-temporal information.

For regression purposes, there is no clear tendency in the techniques that can be used to retrieve the desired data and make the predictions with the least amount of error. Most of the studies that rely on a prediction value, train different machine learning algorithms and assess the performance of each one using quantitative metrics. The performance of the algorithms varies case by case.

5.2.1 Publicly available data

One of the biggest drawbacks of using machine learning is the lack of curated available data to train the algorithms. In most forestry health assessment applications, not only the data acquisition process is necessary, but also generating the ground truthis needed. Generally, the ground truth is acquired with the help of expert knowledge and *in situ* measurements, which is an expensive and time-consuming process; thus studies to create large datasets fulfill a vital role for the scientific community. Weinstein et al. (2021) created a dataset containing LiDAR, RGB, and hyper-spectral information, with manual delineation of individual tree crowns. This dataset can be used to train machine-learning algorithms for tree detection and classification. Other studies compared how the reduction of samples affects the performance of deep learning models. Hao Z. et al. (2022) showed that by randomizing the training dataset and creating more dissimilar samples it is possible to reduce the number of training images without affecting the performance of the model. Research about the retrieval of pigments, particularly chlorophyll, water, and moisture content, is conducted through spectral information at the leaf or canopy level. Several datasets containing samples of multiple leaves and their reflectance are of great help when developing machine learning models for regression purposes, using as input some form of spectral data. Among the most used datasets for these purposes are the following: ANGERS (Jacquemound et al., 2003), which contains the spectral reflectance of 276 live, fresh leaves of 39 species of trees located in Angers, France; alongside chemical and physical measurements such as chlorophyll content and water content. Another dataset of similar characteristics is LOPEX dataset (Hosgood et al., 1993), which presents reflectance data of 330 leaf samples from 45 different tree species, this dataset also presents biochemical properties for the dataset. Both datasets and other similar ones can be found online (https://ecosis.org/). One important model for remote sensing applied to forestry applications is the PROSPECT model (Feret et al., 2008), which recreates spectral reflectance and transmittance at the canopy level, and could be of great use when predicting biochemical properties of leaves including pigment content (Feret et al., 2008). Information about publicly available datasets, including ANGERS, LOPEX and the one presented by Weinstein et al. (2021) is summarized in Table 3

Datasets for forestry applications using deep learning are scarce and, in the reviewed works, every group of researchers created its own databases with annotations, for their intended objectives. However public information is available and it has been compiled at Diez et al. (2021).

5.2.2 Big data approaches

Another future perspective for the assessment of forest health is the use of big-data approaches; under this new perspective, it is possible to use in conjunction with information retrieved from various sources including satellite platforms, airborne and terrestrial vehicles, and *in-situ* measurements to model the ever-changing dynamic of forests. One approach is to use the geological information-modeling system (GIMS), as presented by Varotsos and Krapivin (2017), who used GIMS to perform simulations evolution of the climate-nature-society system.

5.3 Future perspectives for machine learning and remote sensing in forestry health assessment

As shown in this current work, remote sensing aided by machine learning algorithms for forestry health applications is an active research field. As the methods of processing information advance and become more sophisticated, there is the possibility of highly improved forest management practices and contributing to sustainable forest management. Various studies (Liu et al., 2021; Onishi and Ise, 2021; Hell et al., 2022; Shovon et al., 2022), reported improved results in the metrics for tree recognition and tree classification, demonstrating the capabilities of machine learning to generate more precise models.

Another area that will continue to benefit from the improvement of models is the area of wildfire prevention (Jain et al., 2020). Correctly predicting fuel moisture content and biomass is of great help for predicting areas prone to wildfires. As seen in the reviewed works (Cężkowski et al., 2020; Raddi et al., 2021; Wells et al., 2021; Yao et al., 2021; Narmilan et al., 2022; Nasiri et al., 2022; Torre-Tojal et al., 2022), ˙ the use of machine learning algorithms have helped researchers predict biomass of fine fuels and moisture content at leaf and canopy level; thus helping identify dangerous areas for wildfire prevention. Machine learning models, alongside remote surveillance, carried out by UAV or satellite platforms will be of great importance for the prevention of disasters and the correct decision-making in disaster areas Jain et al. (2020).

6 Conclusions

The current state of the art suggests that for regression purposes (i.e estimating tree features, chlorophyll content, water leaf content, and soil moisture content among others); machine learning techniques are suitable. Choosing the imaging systems or sensors depends on the appropriate input data for the model it could be in the form of multi-spectral indices or metrics derived from LiDAR point clouds. However, there is no consensus on which regression technique achieves better performance.

DL techniques are a common trend for tree identification and classification tasks; these methods outperform other classification algorithms such as SVM and random forests, but they present the withdrawal of not enough data for training and validation purposes.

Most recent research is using multi-temporal information to improve the classification of trees from aerial images since the growing stage of trees affects their physical and chemical features.

The characterization of forests and their structure is a complex task due to the nature of the terrain, mixed and dense vegetation, constant evolution due to natural causes (different growth stages of the trees), and external causes (droughts, wildfires, climate change); therefore similar methodologies might not be suitable depending on the ecosystem.

The reviewed articles suggest that assessing forest features through remote sensing and machine learning techniques is a viable trend; since many ML techniques are being used for predicting forest health indices. Most recent works started exploring the use of Deep Learning Models, particularly convolutional neural networks to perform tree classification and recognition; these algorithms show great promise in reducing time for forest inventory and management, however; generating data for the training process, and creating models for general purposes are still some barriers in the use of deep learning techniques.

TABLE 3 Publicly available datasets for forestry health assessment.

Dataset	Content	Information	Case Application
ANGERS	Information from 276 leaves of different species	Visible and infrared spectra. Physical measurements. Biochemical analysis (Pigment content)	Development of model PROSPECT5 for reconstructing leaf reflectance (Feret et al., 2008). Testing machine learning algorithms for pigment estimation (Koirala et al., 2020; Shi et al., 2022).
LOPEX	Information from 330 samples of different species	Visible and infrared spectral. Physical Measurements. Biochemical Analysis (Pigment content).	Development of model PROSPECT5 (Feret et al., 2008). Training machine learning algorithms for pigment estimation Koirala et al. (2020)
Dataset presented by Weinstein et al., 2021).	Multiple sensor data and individual crown delineation.	RGB images. Hyper-spectral images. LiDAR point cloud. Individual image-annotated crowns. Individual field annotated crowns.	Development of individual crown detection algorithms from RGB and hyper-spectral images, and LiDAR point clouds Weinstein et al., 2021).

Author contributions

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

Funding

This work was funded by ANID SCIA ANILLO ACT210052, ANID FB008 21231129, ANID basal center AC3E, and the Direction of Postgrade Program of Federico Santamaria, Technical University.

Acknowledgments

The authors thank ANID SCIA ANILLO ACT210052, ANID FB0008 and DPP Universidad Técnica Federico Santamaría for their support in this work.

Conflict of interest

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

Publisher's note

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

References

Abdollahnejad, A., and Panagiotidis, D. (2020). Tree species classification and health status assessment for a mixed broadleaf-conifer forest with UAS multispectral imaging. *Remote Sens.* 12, 3722. doi: 10.3390/rs12223722

Aboutalebi, M., Torres-Rua, A. F., McKee, M., Kustas, W. P., Nieto, H., Alsina, M. M., et al. (2019). Incorporation of unmanned aerial vehicle (UAV) point cloud products into remote sensing evapotranspiration models. *Remote Sens.* 12, 50. doi: 10.3390/rs12010050

Adão, T., Hruška, J., Pádua, L., Bessa, J., Peres, E., Morais, R., et al. (2017). Hyperspectral imaging: a review on UAV-based sensors, data processing and applications for agriculture and forestry. *Remote Sens.* 9, 1110. doi: 10.3390/rs9111110

Ahmad, U., Alvino, A., and Marino, S. (2021). A review of crop water stress assessment using remote sensing. *Remote Sens.* 13, 4155. doi: 10.3390/rs13204155

Ahmed, S., Nicholson, C. E., Muto, P., Perry, J. J., and Dean, J. R. (2021a). Applied aerial spectroscopy: a case study on remote sensing of an ancient and semi-natural woodland. *PloS One* 16, e0260056. doi: 10.1371/journal.pone.0260056

Ahmed, S., Nicholson, C. E., Muto, P., Perry, J. J., and Dean, J. R. (2021b). The use of an unmanned aerial vehicle for tree phenotyping studies. *Separations* 8, 160. doi: 10.3390/separations8090160

Akkoyun, F. (2022). *Inexpensive multispectral imaging device* (Instrumentation Science & Technology), 1–17.

Arevalo-Ramirez, T., Guevara, J., Rivera, R. G., Villacres, J., Menendez, O., Fuentes, A., et al. (2022). Assessment of multispectral vegetation features for digital terrain modeling in forested regions. *IEEE Trans. Geosci. Remote Sens.* 60, 1–9. doi: 10.1109/tgrs.2021.3109601

Arkin, J., Coops, N. C., Daniels, L. D., and Plowright, A. (2021). Estimation of vertical fuel layers in tree crowns using high density lidar data. *Remote Sens.* 13, 4598. doi: 10.3390/rs13224598

Ashraf, M. A., Maah, M. J., and Yusoff, I. (2011). "Introduction to remote sensing of biomass," in *Biomass and remote sensing of biomass.* Ed. I. Atazadeh (Rijeka: IntechOpen), 8.

Babaeian, E., Paheding, S., Siddique, N., Devabhaktuni, V. K., and Tuller, M. (2021). Estimation of root zone soil moisture from ground and remotely sensed soil information with multisensor data fusion and automated machine learning. *Remote Sens. Environ.* 260, 112434. doi: 10.1016/j.rse.2021

Barber, N., Alvarado, E., Kane, V. R., Mell, W. E., and Moskal, L. M. (2021). Estimating fuel moisture in grasslands using uav-mounted infrared and visible light sensors. *Sensors* 21, 6350. doi: 10.3390/s21196350

Barmpoutis, P., Papaioannou, P., Dimitropoulos, K., and Grammalidis, N. (2020). A review on early forest fire detection systems using optical remote sensing. *Sensors* 20, 6442. doi: 10.3390/s20226442

Becker, S. J., Daughtry, C. S., and Russ, A. L. (2018). Robust forest cover indices for multispectral images. *Photogramm. Eng. Remote Sens.* 84, 505–512. doi: 10.14358/PERS.84.8.505

Blanco-Sacristán, J., Panigada, C., Gentili, R., Tagliabue, G., Garzonio, R., Martín, M. P., et al. (2021). UAV RGB, thermal infrared and multispectral imagery used to investigate the control of terrain on the spatial distribution of dryland biocrust. *Earth Surf. Processes Landforms* 46, 2466–2484. doi: 10.1002/esp.5189

Bohn, F. J., and Huth, A. (2017). The importance of forest structure to biodiversity–productivity relationships. *R. Soc. Open Sci.* 4, 160521. doi: 10.1098/rsos.160521

Breiman, L. (2001). Random forests. *Mach. Learn.* 45, 5–32. doi: 10.1023/a:1010933404324

Cabrera-Ariza, A. M., Lara-Gómez, M. A., Santelices-Moya, R. E., de Larriva, J.-E. M., and Mesas-Carrascosa, F.-J. (2022). Individualization of pinus radiata canopy from 3d UAV dense point clouds using color vegetation indices. *Sensors* 22, 1331. doi: 10.3390/s22041331

Cai, S., Zhang, W., Jin, S., Shao, J., Li, L., Yu, S., et al. (2021). Improving the estimation of canopy cover from UAV-LiDAR data using a pit-free CHM-based method. *Int. J. Digital Earth* 14, 1477–1492. doi: 10.1080/17538947.2021.1921862

Chamola, V., Kotesh, P., Agarwal, A., Naren,, Gupta, N., and Guizani, M. (2021). A comprehensive review of unmanned aerial vehicle attacks and neutralization techniques. *Ad Hoc Networks* 111, 102324. doi: 10.1016/j.adhoc.2020.102324

Cheng, M., Jiao, X., Liu, Y., Shao, M., Yu, X., Bai, Y., et al. (2022). Estimation of soil moisture content under high maize canopy coverage from UAV multimodal data and machine learning. *Agric. Water Manage.* 264, 107530. doi: 10.1016/j.agwat.2022.107530

Cężkowski, W., Szporak-Wasilewska, S., Kleniewska, M., Jó´zwiak, J., Gnatowski, T., D ˛abrowski, P., et al. (2020). Remotely sensed land surface temperature-based water stress index for wetland habitats. *Remote Sens.* 12, 631. doi: 10.3390/rs12040631

Cook, B. I., Smerdon, J. E., Seager, R., and Coats, S. (2014). Global warming and 21st century drying. *Climate Dynamics* 43, 2607–2627. doi: 10.1007/s00382-014-2075-y

Corte, A. P. D., Souza, D. V., Rex, F. E., Sanquetta, C. R., Mohan, M., Silva, C. A., et al. (2020). Forest inventory with high-density UAV-lidar: machine learning approaches for predicting individual tree attributes. *Comput. Electron. Agric.* 179, 105815. doi: 10.1016/j.compag.2020.105815

Dainelli, R., Toscano, P., Di Gennaro, S. F., and Matese, A. (2021). Recent advances in unmanned aerial vehicles forest remote sensing–a systematic review. part ii: research applications. *Forests* 12, 397. doi: 10.3390/f12040397

de Almeida, D. R. A., Broadbent, E. N., Ferreira, M. P., Meli, P., Zambrano, A. M. A., Gorgens, E. B., et al. (2021). Monitoring restored tropical forest diversity and structure through UAV-borne hyperspectral and lidar fusion. *Remote Sens. Environ.* 264, 112582. doi: 10.1016/j.rse.2021.112582

Diez, Y., Kentsch, S., Fukuda, M., Caceres, M. L. L., Moritake, K., and Cabezas, M. (2021). Deep learning in forestry using UAV-acquired RGB data: a practical review. *Remote Sens.* 13, 2837. doi: 10.3390/rs13142837

Eugenio, F. C., Schons, C. T., Mallmann, C. L., Schuh, M. S., Fernandes, P., and Badin, T. L. (2020). Remotely piloted aircraft systems and forests: a global state of the art and future challenges. *Can. J. For. Res.* 50, 705–716. doi: 10.1139/cjfr-2019-0375

Fakhri, S. A., Sayadi, S., Naghavi, H., and Latifi, H. (2022). A novel vegetation index-based workflow for semi-arid, sparse woody cover mapping. *J. Arid Environ.* 201, 104748. doi: 10.1016/j.jaridenv.2022.104748

Fan, G., Nan, L., Chen, F., Dong, Y., Wang, Z., Li, H., et al. (2020). A new quantitative approach to tree attributes estimation based on LiDAR point clouds. *Remote Sens.* 12, 1779. doi: 10.3390/rs12111779

Feng, Q., Yang, J., Liu, Y., Ou, C., Zhu, D., Niu, B., et al. (2020). Multi-temporal unmanned aerial vehicle remote sensing for vegetable mapping using an attention-based recurrent convolutional neural network. *Remote Sens.* 12, 1668. doi: 10.3390/rs12101668

Feret, J.-B., François, C., Asner, G. P., Gitelson, A. A., Martin, R. E., Bidel, L. P., et al. (2008). PROSPECT 4 and 5: advances in the leaf optical properties model separating photosynthetic pigments. *Remote Sens. Environ.* 112, 3030–3043. doi: 10.1016/j.rse.2008.02.012

Flach, P. (2019). "Performance evaluation in machine learning: the good, the bad, the ugly, and the way forward," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9808–9814. doi: 10.1609/aaai.v33i01.33019808

Forbes, B., Reilly, S., Clark, M., Ferrell, R., Kelly, A., Krause, P., et al. (2022). Comparing remote sensing and field-based approaches to estimate ladder fuels and predict wildfire burn severity. *Front. Forests Global Change* 5. doi: 10.3389/ffgc.2022.818713

Fraser, B. T., and Congalton, R. G. (2021a). Estimating primary forest attributes and rare community characteristics using unmanned aerial systems (UAS): an enrichment of conventional forest inventories. *Remote Sens.* 13, 2971. doi: 10.3390/rs13152971

Fraser, B. T., and Congalton, R. G. (2021b). Monitoring fine-scale forest health using unmanned aerial systems (UAS) multispectral models. *Remote Sens.* 13, 4873. doi: 10.3390/rs13234873

Furukawa, F., Laneng, L. A., Ando, H., Yoshimura, N., Kaneko, M., and Morimoto, J. (2021). Comparison of rgb and multispectral unmanned aerial vehicle for monitoring vegetation coverage changes on a landslide area. *Drones* 5, 97. doi: 10.3390/drones5030097

Gade, R., and Moeslund, T. B. (2013). Thermal cameras and applications: a survey. *Mach. Vision Appl.* 25, 245–262. doi: 10.1007/s00138-013-0570-5

Gale, M. G., Cary, G. J., Van Dijk, A. I., and Yebra, M. (2021). Forest fire fuel through the lens of remote sensing: review of approaches, challenges and future directions in the remote sensing of biotic determinants of fire behaviour. *Remote Sens. Environ.* 255, 112282. doi: 10.1016/j.rse.2020.112282

Gallardo-Salazar, J. L., Carrillo-Aguilar, D. M., Pompa-García, M., and Aguirre-Salado, C. A. (2021). Multispectral indices and individual-tree level attributes explain forest productivity in a pine clonal orchard of northern mexico. *Geocarto Int.*, 1–13. doi: 10.1080/10106049.2021.1886341

Goodfellow, I. J., Bengio, Y., and Courville, A. (2016). *Deep learning* (Cambridge, MA, USA: MIT Press). Available at: http://www.deeplearningbook.org.

Guimarães, N., Pádua, L., Marques, P., Silva, N., Peres, E., and Sousa, J. J. (2020). Forestry remote sensing from unmanned aerial vehicles: a review focusing on the data, processing and potentialities. *Remote Sens.* 12, 1046. doi: 10.3390/rs12061046

Guo, X., Liu, Q., Sharma, R. P., Chen, Q., Ye, Q., Tang, S., et al. (2021). Tree recognition on the plantation using UAV images with ultrahigh spatial resolution in a complex environment. *Remote Sens.* 13, 4122. doi: 10.3390/rs13204122

Hao, Z., Post, C. J., Mikhailova, E. A., Lin, L., Liu, J., and Yu, K. (2022). How does sample labeling and distribution affect the accuracy and efficiency of a deep learning model for individual tree-crown detection and delineation. *Remote Sens.* 14, 1561. doi: 10.3390/rs14071561

Hao, Y., Widagdo, F. R. A., Liu, X., Liu, Y., Dong, L., and Li, F. (2022). A hierarchical region-merging algorithm for 3-d segmentation of individual trees using UAV-LiDAR point clouds. *IEEE Trans. Geosci. Remote Sens.* 60, 1–16. doi: 10.1109/tgrs.2021.3121419

Hell, M., Brandmeier, M., Briechle, S., and Krzystek, P. (2022). Classification of tree species and standing dead trees with lidar point clouds using two deep neural networks: pointcnn and 3dmfv-net. *PFG–Journal Photogramm. Remote Sens. Geoinform. Sci.* 90, 103–121. doi: 10.1007/s41064-022-00200-4

Hillman, S., Wallace, L., Lucieer, A., Reinke, K., Turner, D., and Jones, S. (2021). A comparison of terrestrial and uas sensors for measuring fuel hazard in a dry sclerophyll forest. *Int. J. Appl. Earth Observ. Geoinform.* 95, 102261. doi: 10.1016/j.jag.2020.102261

Hologa, R., Scheffczyk, K., Dreiser, C., and Gärtner, S. (2021). Tree species classification in a temperate mixed mountain forest landscape using random forest and multiple datasets. *Remote Sens.* 13, 4657. doi: 10.3390/rs13224657

Hosgood, B., Jacquemound, S., Andreeoli, G., Verdebout, J., Pedrini, A., and Schmuck, G. (1993). *Leaf optical properties experiment database (LOPEX93)* (Cambridge, UK: Tech. rep).

Huo, L., Persson, H. J., and Lindberg, E. (2021). Early detection of forest stress from european spruce bark beetle attack, and a new vegetation index: normalized distance red & swir (ndrs). *Remote Sens. Environ.* 255, 112240. doi: 10.1016/j.rse.2020.112240

Idrissi, M., Hussain, A., Barua, B., Osman, A., Abozariba, R., Aneiba, A., et al. (2022). Evaluating the forest ecosystem through a semi-autonomous quadruped robot and a hexacopter uav. *Sensors* 22, 5497. doi: 10.3390/s22155497

Ilniyaz, O., Kurban, A., and Du, Q. (2022). Leaf area index estimation of pergola-trained vineyards in arid regions based on uav rgb and multispectral data using machine learning methods. *Remote Sens.* 14, 415. doi: 10.3390/rs14020415

Imangholiloo, M., Saarinen, N., Holopainen, M., Yu, X., Hyyppä, J., and Vastaranta, M. (2020). Using leaf-off and leaf-on multispectral airborne laser scanning data to characterize seedling stands. *Remote Sens.* 12, 3328. doi: 10.3390/rs12203328

Jacquemound, S., Bidel, L., Francois, C., and Pavan, G. (2003). *ANGERS leaf optical properties database* (Cambridge, UK: Tech. rep).

Jain, P., Coogan, S. C., Subramanian, S. G., Crowley, M., Taylor, S., and Flannigan, M. D. (2020). A review of machine learning applications in wildfire science and management. *Environ. Rev.* 28, 478–505. doi: 10.1139/er-2020-0019

K.C., S., Ninsawat, S., and Som-ard, J. (2021). Integration of RGB-based vegetation index, crop surface model and object-based image analysis approach for sugarcane yield estimation using unmanned aerial vehicle. *Comput. Electron. Agric.* 180, 105903. doi: 10.1016/j.compag.2020.105903

Khairul, I., and Bhuiyan, A. (2017). *LIDAR sensor for autonomous vehicle. tech. rep., technical report* (Chemnitz, Germany: Technische Universität Chemnitz).

Koirala, B., Zahiri, Z., and Scheunders, P. (2020). A machine learning framework for estimating leaf biochemical parameters from its spectral reflectance and transmission measurements. *IEEE Trans. Geosci. Remote Sens.* 58, 7393–7405. doi: 10.1109/tgrs.2020.2982263

Kopacková-Strnadová, V., Koucká, L., Jelének, J., Lhotáková, Z., and Oulehle, F. (2021). Canopy top, height and photosynthetic pigment estimation using parrot sequoia multispectral imagery and the unmanned aerial vehicle (UAV). *Remote Sens.* 13, 705. doi: 10.3390/rs13040705

Lee, K.-S., Cohen, W. B., Kennedy, R. E., Maiersperger, T. K. , and Gower, S. T. (2004). Hyperspectral versus multispectral data for estimating leaf area index in four different biomes. *Remote Sens. Environ.* 91, 508–520. doi: 10.1016/j.rse.2004.04.010

Li, L., Mu, X., Chianucci, F., Qi, J., Jiang, J., Zhou, J., et al. (2022). Ultrahigh-resolution boreal forest canopy mapping: combining uav imagery and photogrammetric point clouds in a deep-learning-based approach. *Int. J. Appl. Earth Observ. Geoinform.* 107, 102686. doi: 10.1016/j.jag.2022.102686

Li, X., Sun, Z., Lu, S., and Omasa, K. (2021). A multi-angular invariant spectral index for the estimation of leaf water content across a wide range of plant species in different growth stages. *Remote Sens. Environ.* 253, 112230. doi: 10.1016/j.rse.2020.112230

Liang, X., Kankare, V., Hyyppä, J., Wang, Y., Kukko, A., Haggrén, H., et al. (2016). Terrestrial laser scanning in forest inventories. *ISPRS J. Photogramm. Remote Sens.* 115, 63–77. doi: 10.1016/j.isprsjprs.2016.01.006

Liao, L., Cao, L., Xie, Y., Luo, J., and Wang, G. (2022). Phenotypic traits extraction and genetic characteristics assessment of eucalyptus trials based on UAV-borne LiDAR and RGB images. *Remote Sens.* 14, 765. doi: 10.3390/rs14030765

Lin, C., Chen, S.-Y., Chen, C.-C., and Tai, C.-H. (2018). Detecting newly grown tree leaves from unmanned-aerial-vehicle images using hyperspectral target detection techniques. *ISPRS J. Photogramm. Remote Sens.* 142, 174–189. doi: 10.1016/j.isprsjprs.2018.05.022

Lin, F.-C., and Chuang, Y.-C. (2021). Interoperability study of data preprocessing for deep learning and high-resolution aerial photographs for forest and vegetation type identification. *Remote Sens.* 13, 4036. doi: 10.3390/rs13204036

Lin, Y., and Herold, M. (2016). Tree species classification based on explicit tree structure feature parameters derived from static terrestrial laser scanning data. *Agric. For. Meteorol.* 216, 105–114. doi: 10.1016/j.agrformet.2015.10.008

Lin, J., Li, S., Qin, H., Wang, H., Cui, N., Jiang, Q., et al. (2022). *Overview of 3d human pose estimation* (Nevada, USA: CMES-COMPUTER MODELING IN ENGINEERING & SCIENCES).

Lin, Y.-C., Liu, J., Fei, S., and Habib, A. (2021). Leaf-off and leaf-on uav lidar surveys for single-tree inventory in forest plantations. *Drones* 5, 115. doi: 10.3390/drones5040115

Lin, L., Yu, K., Yao, X., Deng, Y., Hao, Z., Chen, Y., et al. (2021). UAV based estimation of forest leaf area index (LAI) through oblique photogrammetry. *Remote Sens.* 13, 803. doi: 10.3390/rs13040803

Linhares, J. M., Monteiro, J. A., Bailão, A., Cardeira, L., Kondo, T., Nakauchi, S., et al. (2020). How good are rgb cameras retrieving colors of natural scenes and paintings?–a study based on hyperspectral imaging. *Sensors* 20, 6242. doi: 10.3390/s20216242

Liu, M., Han, Z., Chen, Y., Liu, Z., and Han, Y. (2021). Tree species classification of lidar data based on 3d deep learning. *Measurement* 177, 109301. doi: 10.1016/j.measurement.2021.109301

Lou, P., Fu, B., He, H., Chen, J., Wu, T., Lin, X., et al. (2021). An effective method for canopy chlorophyll content estimation of marsh vegetation based on multiscale remote sensing data. *IEEE J. Select. Topics Appl. Earth Observ. Remote Sens.* 14, 5311–5325. doi: 10.1109/jstars.2021.3081565

Lu, J., Cheng, D., Geng, C., Zhang, Z., Xiang, Y., and Hu, T. (2021). Combining plant height, canopy coverage and vegetation index from UAV-based RGB images to estimate leaf nitrogen concentration of summer maize. *Biosyst. Eng.* 202, 42–54. doi: 10.1016/j.biosystemseng.2020.11.010

Malabad, A. M., Tatin-Froux, F., Gallinet, G., Colin, J.-M., Chalot, M., and Parelle, J. (2022). A combined approach utilizing UAV 3d imaging methods, *in-situ* measurements, and laboratory experiments to assess water evaporation and trace element uptake by tree species growing in a red gypsum landfill. *J. Hazard. Mater.* 425, 127977. doi: 10.1016/j.jhazmat.2021.127977

Mäyrä, J., Keski-Saari, S., Kivinen, S., Tanhuanpää, T., Hurskainen, P., Kullberg, P., et al. (2021). Tree species classification from airborne hyperspectral and LiDAR data using 3d convolutional neural networks. *Remote Sens. Environ.* 256, 112322. doi: 10.1016/j.rse.2021.112322

McClelland, M. P., van Aardt, J., and Hale, D. (2019). Manned aircraft versus small unmanned aerial system–forestry remote sensing comparison utilizing lidar and structure-from-motion for forest carbon modeling and disturbance detection. *J. Appl. Remote Sens.* 14, 1. doi: 10.1117/1.jrs.14.022202

Migliavacca, M., Musavi, T., Mahecha, M. D., Nelson, J. A., Knauer, J., Baldocchi, D. D., et al. (2021). The three major axes of terrestrial ecosystem function. *Nature* 598, 468–472. doi: 10.1038/s41586-021-03939-9

Miraki, M., and Sohrabi, H. (2021). Using canopy height model derived from UAV imagery as an auxiliary for spectral data to estimate the canopy cover of mixed broadleaf forests. *Environ. Monit. Assess.* 194. doi: 10.1007/s10661-021-09695-7

Miyoshi, G. T., Imai, N. N., Tommaselli, A. M. G., de Moraes, M. V. A., and Honkavaara, E. (2020). Evaluation of hyperspectral multitemporal information to improve tree species identification in the highly diverse atlantic forest. *Remote Sens.* 12, 244. doi: 10.3390/rs12020244

Morgan, G. R., Wang, C., and Morris, J. T. (2021). RGB Indices and canopy height modelling for mapping tidal marsh biomass from a small unmanned aerial system. *Remote Sens.* 13, 3406. doi: 10.3390/rs13173406

Müllerová, J., Gago, X., Bucas, M., Company, J., Estrany, J., Fortesa, J., et al. (2021). Characterizing vegetation complexity with unmanned aerial systems (UAS) – a framework and synthesis. *Ecol. Indic.* 131, 108156. doi: 10.1016/j.ecolind.2021.108156

Munnaf, M., Haesaert, G., Meirvenne, M. V., and Mouazen, A. (2020). "Site-specific seeding using multi-sensor and data fusion techniques: a review," in *Advances in agronomy* (Elsevier), 241–323. doi: 10.1016/bs.agron.2019.08.001

Narmilan, A., Gonzalez, F., Salgadoe, A. S. A., Kumarasiri, U. W. L. M., Weerasinghe, H. A. S., and Kulasekara, B. R. (2022). Predicting canopy chlorophyll content in sugarcane crops using machine learning algorithms and spectral vegetation indices derived from uav multispectral imagery. *Remote Sens.* 14, 1140. doi: 10.3390/rs14051140

Nasiri, V., Darvishsefat, A. A., Arefi, H., Griess, V. C., Sadeghi, S. M. M., and Borz, S. A. (2022). Modeling forest canopy cover: a synergistic use of sentinel-2, aerial photogrammetry data, and machine learning. *Remote Sens.* 14, 1453. doi: 10.3390/rs14061453

Neuville, R., Bates, J. S., and Jonard, F. (2021). Estimating forest structure from UAV-mounted LiDAR point cloud using machine learning. *Remote Sens.* 13, 352. doi: 10.3390/rs13030352

Omasa, K., Oki, K., and Suhama, T. (2006). "Remote sensing from satellites and aircraft (American society of agricultural and biological engineers), chap. 5," in *Precision agriculture* (Maryland, USA: American Society of Agricultural Engineers), 231–244.

Onishi, M., and Ise, T. (2021). Explainable identification and mapping of trees using uav rgb image and deep learning. *Sci. Rep.* 11, 1–15. doi: 10.1038/s41598-020-79653-9

Osco, L. P., Junior, J. M., Ramos, A. P. M., de Castro Jorge, L. A., Fatholahi, S. N., de Andrade Silva, J., et al. (2021). A review on deep learning in UAV remote sensing. *Int. J. Appl. Earth Observ. Geoinform.* 102, 102456. doi: 10.1016/j.jag.2021.102456

Pajares, G. (2015). Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *Photogramm. Eng. Remote Sens.* 81, 281–330. doi: 10.14358/pers.81.4.281

Pérez-Cabello, F., Montorio, R., and Alves, D. B. (2021). Remote sensing techniques to assess post-fire vegetation recovery. *Curr. Opin. Environ. Sci. Health* 21, 100251. doi: 10.1016/j.coesh.2021.100251

Pulido, D., Salas, J., Rös, M., Puettmann, K., and Karaman, S. (2020). Assessment of tree detection methods in multispectral aerial images. *Remote Sens.* 12, 2379. doi: 10.3390/rs12152379

Puliti, S., Dash, J. P., Watt, M. S., Breidenbach, J., and Pearse, G. D. (2019). A comparison of UAV laser scanning, photogrammetry and airborne laser scanning for precision inventory of small-forest properties. *Forest.: Int. J. For. Res.* 93, 150–162. doi: 10.1093/forestry/cpz057

Qiao, L., Gao, D., Zhao, R., Tang, W., An, L., Li, M., et al. (2022). Improving estimation of LAI dynamic by fusion of morphological and vegetation indices based on UAV imagery. *Comput. Electron. Agric.* 192, 106603. doi: 10.1016/j.compag.2021.106603

Qiu, Z., Ma, F., Li, Z., Xu, X., and Du, C. (2022). Development of prediction models for estimating key rice growth variables using visible and nir images from unmanned aerial systems. *Remote Sens.* 14, 1384. doi: 10.3390/rs14061384

Raddi, S., Giannetti, F., Martini, S., Farinella, F., Chirici, G., Tani, A., et al. (2021). Monitoring drought response and chlorophyll content in quercus by consumer-grade, near-infrared (NIR) camera: a comparison with reflectance spectroscopy. *New Forests* 53, 241–265. doi: 10.1007/s11056-021-09848-z

Raj, R., Walker, J. P., Pingale, R., Banoth, B. N., and Jagarlapudi, A. (2021). Leaf nitrogen content estimation using top-of-canopy airborne hyperspectral data. *Int. J. Appl. Earth Observ. Geoinform.* 104, 102584. doi: 10.1016/j.jag.2021.102584

Ramirez, W. A., Mishra, G., Panda, B. K., Jung, H.-W., Lee, S.-H., Lee, I., et al. (2022). Multispectral camera system design for replacement of hyperspectral cameras for detection of aflatoxin b 1. *Comput. Electron. Agric.* 198, 107078. doi: 10.1016/j.compag.2022.107078

Reilly, S., Clark, M. L., Bentley, L. P., Matley, C., Piazza, E., and Oliveras Menor, I. (2021). The potential of multispectral imagery and 3d point clouds from unoccupied aerial systems (uas) for monitoring forest structure and the impacts of wildfire in mediterranean-climate forests. *Remote Sens.* 13, 3810. doi: 10.3390/rs13193810

Resop, J. P., Lehmann, L., and Hession, W. C. (2021). Quantifying the spatial variability of annual and seasonal changes in riverscape vegetation using drone laser scanning. *Drones* 5, 91. doi: 10.3390/drones5030091

Ribas Costa, V. A., Durand, M., Robson, T. M., Porcar-Castell, A., Korpela, I., and Atherton, J. (2022). Uncrewed aircraft system spherical photography for the vertical characterization of canopy structural traits. *New Phytol.* 234, 735–747. doi: 10.1111/nph.17998

Rodríguez, A. G. F., Flores-Garnica, J. G., Gonz´alez-Eguiarte, D. R., Gallegos-Rodríguez, A., Zarazúa-Villaseñor, P., and Mena-Munguía, S. (2021). Comparative analysis of spectral indices to locate and size levels of severity of forest fires. *Invest. Geográficas.* doi: 10.14350/rig.60396

Sangjan, W., and Sankaran, S. (2021). Phenotyping architecture traits of tree species using remote sensing techniques. *Trans. ASABE* 64, 1611–1624. doi: 10.13031/trans.14419

Sapes, G., Lapadat, C., Schweiger, A. K., Juzwik, J., Montgomery, R., Gholizadeh, H., et al. (2022). Canopy spectral reflectance detects oak wilt at the landscape scale using phylogenetic discrimination. *Remote Sens. Environ.* 273, 112961. doi: 10.1016/j.rse.2022.112961

Shi, S., Xu, L., Gong, W., Chen, B., Chen, B., Qu, F., et al. (2022). A convolution neural network for forest leaf chlorophyll and carotenoid estimation using hyperspectral reflectance. *Int. J. Appl. Earth Observ. Geoinform.* 108, 102719. doi: 10.1016/j.jag.2022.102719

Shin, P., Sankey, T., Moore, M., and Thode, A. (2018). Evaluating unmanned aerial vehicle images for estimating forest canopy fuels in a ponderosa pine stand. *Remote Sens.* 10, 1266. doi: 10.3390/rs10081266

Shovon, T. A., Sprott, A., Gagnon, D., and Vanderwel, M. C. (2022). Using imagery from unmanned aerial vehicles to investigate variation in snag frequency among forest stands. *For. Ecol. Manage.* 511, 120138. doi: 10.1016/j.foreco.2022.120138

Stitt, J. M., Hudak, A. T., Silva, C. A., Vierling, L. A., and Vierling, K. T. (2022). Evaluating the use of lidar to discern snag characteristics important for wildlife. *Remote Sens.* 14, 720. doi: 10.3390/rs14030720

Suwardhi, D., Fauzan, K. N., Harto, A. B., Soeksmantono, B., Virtriana, R., and Murtiyoso, A. (2022). 3d modeling of individual trees from lidar and photogrammetric point clouds by explicit parametric representations for green open space (gos) management. *ISPRS Int. J. Geo Inform.* 11, 174. doi: 10.3390/ijgi11030174

Talavera, L., Costas, S., and Ferreira, Ó. (2022). A new index to assess the state of dune vegetation derived from true colour images. *Ecol. Indic.* 137, 108770. doi: 10.1016/j.ecolind.2022.108770

Terryn, L., Calders, K., Bartholomeus, H., Bartolo, R. E., Brede, B., D'hont, B., et al. (2022). Quantifying tropical forest structure through terrestrial and UAV laser scanning fusion in australian rainforests. *Remote Sens. Environ.* 271, 112912. doi: 10.1016/j.rse.2022.112912

Tesfamichael, S. G., van Aardt, J., Roberts, W., and Ahmed, F. (2018). Retrieval of narrow-range LAI of at multiple lidar point densities: application on eucalyptus grandis plantation. *Int. J. Appl. Earth Observ. Geoinform.* 70, 93–104. doi: 10.1016/j.jag.2018.04.014

Torres, P., Rodes-Blanco, M., Viana-Soto, A., Nieto, H., and García, M. (2021). The role of remote sensing for the assessment and monitoring of forest health: a systematic evidence synthesis. *Forests* 12, 1134. doi: 10.3390/f12081134

Torre-Tojal, L., Bastarrika, A., Boyano, A., Lopez-Guede, J. M., and Graña, M. (2022). Above-ground biomass estimation from LiDAR data using random forest algorithms. *J. Comput. Sci.* 58, 101517. doi: 10.1016/j.jocs.2021.101517

Toth, C., and Jóźków, G. (2016). Remote sensing platforms and sensors: a survey. *ISPRS J. Photogramm. Remote Sens.* 115, 22–36. doi: 10.1016/j.isprsjprs.2015.10.004

Tran, T. V., Reef, R., and Zhu, X. (2022). A review of spectral indices for mangrove remote sensing. *Remote Sens.* 14, 4868. doi: 10.3390/rs14194868

Trencanová, B., Proença, V., and Bernardino, A. (2022). Development of semantic maps of vegetation cover from UAV images to support planning and management in fine-grained fire-prone landscapes. *Remote Sens.* 14, 1262. doi: 10.3390/rs14051262

Trumbore, S., Brando, P., and Hartmann, H. (2015). Forest health and global change. *Science* 349, 814–818. doi: 10.1126/science.aac6759

Tuominen, S., Balazs, A., Honkavaara, E., Pölönen, I., Saari, H., Hakala, T., et al. (2017). Hyperspectral UAV-imagery and photogrammetric canopy height model in estimating forest stand variables. *Silva Fennica* 51. doi: 10.14214/sf.7721

UNCCD (2022) *Drought in numbers*. Available at: https://www.unccd.int/ resources/publications/drought-numbers.

Varotsos, C. A., and Krapivin, V. F. (2017). A new big data approach based on geoecological information modeling system. *Big Earth Data* 1, 47–63. doi: 10.1080/20964471.2017.1397405

Varotsos, C. A., Krapivin, V. F., and Mkrtchyan, F. A. (2020). A new passive microwave tool for operational forest fires detection: a case study of siberia in 2019. *Remote Sens.* 12, 835. doi: 10.3390/rs12050835

Villacrés, J., and Cheein, F. A. A. (2022). Construction of 3d maps of vegetation indices retrieved from UAV multispectral imagery in forested areas. *Biosyst. Eng.* 213, 76–88. doi: 10.1016/j.biosystemseng.2021.11.025

Vizireanu, I., Calcan, A., Grigoras, G., and Raducanu, D. (2020). Detection of trees features from a forestry area using airborne LiDAR data. *INCAS Bull.* 13, 225–236. doi: 10.13111/2066-8201.2021.13.1.23

Wan, L., Zhou, W., He, Y., Wanger, T. C., and Cen, H. (2022). Combining transfer learning and hyperspectral reflectance analysis to assess leaf nitrogen concentration across different plant species datasets. *Remote Sens. Environ.* 269, 112826. doi: 10.1016/j.rse.2021.112826

Watt, M. S., Buddenbaum, H., Leonardo, E. M. C., Estarija, H. J. C., Bown, H. E., Gomez-Gallego, M., et al. (2020). Using hyperspectral plant traits linked to photosynthetic efficiency to assess n and p partition. *ISPRS J. Photogramm. Remote Sens.* 169, 406–420. doi: 10.1016/j.isprsjprs.2020.09.006

Webster, C., Westoby, M., Rutter, N., and Jonas, T. (2018). Three-dimensional thermal characterization of forest canopies using UAV photogrammetry. *Remote Sens. Environ.* 209, 835–847. doi: 10.1016/j.rse.2017.09.033

Weinstein, B. G., Graves, S. J., Marconi, S., Singh, A., Zare, A., Stewart, D., et al. (2021). A benchmark dataset for canopy crown detection and delineation in co-registered airborne RGB, LiDAR and hyperspectral imagery from the national ecological observation network. *PloS Comput. Biol.* 17, e1009180. doi: 10.1371/journal.pcbi.1009180

Wells, A. G., Munson, S. M., Sesnie, S. E., and Villarreal, M. L. (2021). Remotely sensed fine-fuel changes from wildfire and prescribed fire in a semi-arid grassland. *Fire* 4, 84. doi: 10.3390/fire4040084

Xu, R., Li, C., and Bernardes, S. (2021). Development and testing of a UAV-based multi-sensor system for plant phenotyping and precision agriculture. *Remote Sens.* 13, 3517. doi: 10.3390/rs13173517

Xu, Z., Li, W., Li, Y., Shen, X., and Ruan, H. (2019). Estimation of secondary forest parameters by integrating image and point cloud-based metrics acquired from unmanned aerial vehicle. *J. Appl. Remote Sens.* 14, 22204. doi: 10.1117/1.jrs.14.022204

Xu, C., Zeng, Y., Zheng, Z., Zhao, D., Liu, W., Ma, Z., et al. (2022). Assessing the impact of soil on species diversity estimation based on uav imaging spectroscopy in a natural alpine steppe. *Remote Sens.* 14, 671. doi: 10.3390/rs14030671

Xu, P., Zhou, T., Yi, C., Luo, H., Zhao, X., Fang, W., et al. (2018). Impacts of water stress on forest recovery and its interaction with canopy height. *Int. J. Environ. Res. Public Health* 15, 1257. doi: 10.3390/ijerph15061257

Yan, D., Li, J., Yao, X., and Luan, Z. (2022). Integrating UAV data for assessing the ecological response of spartina alterniflora towards inundation and salinity gradients in coastal wetland. *Sci. Total Environ.* 814, 152631. doi: 10.1016/j.scitotenv.2021.152631

Yang, R., and Kan, J. (2020). Classification of tree species at the leaf level based on hyperspectral imaging technology. *J. Appl. Spectrosc.* 87, 184–193. doi: 10.1007/s10812-020-00981-9

Yang, R., and Kan, J. (2022). Classification of tree species in different seasons and regions based on leaf hyperspectral images. *Remote Sens.* 14, 1524. doi: 10.3390/rs14061524

Yang, R., Liu, L., Liu, Q., Li, X., Yin, L., Hao, X., et al. (2022). Validation of leaf area index measurement system based on wireless sensor network. *Sci. Rep.* 12, 1–13. doi: 10.1038/s41598-022-08373-z

Yao, X., Chen, S., Ding, S., Zhang, M., Cui, Z., Linghu, S., et al. (2021). Temperature, moisture, hyperspectral vegetation indexes, and leaf traits regulated soil respiration in different crop planting fields. *J. Soil Sci. Plant Nutr.* 21, 3203–3220. doi: 10.1007/s42729-021-00600-2

Yu, J.-W., Yoon, Y.-W., Baek, W.-K., and Jung, H.-S. (2021). Forest vertical structure mapping using two-seasonal optic images and LiDAR DSM acquired from UAV platform through random forest, XGBoost, and support vector machine approaches. *Remote Sens.* 13, 4282. doi: 10.3390/rs13214282

Zhang, C., Xia, K., Feng, H., Yang, Y., and Du, X. (2020). Tree species classification using deep learning and RGB optical images obtained by an unmanned aerial vehicle. *J. Forest. Res.* 32, 1879–1888. doi: 10.1007/s11676-020-01245-0

Zhang, C., Xia, K., Feng, H., Yang, Y., and Du, X. (2021). Tree species classification using deep learning and rgb optical images obtained by an unmanned aerial vehicle. *J. Forest. Res.* 32, 1879–1888. doi: 10.1007/s11676-020-01245-0

Zhang, K., Yang, C., Li, X., Zhou, C., and Zhong, R. (2020). High-efficiency microsatellite-using super-resolution algorithm based on the multi-modality super-cmos sensor. *Sensors* 20, 4019. doi: 10.3390/s20144019

Zhang, Y., Yang, W., Sun, Y., Chang, C., Yu, J., and Zhang, W. (2021). Fusion of multispectral aerial imagery and vegetation indices for machine learning-based ground classification. *Remote Sens.* 13, 1411. doi: 10.3390/rs13081411

Zhao, Q., Yu, L., Du, Z., Peng, D., Hao, P., Zhang, Y., et al. (2022). An overview of the applications of earth observation satellite data: impacts and future trends. *Remote Sens.* 14, 1863. doi: 10.3390/rs14081863

Zhuo, W., Wu, N., Shi, R., and Wang, Z. (2022). UAV mapping of the chlorophyll content in a tidal flat wetland using a combination of spectral and frequency indices. *Remote Sens.* 14, 827. doi: 10.3390/rs14040827

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 07 July 2023
DOI 10.3389/fpls.2023.1142957

Deep reinforcement learning enables adaptive-image augmentation for automated optical inspection of plant rust

Shiyong Wang[1†], Asad Khan[2*], Ying Lin[1], Zhuo Jiang[3], Hao Tang[4*], Suliman Yousef Alomar[5], Muhammad Sanaullah[6] and Uzair Aslam Bhatti[4*†]

[1]School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, China, [2]Metaverse Research Institute, School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, China, [3]College of Food Science, South China Agricultural University, Guangzhou, China, [4]School of Information and Communication Engineering, Hainan University, Haikou, China, [5]Zoology Department, College of Science, King Saud University, Riyadh, Saudi Arabia, [6]Department of Computer Science, Bahauddin Zakariya University, Multan, Pakistan

OPEN ACCESS

EDITED BY
Yaqoob Majeed,
University of Agriculture, Faisalabad, Pakistan

REVIEWED BY
Sushma Jaiswal,
Guru Ghasidas Vishwavidyalaya, India
Muthusamy Ramakrishnan,
Nanjing Forestry University, China
Jianming Zhang,
Chinese Academy of Science, China

*CORRESPONDENCE
Asad Khan
asad@gzhu.edu.cn
Hao Tang
melineth@hainanu.edu.cn
Uzair Aslam Bhatti
uzairaslambhatti@hotmail.com

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

RECEIVED 12 January 2023
ACCEPTED 29 May 2023
PUBLISHED 07 July 2023

CITATION
Wang S, Khan A, Lin Y, Jiang Z, Tang H, Alomar SY, Sanaullah M and Bhatti UA (2023) Deep reinforcement learning enables adaptive-image augmentation for automated optical inspection of plant rust. *Front. Plant Sci.* 14:1142957. doi: 10.3389/fpls.2023.1142957

This study proposes an adaptive image augmentation scheme using deep reinforcement learning (DRL) to improve the performance of a deep learning-based automated optical inspection system. The study addresses the challenge of inconsistency in the performance of single image augmentation methods. It introduces a DRL algorithm, DQN, to select the most suitable augmentation method for each image. The proposed approach extracts geometric and pixel indicators to form states, and uses DeepLab-v3+ model to verify the augmented images and generate rewards. Image augmentation methods are treated as actions, and the DQN algorithm selects the best methods based on the images and segmentation model. The study demonstrates that the proposed framework outperforms any single image augmentation method and achieves better segmentation performance than other semantic segmentation models. The framework has practical implications for developing more accurate and robust automated optical inspection systems, critical for ensuring product quality in various industries. Future research can explore the generalizability and scalability of the proposed framework to other domains and applications. The code for this application is uploaded at https://github.com/lynnkobe/Adaptive-Image-Augmentation.git.

KEYWORDS

adaptive image augmentation, deep reinforcement learning, deep Q-learning, automated optical inspection, semantic segmentation

1 Introduction

Automated optical inspection (AOI) provides a flexible and efficient method of object monitoring. In agriculture, AOI can be used for early screening of leaf diseases to support timely intervention to prevent leaf rust. Leaf rust is a type of plant disease also known as red spot disease or sheep beard. There are 4,000 known species of leaf rust that attack a wide range of crops such as beans, tomatoes, and roses (Liu et al., 2022; Bhatti et al., 2023). Disease spots first appear as white and slightly raised spots on the lower cuticles of the lower (older) leaves of mature plants. Over time, the disease spots become covered in reddish-orange spore masses. Later, pustules form and turn yellow-green and eventually black. Severe infestations can cause foliage to chlorosis, deform, and eventually fall off (Jain et al., 2019; Bhatti et al., 2021; Lu et al., 2023; Wang et al., 2023; Yang et al., 2022; Zhang et al., 2022). The spread of this disease will seriously affect agricultural production and cause huge losses. Thus, detecting plant disease and rust is very important and effective for protecting plant growth and development, improving crop yield and quality, reducing pesticide use, and saving time and cost (Bhatti et al., 2022; Shoaib et al., 2023).

Artificial intelligence-enhanced AOI methods based on computer vision and deep learning are promising solutions for the adaptive identification of plant diseases (Liu and Wang, 2021). Algorithms that incorporate the two major computer vision tasks—classification and detection—have been widely used in plant disease detection. In terms of classification algorithms, Sethy et al. (2020) used convolutional neural networks (CNNs), ResNet50, to extract features, which were then fed to a support vector machine (SVM) for the disease classification, achieving an F1 score of 0.9838. Zhong and Zhao (2020) proposed three methods based on the DenseNet-121 deep convolutional network: regression, multi-label classification, and focal loss function to identify apple leaf diseases and improve the detection accuracy in unbalanced plant disease datasets. In terms of detection algorithms, Zhou et al. (2019) proposed a fast rice disease detection method based on the fusion of FCM-KM and Faster R-CNN to improve detection accuracy and reduce detection time. Sun et al. (2020) proposed a CNN-based multi-scale feature fusion instance detection method based on the improved SSD to detect corn leaf blight on complex backgrounds, with the highest average precision reaching 91.83%.

The classification and detection of plant diseases are only possible to judge whether the disease occurs in certain locations (Di and Li, 2022; Khan et al., 2022; Yan et al., 2022; Deng et al., 2023; Wang et al., 2023). Using computer vision segmentation algorithms, the size and shape of plant rust spots can be obtained (Wang et al., 2021; Ban et al., 2022; Shoaib et al., 2022; Zhang et al., 2022; Dang et al., 2023; Wang et al., 2023), and the severity of rust occurrence can be quantitatively evaluated. He et al. (2021) proposed an asymmetric shuffle convolutional neural network (ASNet) based on Mask R-CNN to segment three diseases, including apple rust, with an average segmentation accuracy of 94.7%. Lin et al. (2019) proposed a U-net-based CNN to segment powdery mildew from cucumber leaf images at the pixel level. Unfortunately, compared with the classification and detection of diseases, there is still little research on applying deep learning segmentation networks for rust identification.

In the study of rust detection, the size of the available data set is limited, and manual labeling requires a lot of time and effort. The traditional solution to image augmentation is to perform simple image processing, which has been verified to improve the performance of plant image segmentation. Lin et al. (2019) proposed improving the U-net segmentation network by using image augmentation technology to expand the training set to train the semantic segmentation model better. Zhang et al. (2022) proposed the DMCNN model, which obtained twice the data after image augmentation and achieved an average apple disease detection rate of more than 99.5%. The research proves that sample size and data quality are critical to improving detection accuracy. Unfortunately, whether there is redundancy in the data set obtained by image augmentation or whether the data quality is good or bad (Elmore and Lee, 2021; Dang et al., 2023; Xiong et al., 2023) is a question worth exploring. Blind pursuit of a sample size for inappropriate image augmentation may adversely affect the model.

Several image augmentation methods have been proposed, such as rotation and cropping. However, no single approach can always outperform others, and the image quality generated by these augmentation methods is uncertain. In other words, the bottleneck of current image augmentation methods is that it is difficult to define the optimal augmentation operation to achieve the most significant performance improvement for semantic segmentation. Currently, multiple augmentation methods are generally used together: all methods for the complete image set, one for a separated subset, or one for a randomly sampled subset. However, none of these assignment mechanisms can guarantee the best match between an image and an available augmentation method. To overcome this problem, deep reinforcement learning (DRL)-based image augmentation methods have been proposed (Yang et al., 2023). DRL is a machine learning technique that enables a software agent to optimize its decision-making policy by interacting with its environment (Zhou et al., 2021). Le et al. (2022) stated that DRL can automatically learn how to augment datasets effectively. Qin et al. (2020) developed a novel automatic learning-based image augmentation method for medical image segmentation, using DRL to model the augmentation task as a trial-and-error process.

However, image augmentation and image segmentation were previously trained in separate ways (Di and Li, 2022). The image segmentation results cannot provide feedback to the DRL-based image augmentation model. Therefore, we propose a DRL-enabled adaptive image augmentation framework based on the Deep Q-learning (DQN) algorithm and the semantic segmentation model, DeepLab-v3+, for apple rust detection. DQN learns the Q-value function with a deep neural network and uses the experience playback and the target network to improve the stability and learning effect (Xu et al., 2022). The main contributions of this study are as follows:

(1) A DRL-enabled adaptive image augmentation framework is proposed to adaptively select the best-matched image

augmentation methods according to the image features. This way, an effective augmented image set is constructed from the original image set.

(2) The DeepLab-v3+ model is applied. It is pre-trained by the original image set and retrained in conjunction with the augmentation image set. The model is retrained in a transfer-learning way, featuring fast fine-tuning. The retrained model outputs average performance over the test image set as an evaluation index for the augmented image. Furthermore, the evaluation index provided feedback to the DRL model as a reward.

(3) The superiority of the DRL-enabled adaptive image augmentation framework is verified by comparing it with other image augmentation methods and semantic segmentation models over a set of performance indexes.

(4) The main finding is that the DRL-enabled adaptive image augmentation framework can best match image augmentation methods with the image features and the underlying segmentation model.

This paper provides an end-to-end, robust, and effective method for segmenting rust spots at the pixel level, providing a valuable tool for farmers and botanists to assess the severity of rust.

2 Method

The DRL-enabled adaptive image augmentation framework is depicted in Figure 1. The DQN model acts as the Agent, and the image set is treated as the environment. The Agent and the Environment repeatedly interact through the signals: state s_t, action a_t, and reward r_t. The state s_t and the reward r_t are output by the environment to the Agent while the action a_t is determined by the Agent and executed in the environment. The interaction process consists of episodes, which in turn comprise multiple steps. The experience data are collected during the interaction process and used to train the Agent until the Agent can best match the augmentation methods and the images. In this specific scenario, the Agent can augment a given image appropriately so that the augmented image set can enable the segmentation model to output better performance.

The detailed interaction process is illustrated in Figure 2. A group of objects, e.g., images, states, and actions, are represented as a vector when the precedence relationship should be maintained; otherwise, the group of things is encapsulated with a set. In any round of interaction t, the geometric and pixel indicators are applied to extract the image features of the father image vector I_{t-1}, which are then used to construct the state vector s_t. After that, the action vector a_t is determined based on the state vector s_t and the Agent policy function $\pi_\theta(a_t|s_t)$. The actions in a_t represent image augmentation methods selected individually for each image in I_{t-1}. Therefore, a_t will produce a child image vector I_t after being executed. After that, the child image vector is combined with the pre-training image set I_0 to construct a retraining image set. Then, the retraining image set is used to retrain the pre-trained image segmentation model, DeepLab-v3+. Finally, the retrained model is tested on the test image set I_{test}, and the testing results are used to generate the reward r_t. At this moment, the data (s_t, a_t, r_t) can be collected.

In the next round, the I_t is used as the father image vector, and the above process is repeated so that the data $(s_{t+1}, a_{t+1}, r_{t+1})$ can be collected. In addition, the data (s_t, a_t, r_t, s_{t+1}) need storing in the experience replay buffer for training the Agent policy function $\pi_\theta(a_t|s_t)$. After the process is repeated T times, an episode is said to be completed. To begin the next episode, reset t to 1, and restore the pre-training image set I_0 as the father image vector. The number of episodes, L, is another hyperparameter like the number of steps T within an episode, which means a total of L by T steps should be executed.

The Agent policy function $\pi_\theta(a_t|s_t)$ evolves during the above interaction process. A number of S samples are extracted from the experience replay buffer and applied to update the parameter θ of $\pi_\theta(a_t|s_t)$. The hyperparameters, e.g., L, T, and S need adjusting and $\pi_\theta(a_t|s_t)$ need updating till the performance is satisfied.

2.1 Image set and image vector

The original image set is divided into two subsets. Twenty percent of the images are sampled randomly from the original image set, forming the test image set I_{test} that is used to test the DeepLab-v3+ model. The remaining 80% of images are collected by a subset denoted as I_0, which is called the pre-training image set. Let $I_0 = \{ I_{0,1}, I_{0,2}, \ldots, I_{0,m}\} = \{(x_1^0, y_1^0), (x_2^0, y_2^0), \ldots, (x_m^0, y_m^0)\}$, where x_i^0 and y_i^0 are the ith image and its corresponding label image, and m is the total number of samples in the image set. Through the image augmentation procedure, an image in $I_{t-1}(t = 1 \ldots T)$ is applied to an image augmentation method to produce an augmented image, and all the augmented images make up the augmented image set $I_t = \{ I_{t,1}, I_{t,2}, \ldots, I_{t,m}\} = \{ (x_1^t, y_1^t), (x_2^t, y_2^t), \ldots, (x_m^t, y_m^t)\}$.

During the DQN augmentation process, the image sets are represented as vectors. In an image vector, the images are queued in a line, each occupying a fixed and unique position. At the first step of an episode, i.e., $t = 1$, I_0 is used as the father image vector denoted as I_{t-1}. Then the images in I_{t-1} are augmented to produce the child image vector denoted as I_t. The image vectors are used instead of image sets because the corresponding relationship between I_{t-1} and I_t should be maintained. In other words, the first image in I_t is produced from the first image in I_{t-1} and so forth. It is noted that the images in I_{t-1} are applied to image augmentation methods independently.

The pre-training image set I_0 alone is used to pre-train the DeepLab-v3+ model. In contrast, I_0 is combined with the augmented image set I_t to retrain the pre-trained DeepLab-v3+ model to verify the effect of I_t. In other words, the I_0 and I_{test} are used to pre-train and test the semantic segmentation model DeepLab-v3+. The pre-trained DeepLab-v3+ model is retrained and tested by $I_0 U \quad I_t$ and I_{test} to see the influence of the augmented image set I_t on the pre-trained model.

In the next step, the newly produced image vector I_t instead of I_{t-1} is used as the father image vector to produce its child image

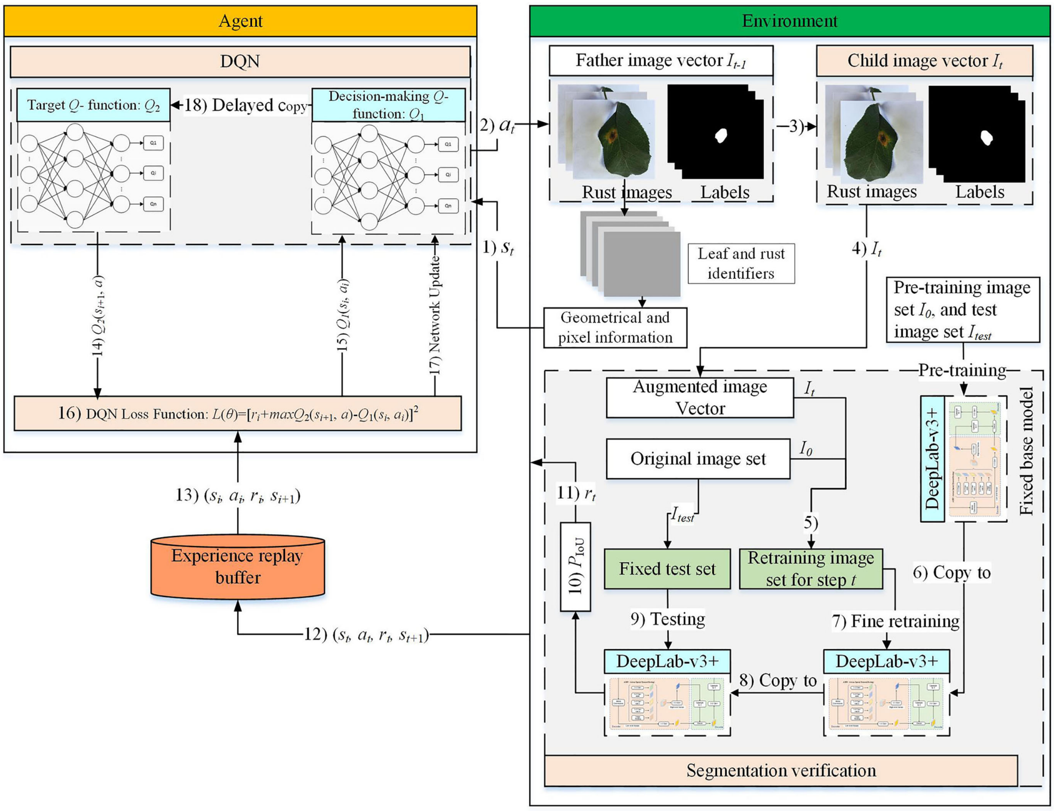


FIGURE 1
DRL-enabled adaptive image augmentation framework.

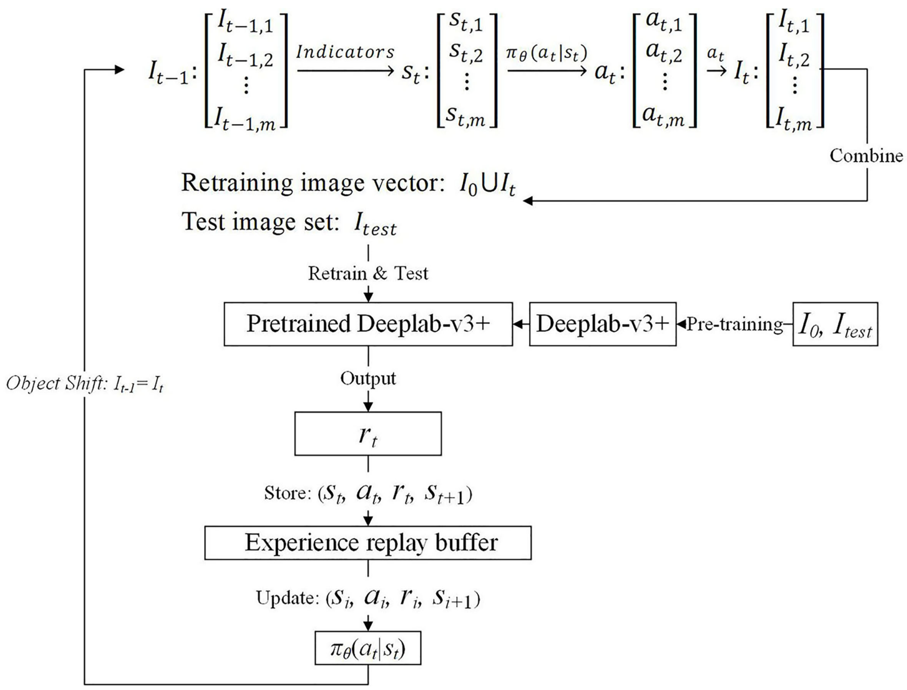

$I_{t-1}: \begin{bmatrix} I_{t-1,1} \\ I_{t-1,2} \\ \vdots \\ I_{t-1,m} \end{bmatrix} \xrightarrow{Indicators} s_t: \begin{bmatrix} s_{t,1} \\ s_{t,2} \\ \vdots \\ s_{t,m} \end{bmatrix} \xrightarrow{\pi_\theta(a_t|s_t)} a_t: \begin{bmatrix} a_{t,1} \\ a_{t,2} \\ \vdots \\ a_{t,m} \end{bmatrix} \xrightarrow{a_t} I_t: \begin{bmatrix} I_{t,1} \\ I_{t,2} \\ \vdots \\ I_{t,m} \end{bmatrix}$
Combine
Retraining image vector: $I_0 \cup I_t$
Test image set: I_{test}
Retrain & Test
Pretrained Deeplab-v3+
Deeplab-v3+
Pre-training
I_0, I_{test}
Object Shift: $I_{t-1}=I_t$
Output
r_t
Store: (s_t, a_t, r_t, s_{t+1})
Experience replay buffer
Update: (s_i, a_i, r_i, s_{i+1})
$\pi_\theta(a_t|s_t)$

FIGURE 2
DRL-enabled adaptive image augmentation process.

vector I_{t+1}. Then, I_{t+1} is united with I_0 to construct another retraining image set to test the augmentation effect of I_{t+1} based on the pre-trained DeepLab-v3+ model. To sum up, the newly produced child image vector is used as the father image vector in the next step until the episode ends. However, to begin a new episode, the pre-training image set I_0 is used as the father image vector again, and the image vectors produced in the last episode are discarded. It is noted that the pre-trained DeepLab-v3+ model is restored in every retraining process and is used as a base model to observe the effect of the augmentation methods on the augmented image sets.

2.2 MDP model for DRL

The DRL-based optimization features a Markov decision process (MDP) (Han et al., 2021). The Agent selects an action from the candidate's actions based on the current state of the environment. The execution of the action will introduce a state change to the environment which in turn generates a reward to the Agent. The Agent decides (i.e., selects an action) based on the current state only, not depending on the previous states. This design contributes to simplifying the Agent policy function but requires sophisticated state representation. The reward guides the evolution of the policy function. Therefore, maximizing cumulative compensation should correspond to the best selection policy of augmentation methods for any given image set. Although the single-step reward can be positive (a prize), negative (a penalty), or zero, the Agent should tolerate the short-term penalty while pursuing the maximum cumulative reward. The actions are candidate image augmentation methods that have been proven to be effective in certain circumstances. The best state-action match, however, is still unknown, leaving optimization space for DRL. Therefore, the state, action, and reward design will significantly influence DRL's optimization quality (Ladosz et al., 2022).

2.2.1 State

An amount of information is extracted from the image vector to describe the state of the environment. In this study, each image's geometrical information and pixel information comprise a state for a given image vector. At first, one segmentation model, called LeafIdentifier, is trained to separate a leaf from its background. Furthermore, the other segmentation model, called RustIdentifier, is trained to separate the rust from a leaf. The LeafIdentifier and the RustIdentifier models are developed based on the DeepLab-v3+ model but prepared with different datasets. The image set I_0 with the leaf label is used to train the LeafIdentifier model, while the image set I_0 with the rust label is used to train the RustIdentifier model.

After that, the centroid and area of the leaf and the rust can be calculated. In addition, the pixel values can be averaged according to the RGB color channels for the leaf and the rust, respectively. Therefore, a state element that describes the ith image is:

$$s_{t,i} = \left\{x_{l,i}, y_{l,i}, A_{l,i}, R_{l,i}, G_{l,i}, B_{l,i}, x_{r,i}, y_{r,i}, A_{r,i}, R_{r,i}, G_{r,i}, B_{r,i}\right\}$$

where, $x_{l,i}$ and $y_{l,i}$ are the centroid coordinates of a leaf, $A_{l,i}$ is the area of a leaf, and $R_{l,i}$, $G_{l,i}$, and $B_{l,i}$ are the average pixel values of a leaf, corresponding to the RGB color channels, respectively; $x_{r,i}$, $y_{r,i}$, $A_{r,i}$, $R_{r,i}$, $G_{r,i}$, and $B_{r,i}$ are the corresponding elements for the rusts on the leaf.

Therefore, the state vector has the same number of elements as the father image vector, and their elements have a one-to-one corresponding relationship.

2.2.2 Action

Eight kinds of image augmentation methods are selected as actions, as shown in Table 1. The *original image* operation does not change the image. The *vertical flip* operation makes an image flip vertically, while the *horizontal flip* operation makes an image flip horizontally. However, the *vertical and horizontal flip* operations apply the two operations together to a single image. The *clockwise rotation* operation causes an image to rotate 30° clockwise around the center point. The *affine transformation* is a type of geometric transformation that preserves collinearity and the ratios of distances between points on a line. The *crop* operation is to crop the original image and then resize it to the original size. When applying the noise-adding operation, random white Gaussian noise will be added to a given image. Each image augmentation method is assigned a unique number, i.e., 0, 1, 2,…7. In this study, $a_i(i = 0 \dots 7)$ is used to represent the eight candidates' actions, and $a_t(t = 1 \dots T)$ is used to indicate the action vector consisting of actions selected independently for each image in the decision step t. Therefore, the different elements of a_t possible correspond to the same a_i.

2.2.3 Reward

The reward is a numerical evaluation of an action selected by the Agent:

$$r_t = 100(d_t - d_{t-1}) \tag{1}$$

where, d_t refers to the Dice ratio, defined as follows:

$$d_t = \frac{2}{|I_{test}|} \sum_{(x_j, y_j) \in I_{test}} P_{IoU} \tag{2}$$

where, $|I_{test}|$ is the number of elements in the test image set I_{test}, and $P_{\mathrm{IoU}} \in [0, 1]$ represents the segmentation effect of the retrained DeepLab-v3+ model on an image of I_{test}:

$$P_{\mathrm{IoU}} = \frac{|\hat{y}_j \cap y_j|}{|\hat{y}_j \cup y_j|}, \ y_j \in (x_j, y_j) \in I_{test} \tag{3}$$

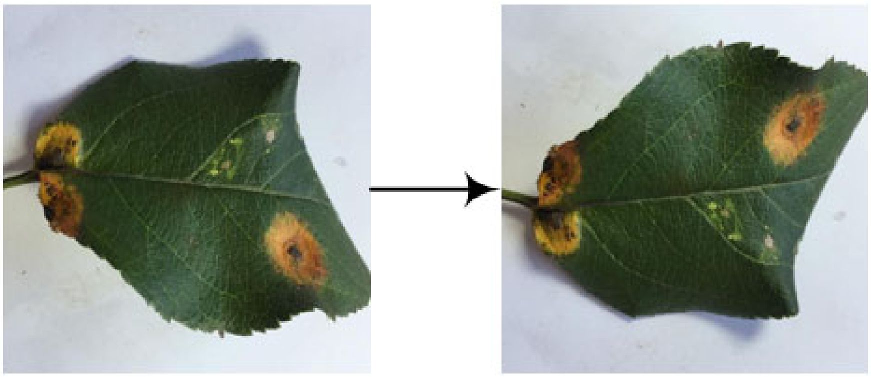
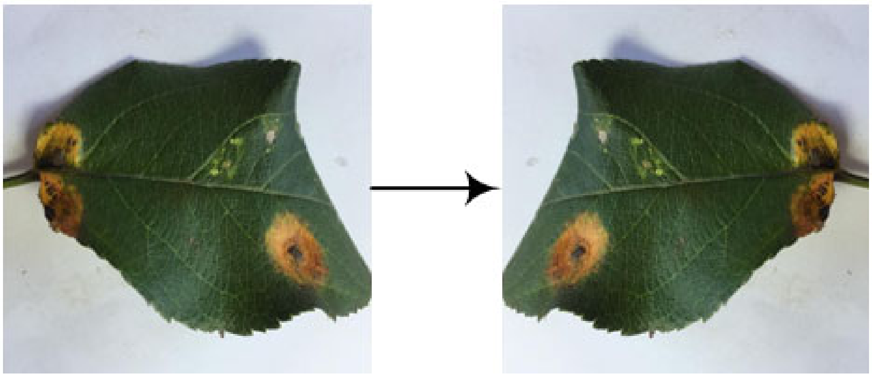
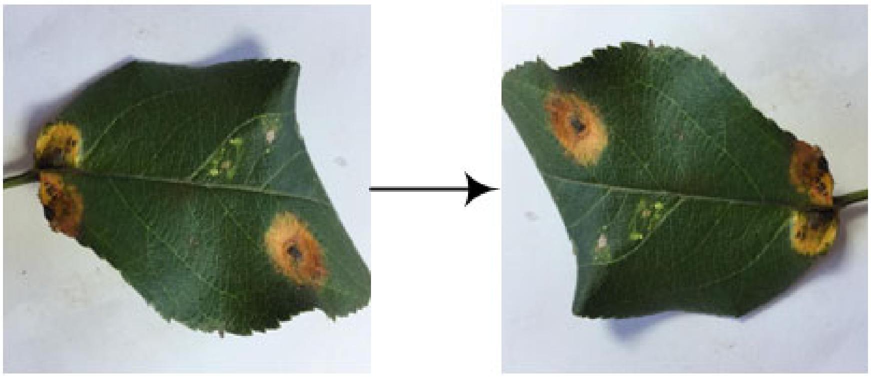
where, $\hat{y}_j$ is the predicted label image output by the retrained DeepLab-v3+ model, and y_j is the expected label image, both for the image x_j in the test image set I_{test}; $|\hat{y}_j \cap \quad y_j|$ and $|\hat{y}_j \cup \quad y_j|$ are the intersection and union area of the predicted and expected label images, respectively:

$$\hat{y}_j = f(x_j; \theta_{I_0 \cup I_t}), \ x_j \in (x_j, y_j) \in I_{test} \tag{4}$$

where f denotes the retrained DeepLab-v3+ model, and $\theta_{I_0 U \quad I_t}$ denotes the parameters updated by the retraining image set $I_0 U \quad I_t$.

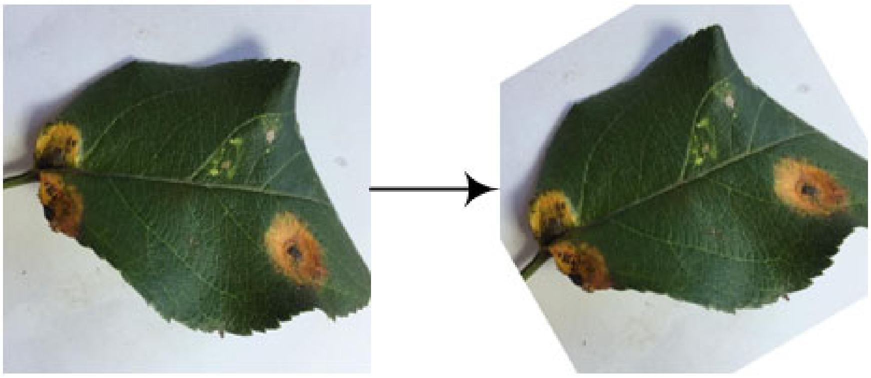
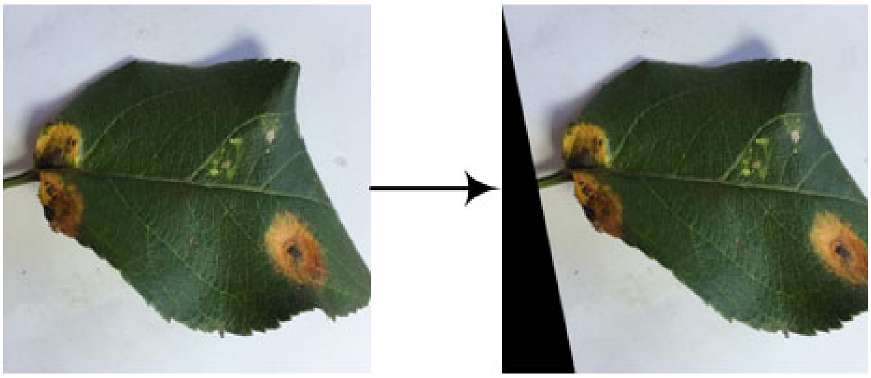
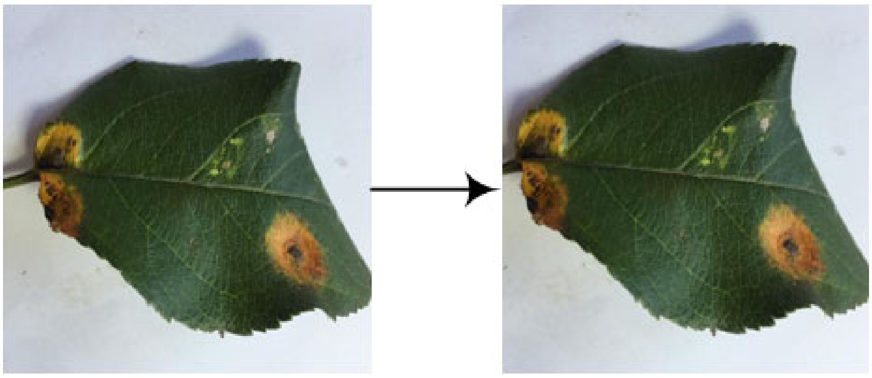
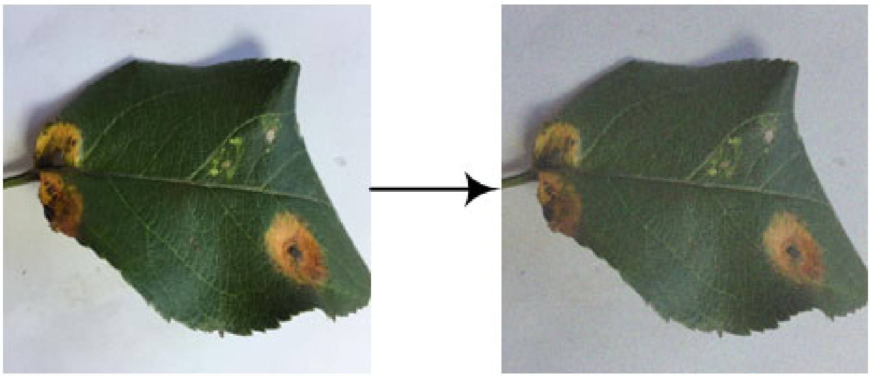
To sum up, d_i^t indicates the overall influence of the selected augmentation methods, a_t, for a given image vector I_t. As every I_t is used to retrain the same pre-trained Deeplab-v3+ model, and the

TABLE 1 Action definition.

a_i	Actions	Examples	Description
0	Original image		The resultant image is the same as the original one.
1	Vertical flip		The resultant image mirrors the original one along the horizontal center line.
2	Horizontal flip		The resultant image mirrors the original one along the vertical center line.
3	Vertical and horizontal flip		The original image is flipped vertically and horizontally to produce the resultant image.

(Continued)

TABLE 1 Continued

a_i	Actions	Examples	Description
4	Clockwise rotation		The original image is rotated 30° clockwise around the center point to produce the resultant image.
5	Affine transformation		The original image is transformed with the matrix [[1, 0.2, 0], [0, 1, 0]] to produce the resultant image.
6	Crop		The first 25 rows and 25 columns of pixels of the original image are trimmed and then the image is resized to 512 × 512 pixels to produce the resultant image.
7	Noise-adding		Some random white Gaussian noise is added to the original image to produce the resultant image.

retrained DeepLab-v3+ model is tested on the same test image set I_{test}, d_i^t can be used for augmentation effect comparison and reward calculation.

2.3 Semantic segmentation model

A semantic segmentation model is integrated into the framework to evaluate the image augmentation effect. Based on the evaluation results, rewards can be produced, and feedback can be provided to the DQN model, which adjusts the Agent policy function accordingly.

2.3.1 Model selection

At present, plant disease segmentation methods based on deep learning mainly include semantic segmentation and instance segmentation. Instance segmentation is more potent as it can distinguish different objects, while semantic segmentation can only determine things from the background. However, the semantic segmentation method is a better choice for this study, as it can meet the verification requirements, is simple and requires less computing resource consumption.

Deep learning-based semantic segmentation methods can improve accuracy and efficiency significantly compared with traditional methods. Currently, commonly used deep learning semantic segmentation models include FCN (Long et al., 2015), U-Net (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), and DeepLab (Chen et al., 2014). The specific analysis is shown in Table 2 (Chen et al., 2017). It can be seen that the DeepLab-v3+ model (Chen et al., 2018) has the highest accuracy and the best application effect. Therefore, the DeepLab-v3+ model is used in this study.

The DeepLab-v3+ model can convert an image into a prediction highlighting diseased areas from the background (Tian et al., 2019). In the rust detection application, each pixel in the apple rust leaf image is assigned to one of the mutually exclusive classes: disease spots VS background, to complete the segmentation of disease spots from the background (Kuang and Wu, 2019).

2.3.2 Deeplab-v3+ model

As shown in Figure 3, the DeepLab-v3+ model adds a simple and effective decoder layer to the DeepLab-v3 model to refine the segmentation results. Furthermore, in the Encoder part, the Atrous Spatial Pyramid Pooling (ASPP) module is constructed using Atrous convolution and the Spatial Pyramid Pooling module (SPP). Atrous convolution is the process of adding spaces between convolution kernel elements to expand the convolution kernel. The SPP performs pooling operations at different resolution levels to capture rich contextual information. Consequently, five different outputs are obtained through the five distinct processes of ASPP to produce a high-level feature, and the Atrous convolution outputs a low-level component. In the Decoder part, the high-level feature is first up-sampled by 4 and then connected with the low-level quality. The concatenation passes through 3 × 3 convolutions and is then up-sampled by 4 to give the predicted label image.

2.3.3 Model evaluation

To evaluate the segmentation effect of the DeepLab-v3+ model from multiple perspectives, the confusion matrix is calculated (Chen and Zhu, 2019), as shown in Table 3.

- K_{TP} is the true positive, indicating the number of disease spot pixels that are correctly classified into the disease spot region.
- K_{FP} is the false positive, indicating the number of background pixels that are wrongly classified into the disease spot region.
- K_{TN} is the true negative, indicating the number of background pixels that are correctly classified into the background region.
- K_{FN} is the false negative, indicating the number of disease spot pixels wrongly classified into the background region.

After that, five performance indexes are defined based on K_{TP}, K_{FP}, K_{TN}, and K_{TN} (Wang et al., 2020).

TABLE 2 Performance comparison of deep learning-based semantic segmentation models.

Proposed time	Network model	Segmentation accuracy	Training time	Algorithm Features
2014	FCN	C	B	Based on the CNN network, it introduces a deconvolution layer.
2014	DeepLab-v1	B	C	It combines dilated convolutions with DCNN networks and optimizes with fully connected conditional random fields.
2015	U-Net	B	–	It is completely symmetrical and the decoder is added with convolution and deepening.
2016	DeepLab-v2	B	C	It uses dilated convolutional layers instead of up-sampling and uses multi-scale spatial pyramid pooling.
2017	SegNet	C	C	It utilizes the encoder-decoder network structure and recovers the image size by up-sampling.
2018	DeepLab-v3+	A	C	It uses an encoder-decoder network structure to improve the segmentation of object edges and introduces dilated convolutions.

A. Very Good, B. Good, C. Fair.

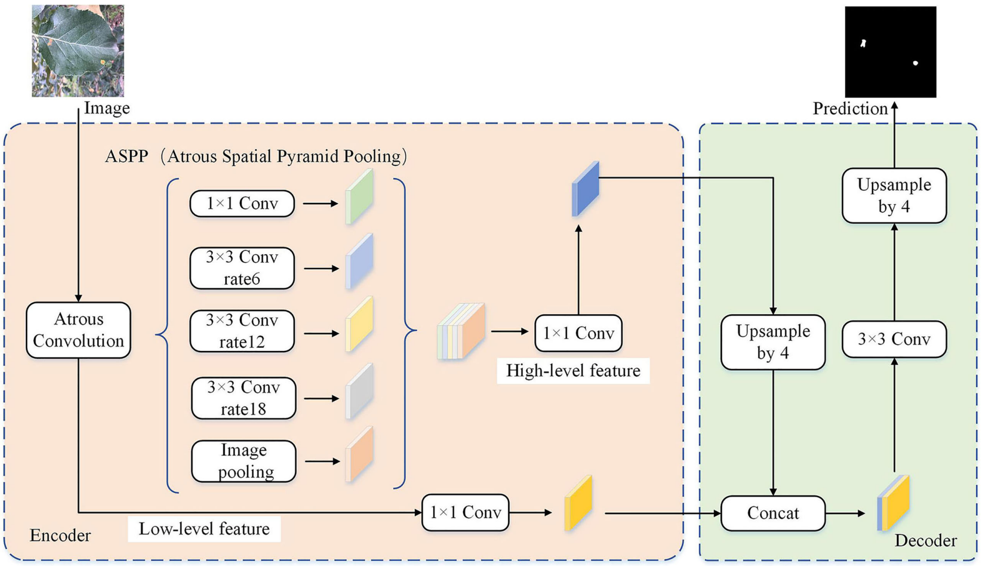


FIGURE 3
The network structure of the DeepLab-v3+ model.

$$P_A = \frac{K_{\mathrm{TP}} + K_{\mathrm{TN}}}{K_{\mathrm{TP}} + K_{\mathrm{TN}} + K_{\mathrm{FP}} + K_{\mathrm{FN}}} \tag{5}$$

where, $P_A \in [0, 1]$ tells how many pixels are correctly classified relative to the total number of pixels.

$$P_{MPA} = \frac{1}{2}\left(\frac{K_{\mathrm{TP}}}{K_{\mathrm{TP}} + K_{\mathrm{FP}}} + \frac{K_{\mathrm{TN}}}{K_{\mathrm{TN}} + K_{\mathrm{FN}}}\right) \tag{6}$$

where, $P_{\mathrm{MPA}} \in [0, 1]$ averages correctly classified disease spot pixels and background pixels relative to the predicted total disease spot pixels and the total background pixels, respectively.

$$P_{CPA} = \frac{K_{\mathrm{TP}}}{K_{\mathrm{TP}} + K_{\mathrm{FP}}} \tag{7}$$

where, $P_{CPA} \in [0, 1]$ tells how many disease spot pixels are correctly classified relative to the predicted total disease spot pixels.

$$P_{IoU} = \frac{K_{\mathrm{TP}}}{K_{\mathrm{TP}} + K_{\mathrm{FN}} + K_{\mathrm{FP}}} \tag{8}$$

where, $P_{IoU} \in [0, 1]$ tells how many disease spot pixels are correctly classified relative to the union of the predicted and expected disease spot pixels.

$$P_{\mathrm{MIoU}} = \frac{1}{2}\left(\frac{K_{\mathrm{TP}}}{K_{\mathrm{TP}} + K_{\mathrm{FN}} + K_{\mathrm{FP}}} + \frac{K_{\mathrm{TN}}}{K_{\mathrm{TN}} + K_{\mathrm{FP}} + K_{\mathrm{FN}}}\right) \tag{9}$$

where, $P_{\mathrm{MIoU}} \in [0, 1]$ averages correctly classified disease spot pixels and background pixels relative to the union of the predicted and expected disease spot pixels and the union of the predicted and expected background pixels, respectively.

2.4 Model training

According to the MDP mentioned above and semantic segmentation models, the main training steps are summarized as follows:

- Preprocessing: Producing leaf labels and rust labels for the original image set and dividing it into the pre-training image set I_0 and the test image set I_{test}; pre-training the DeepLab-v3+ model with I_0, I_{test}, and the leaf labels to generate the LeafIdentifier; pre-training the DeepLab-v3+ model with I_0, I_{test}, and the rust labels to generate the RustIdentifier; selecting DQN as the specific DRL model, and initializing the decision-making Q-function Q_1 and the target Q-function Q_2 for DQN.
- Image augmentation: Taking the child image vector in step $t-1$, i.e., I_{t-1}, as the father image vector in step t; using the LeafIdentifier, RustIdentifier, and the geometric and pixel indicators to process the images in I_{t-1}, one by one, to

TABLE 3 Confusion matrix of disease spot detection.

Pixel point classification area		Expected class	
		Disease spot	Background
Predicted class	Disease spot	K_{TP}	K_{FP}
	Background	K_{FN}	K_{TN}

generate the state vector s_t, i.e., the processing result of one image contributes one element in s_t; using Q_1 to determine one action for each state element, generating the action vector a_t, and one state element corresponds to one action element; executing the action elements in a_t to the corresponding image elements in I_{t-1} to produce the child image vector I_t; getting s_{t+1} from I_t.

- Verification: Constructing the retraining image set, the element of which is $I_0U \quad I_t$ that means I_0 plus I_t gives a training image set; restoring the pre-trained DeepLab-v3+ model; fine retraining the model with $I_0U \quad I_t$; testing the retrained model against I_{test}, storing the results, and calculating the reward r_t; storing $(s_t,\ a_t,\ r_t, s_{t+1})$ into the experience replay buffer.
- DQN network updating: Sampling a batch of data, $(s_i,\ a_i,\ r_i, s_{i+1})$, from the experience replay buffer; calculating the loss function, $L(\theta)$, with Q_1, Q_2, and the sampled data; updating Q_1 with $L(\theta) = [r_i + \max_a Q_2(s_{i+1},\ a) - Q_1(s_i,\ a_i)]^2$ and the backpropagation algorithm; copying the parameters of Q_1 to Q_2 every C steps to update Q_2. Q_2 is updated C times slower than Q_1 for improving stability.
- Starting the next step or a new episode: The above steps except preprocessing are repeated for every step of an episode until the episode ends. To start a new episode, the pre-training image set I_0 is restored as the father image vector for the first step of the episode, and the above steps except preprocessing are repeated until the episode ends.

In summary, the specific DRL algorithm, DQN, is used in this study to organize an adaptive image augmentation scheme. The DQN is assisted with the geometric and pixel indicators for state extraction, the DeepLab-v3+ model for verifying the augmented images and generating the reward, and the image augmentation methods as actions. The image and its accompanying label image are processed in the same way by the selected image augmentation method. The DeepLab-v3+ model is pre-trained once and restored for every retraining operation. DQN parameters keep updating through all the steps and episodes, i.e., they are not reset or restored from a previous step or episode.

3 Experimental results and discussion

3.1 Data sources and image preprocessing

The experimental data comes from the open-source apple leaf disease image dataset on the Baidu AI Studio Development platform, with a resolution of 512 × 512 pixels. Among them, there are 438 images of apple leaf rust, including images collected in various environments, all of which are used in this study. Some representative images are shown in Figure 4A. The EIseg software (Xian et al., 2016) uses the latest deep learning algorithms and models to greatly reduce annotation effort. Therefore, it is used to mark the image, distinguishing the disease spot areas and the whole leaf from the background, to produce labels, as shown in Figures 4B, C. The label images have the same resolution as the original images.

The image set was divided according to the ratio of 8:2, and the image and its label image would not separate during division. As a result, there were 350 images in the pre-training image set I_0, and 88 images in the test image set I_{test}, respectively.

3.2 DeepLab-v3+ model pre-training

The training hardware platform consisted of a Platinum 8358P CPU, a GTX 3090 GPU, and 24 GB of running memory. The software was built with the deep learning framework Pytorch. The testing results indicated that the DeepLab-v3+ model could process about 379 sets of images per second. During training, it took about 4 s to complete each epoch. As DeepLab-v3+ was set to 1,000 epochs in our experiment, it took about 4,000 s in total to complete the pre-training of the DeepLab-v3+ model.

The loss curve and the five performance indexes are shown in Figure 5. The DeepLab-v3+ model converges after about 239 epochs, where the loss is about 3.42e−3. The average P_A, P_{MPA}, P_{MIoU}, P_{CPA}, and P_{IoU} are 0.9956, 0.9444, 0.9131, 0.8905, and 0.8307, respectively. In the verification stage, the pre-trained DeepLab-v3+ model is retrained with $I_0U \quad I_t$ in a fast-fine-tuning way. If the retrained DeepLab-v3+ model can output better performance, the augmented images I_t are said to improve segmentation performance, which means the DRL model can select proper augmentation methods.

3.3 DQN model training

The hardware platform for DQN training consisted of a 24 vCPU AMD EPYC 7642 48-Core processor and a single NVIDIA GTX 3090 GPU with 24 GB of running memory. The DQN algorithm was developed with PyTorch and Python 3.8.10. For each training step of the proposed method, the image augmentation set could be generated in 25 s, and it took about 165 s to complete the parameter fine-tuning of the DeepLab-v3+ model and about 0.003 s to update the parameters of DQN. Therefore, it took about 3.16 min to complete each step and 9.48 min to complete one episode for the proposed method. As DQN was set to 300 episodes in our experiment, it took about 2,844 min in total.

As shown in Figure 6, the reward is very small at the beginning, i.e., −2.975. As the training process progresses, the reward increases significantly and then fluctuates around zero. To sum up, the results show that the reward increases from −2.975 to 0.9826 during DQN training, achieving an improvement of nearly 3.958. That is to say, the effect of the DQN model on disease spot segmentation is greatly improved, which proves that the model can automatically learn how to adopt reasonable and most effective image augmentation methods according to the image features.

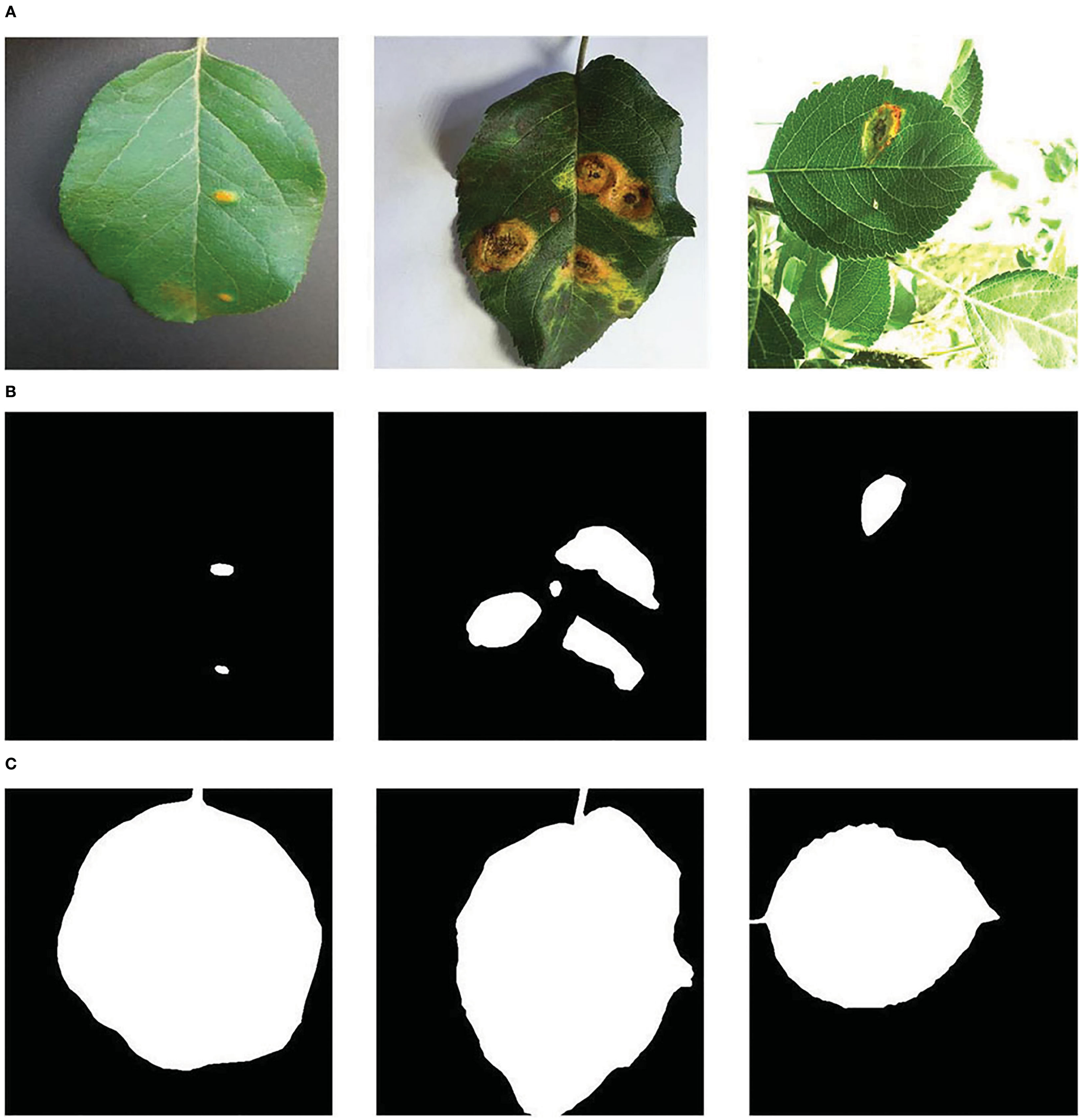


FIGURE 4
Samples of **(A)** the apple rust images, **(B)** the rust labels, and **(C)** the leaf labels.

3.4 Performance comparison of the image augmentation methods

The DQN model was compared with every single method listed in Table 1, i.e., No. 0: *original image*; No. 1: *vertical flip*; No. 2: *horizontal flip*; No. 3: *vertical and horizontal flip*; No. 4: *clockwise rotation*; No. 5: *affine transformation*; No. 6: *crop*; and No. 7: *noise adding*. For the ith ($i = 0 \dots 7$) image augmentation method, the images in I_0 were augmented by the same augmentation method to produce an augmented image set. Then I_0 was combined with the augmented image set to construct a retraining image set. The retraining image set was used to retrain the pre-trained DeepLab-v3+ model, and the retrained model was tested on the I_{test}. This way, a separate set of performance indexes, e.g., P_{IoU} and P_{CPA} , were produced for each image augmentation method for comparison.

Figure 7 shows the augmentation effect of different methods. The *original image* augmentation method achieves an average P_{IoU} value of 0.8117, which is the lowest. The *affine transformation* augmentation method achieves an average P_{CPA} value of 0.9059, which is also the lowest. In contrast, the DQN augmentation method achieves the best performance, with P_{IoU} value of 0.8426 and P_{CPA} value of 0.9255. Therefore, this experimental result confirms the effectiveness of the DQN model in adaptively selecting the augmentation methods according to the image

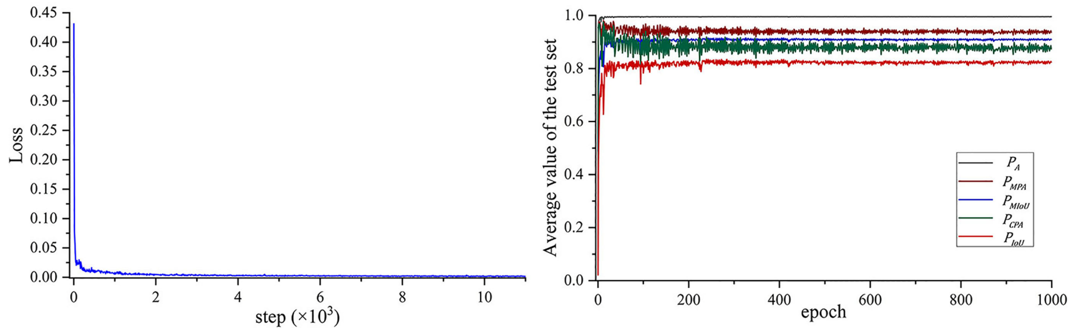


FIGURE 5
Training histories of **(A)** the loss and **(B)** the performance output on the test image set.

features. The testing results showed that the DQN model could generate 12 augmentation image sets (with labels) per second, and the performance was maximum.

3.5 Performance comparison of the semantic segmentation models

The DeepLab-v3+ model (denoted as DQN-DeepLab-v3+) was compared with the FCN and SegNet models. Firstly, the DQN-DeepLab-v3+, FCN, and SegNet models were pre-trained with I_0 and I_{test}, respectively. Secondly, let the proposed DQN model output an augmentation image set. Thirdly, a retraining image set was constructed with I_0 and the augmented image set, and then the retraining image set was used to retrain the DQN-DeepLab-v3+, FCN, and SegNet models, respectively. Finally, the retrained DQN-DeepLab-v3+, FCN, and SegNet models were respectively tested on I_{test} to get a separate set of average performance indexes for comparison.

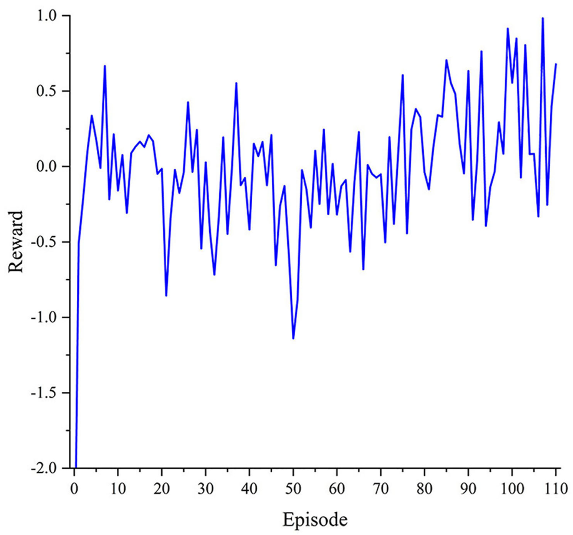


FIGURE 6
Training histories of the reward.

DeepLab-v3+ with random augmentation (denoted as RanAug-DeepLab-v3+) was also constructed for comparison. RanAug-DeepLab-v3+ was pre-trained, retrained, and tested following the same procedure as the DQN-DeepLab-v3+, FCN, and SegNet models. The only difference was that a random augmented image set was used instead of the expanded image set output by the DQN model. Furthermore, the test results of the pre-trained DeepLab-v3+ model were used as the baseline, as any augmented images did not retrain it.

As shown in Figure 8, the proposed DQN-DeepLab-v3+ model achieves the best performance on all the indexes. P_A, P_{MPA}, P_{MIoU} , P_{CPA}, and P_{IoU} reaches 0.9959, 0.9617, 0.9192, 0.9255, and 0.8426, respectively, which are up to 0.2%, 3.7%, 3.9%, 7.3%, and 7.6% higher than other methods. In contrast, the SegNet achieves the worst performance, mainly by focusing on optimizing memory usage. The version of the FCN model is also relatively low due to the limited size of the perceptual area, easy loss of edge information, and low computational efficiency. These results confirm that the DQN-DeepLab-v3+ model is superior to the FCN and SegNet models. On the other hand, some performance indicators of RanAug-DeepLab-v3+ are lower than those of DeepLab-v3+, indicating that the random augmentation tends to harm the segmentation performance. In contrast, the DQN-DeepLab-v3+ model surpasses DeepLab-v3+, showing adaptive augmentation can improve segmentation performance.

4 Conclusion

Deep learning-based automated optical inspection can benefit from image augmentation, which enlarges the image quantity for training and testing. However, one significant challenge is that any single image augmentation method cannot achieve consistent performance over all the images. To address this issue, a DRL-enabled adaptive image augmentation framework is proposed in this paper. The specific DRL algorithm, DQN, is used in this study to organize an adaptive image augmentation scheme. Given an image vector, segmentation models and key indicators are used to extract image features and generate the state vector; the Agent policy function determines the action vector based on the state vector; and the actions produce an augmented image vector. To

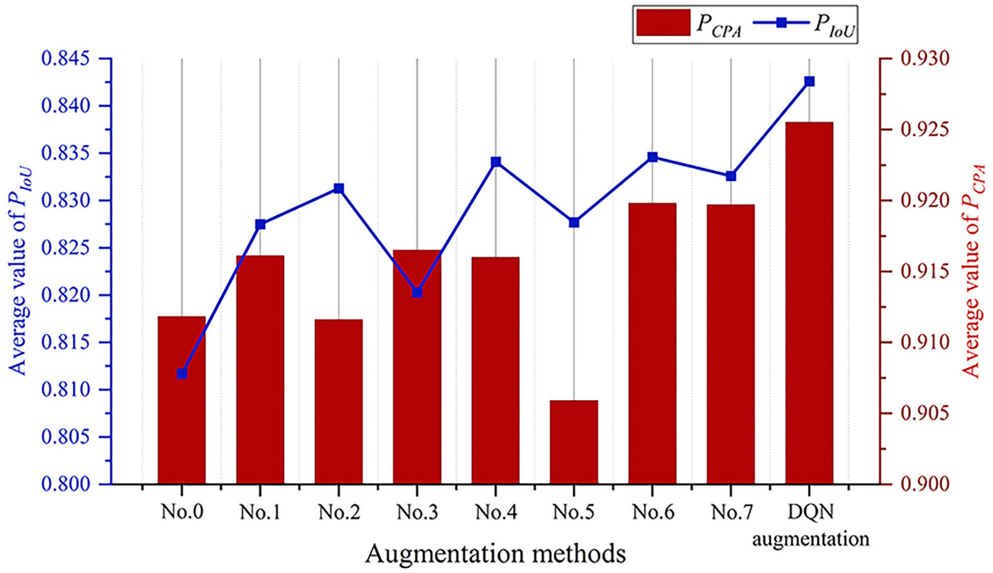


FIGURE 7
Augmentation effect of different methods.

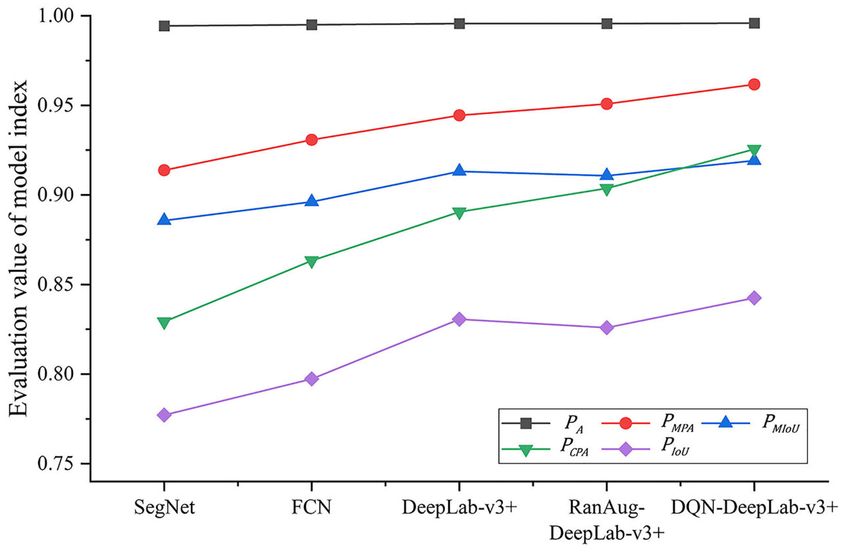


FIGURE 8
Segmentation effect of different models.

evaluate the image augmentation effect, a raised image is used to fine-tune a pre-trained semantic segmentation model, DeepLab-v3 +, and the resultant model is tested against a fixed test image set. Based on the evaluation results, the reward is constructed, and feedback is sent to the DQN model, which updates the Agent policy function accordingly. Through iterations, the Agent policy function is optimized. The proposed DRL-enabled adaptive image augmentation framework achieves better augmentation performance than any single image augmentation method and better segmentation performance than other semantic segmentation models. The experimental results confirm that the DRL-enabled adaptive image augmentation framework can adaptively select augmentation methods that best match the images and the semantic segmentation model.

Future work should consider more advanced image augmentation methods, segmentation targets, and a more flexible and efficient DRL framework to provide more effective detection schemes for complex AOI application scenarios.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://aistudio.baidu.com/aistudio/datasetdetail/11591.

Author contributions

SW, AK, YL, ZJ, HT, SA, MS and UB were responsible for question formulation, method, experimental design, and manuscript writing. YL, ZJ, HT, SA, MS and UB contributed to the issue investigation. HT contributed to the data analysis and AK funded the research. All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This work was supported by the Guangzhou Government Project (Grant No. 62216235), the National Natural Science Foundation of China (Grant No. 622260-1), the Natural Science Foundation of Guangdong Province (Grant Nos. 2022A1515240061 and 2023A1515012975), the Foshan Industrial Field Science and Technology Research Project (Grant No. 2020001006827), and the Characteristic and Innovative Project for Guangdong Regular Universities (Grant No. 2021KTSCX005). The authors thank the Researchers Supporting Project Number (RSP2023R35), King Saud University, Riyadh, Saudi Arabia, and Hainan Provincial Natural Science Foundation of China (No. 123QN182) and Tianjin University-Hainan University Independent Innovation Fund Cooperation Project (NO. HDTD202301, 2023XSU-0035).

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

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 (12), 2481–2495. doi: 10.1109/TPAMI.2016.2644615

Ban, Y., Liu, M., Wu, P., Yang, B., Liu, S., Yin, L., et al. (2022). Depth estimation method for monocular camera defocus images in microscopic scenes. *Electron. (Basel)* 11 (13), 2012. doi: 10.3390/electronics11132012

Bhatti, U. A., Tang, H., Wu, G., Marjan, S., and Hussain, A. (2023). Deep learning with graph convolutional networks: an overview and latest applications in computational intelligence. *Int. J. Intelligent Syst.* 2023, 1–28. doi: 10.1155/2023/8342104

Bhatti, U. A., Yu, Z., Chanussot, J., Zeeshan, Z., Yuan, L., Luo, W., et al. (2021). Local similarity-based spatial–spectral fusion hyperspectral image classification with deep CNN and gabor filtering. *IEEE Trans. Geosci. Remote Sens.* 60, 1–15. doi: 10.1109/TGRS.2021.3090410

Bhatti, U. A., Zeeshan, Z., Nizamani, M. M., Bazai, S., Yu, Z., and Yuan, L. (2022). Assessing the change of ambient air quality patterns in jiangsu province of China pre-to post-COVID-19. *Chemosphere* 288, 132569. doi: 10.1016/j.chemosphere.2021.132569

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017). Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (4), 834–848. doi: 10.1109/TPAMI.2017.2699184

Chen, Z. Z., and Zhu, H. (2019). Visual quality evaluation for semantic segmentation: subjective assessment database and objective assessment measure. *IEEE Trans. Image Process.* 28 (12), 5785–5796. doi: 10.1109/tip.2019.2922072

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)*. 801–818.

Dang, W., Xiang, L., Liu, S., Yang, B., Liu, M., Yin, Z, et al. (2023). A feature matching method based on the convolutional neural network. *J. Imaging Sci. Techn.* doi: 10.2352/J.ImagingSci.Technol.2023.67.3.030402

Deng, Y., Zhang, W., Xu, W., Shen, Y., and Lam, W. (2023). Nonfactoid question answering as query-focused summarization with graph-enhanced multihop inference. *IEEE Trans. Neural Networks Learn. Systems.* doi: 10.1109/TNNLS.2023.3258413

Di, J., and Li, Q. (2022). A method of detecting apple leaf diseases based on improved convolutional neural network. *PLoS One* 17 (2), e0262629. doi: 10.1371/journal.pone.0262629

Elmore, J. G., and Lee, C. S. I. (2021). Data quality, data sharing, and moving artificial intelligence forward. *JAMA Netw. Open* 4 (8), 2. doi: 10.1001/jamanetworkopen.2021.19345

Han, Y., Cameron, J. N., Wang, L. Z., Pham, H., and Beavis, W. D. (2021). Dynamic programming for resource allocation in multi-allelic trait introgression. *Front. Plant Sci.* 12. doi: 10.3389/fpls.2021.544854

He, Z. F., Huang, J. X., Liu, Q., and Zhang, Y. H. (2021). Apple leaf disease segmentation based on asymmetric shuffled convolutional neural network. *Trans. Chin. Soc. Agric. Machinery* 52 (08), 221–230. doi: 10.6041/j.issn.1000-1298.2021.08.022

Jain, A., Sarsaiya, S., Wu, Q., Lu, Y. F., and Shi, J. S. (2019). A review of plant leaf fungal diseases and its environment speciation. *Bioengineered* 10 (1), 409–424. doi: 10.1080/21655979.2019.1649520

Khan, A. I., Quadri, S. M. K., Banday, S., and Latief Shah, J. (2022). Deep diagnosis: a real-time apple leaf disease detection system based on deep learning. *Comput. Electron. Agric.* 198. doi: 10.1016/j.compag.2022.107093

Kuang, H. Y., and Wu, J. J. (2019). Research review of image semantic segmentation technology based on deep learning. *Comput. Eng. Appl.* 55 (19), 12–21+42. doi: 10.3778/j.issn.1002-8331.1905-0325

Ladosz, P., Weng, L. L., Kim, M., and Oh, H. (2022). Exploration in deep reinforcement learning: a survey. *Inf. Fusion* 85, 1–22. doi: 10.1016/j.inffus.2022.03.003

Le, N., Rathour, V. S., Yamazaki, K., Luu, K., and Savvides, M. (2022). Deep reinforcement learning in computer vision: a comprehensive survey. *Artif. Intell. Rev.* 55 (4), 2733–2819. doi: 10.1007/s10462-021-10061-9

Lin, K., Gong, L., Huang, Y. X., Liu, C. L., and Pan, J. (2019). Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. *Front. Plant Sci.* 10. doi: 10.3389/fpls.2019.00155

Liu, H. Y., Jiao, L., Wang, R. J., Xie, C. J., Du, J. M., Chen, H. B., et al. (2022). WSRD-net: a convolutional neural network-based arbitrary-oriented wheat stripe rust detection method. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.876069

Liu, J., and Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods* 17 (1), 22. doi: 10.1186/s13007-021-00722-9

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*. 3431–3440.

Lu, S., Ding, Y., Liu, M., Yin, Z., Yin, L., and Zheng, W. (2023). Multiscale feature extraction and fusion of image and text in VQA. *Int. J. Comput. Intell. Syst.* 16 (1), 54. doi: 10.1007/s44196-023-00233-6

Qin, T. X., Wang, Z. Y., He, K. L., Shi, Y. H., Gao, Y., Shen, D. G., et al. (2020). "Automatic data augmentation *via* deep reinforcement learning for effective kidney tumor segmentation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, New York. 1419–1423.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-Net: convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. 234–241.

Sethy, P. K., Barpanda, N. K., Rath, A. K., and Behera, S. K. (2020). Deep feature based rice leaf disease identification using support vector machine. *Comput. Electron. Agric.* 175, 9. doi: 10.1016/j.compag.2020.105527

Shoaib, M., Hussain, T., Shah, B., Ullah, I., Shah, S. M., Ali, F., et al. (2022). Deep learning-based segmentation and classification of leaf images for detection of tomato plant disease. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.1031748

Shoaib, M., Shah, B., EI-Sappagh, S., Ali, A., Ullah, A., Alenezi, F., et al. (2023). An advanced deep learning models-based plant disease detection: a review of recent research. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1158933

Sun, J., Yang, Y., He, X. F., and Wu, X. H. (2020). Northern maize leaf blight detection under complex field environment based on deep learning. *IEEE Access* 8, 33679–33688. doi: 10.1109/access.2020.2973658

Tian, X., Wang, L., and Ding, Q. (2019). A survey of image semantic segmentation methods based on deep learning. *J. Software* 30 (02), 440–468. doi: 10.13328/j.cnki.jos.005659

Wang, X., Feng, H., Chen, T., Zhao, S., Zhang, J., and Zhang, X. (2021). Gas sensor technologies and mathematical modelling for quality sensing in fruit and vegetable cold chains: a review. *Trends Food Sci. Technol.* 110, 483–492. doi: 10.1016/j.tifs.2021.01.073

Wang, S., Hu, X., Sun, J., and Liu, J. (2023). Hyperspectral anomaly detection using ensemble and robust collaborative representation. *Inf. Sci.* 624, 748–760. doi: 10.1016/j.ins.2022.12.096

Wang, L., Li, X., Gao, F., Liu, Y., Lang, S., Wang, C., et al. (2023). Effect of ultrasound combined with exogenous GABA treatment on polyphenolic metabolites and antioxidant activity of mung bean during germination. *Ultrasonics Sonochem.* 94, 106311. doi: 10.1016/j.ultsonch.2023.106311

Wang, Z. B., Wang, E., and Zhu, Y. (2020). Image segmentation evaluation: a survey of methods. *Artif. Intell. Rev.* 53 (8), 5637–5674. doi: 10.1007/s10462-020-09830-9

Xian, M., Xu, F., Cheng, H. D., Zhang, Y. T., and Ding, J. R. (2016). "EISeg: effective interactive segmentation," in *23rd International Conference on Pattern Recognition (ICPR)*. 1982–1987.

Xiong, H., Lu, D., Li, Z., Wu, J., Ning, X., Lin, W., et al. (2023). The DELLA-ABI4-HY5 module integrates light and gibberellin signals to regulate hypocotyl elongation. *Plant Commun.*, 100597. doi: 10.1016/j.xplc.2023.100597

Xu, C., Zhang, Y., Wang, W. G., and Dong, L. G. (2022). Pursuit and evasion strategy of a differential game based on deep reinforcement learning. *Front. Bioeng. Biotechnol.* 10. doi: 10.3389/fbioe.2022.827408

Yan, Y., Jarvie, S., Liu, Q., and Zhang, Q. (2022). Effects of fragmentation on grassland plant diversity depend on the habitat specialization of species. *Biol. Conserv.* 275, 109773. doi: 10.1016/j.biocon.2022.109773

Yang, Z. H., Sinnott, R. O., Bailey, J., and Ke, Q. H. (2023). A survey of automated data augmentation algorithms for deep learning-based image classification tasks. *Knowl. Inf. Syst.* doi: 10.1007/s10115-023-01853-2

Yang, Z., Xu, J., Yang, L., and Zhang, X. (2022). Optimized dynamic monitoring and quality management system for post-harvest matsutake of different preservation packaging in cold chain. *Foods* 11 (17). doi: 10.3390/foods11172646

Zhang, L., Buatois, L. A., and Mángano, M. G. (2022). Potential and problems in evaluating secular changes in the diversity of animal-substrate interactions at ichnospecies rank. *Terra Nova*. doi: 10.1111/ter.12596

Zhang, W. Z., Zhou, G. X., Chen, A. B., and Hu, Y. H. (2022). Deep multi-scale dual-channel convolutional neural network for Internet of things apple disease detection. *Comput. Electron. Agric.* 194, 11. doi: 10.1016/j.compag.2022.106749

Zhong, Y., and Zhao, M. (2020). Research on deep learning in apple leaf disease recognition. *Comput. Electron. Agric.* 168, 6. doi: 10.1016/j.compag.2019.105146

Zhou, S. K., Le, H. N., Luu, K., Nguyen, H. V., and Ayache, N. (2021). Deep reinforcement learning in medical imaging: a literature review. *Med. Image Anal.* 73, 20. doi: 10.1016/j.media.2021.102193

Zhou, G. X., Zhang, W. Z., Chen, A. B., He, M. F., and Ma, X. S. (2019). Rapid detection of rice disease based on FCM-KM and faster r-CNN fusion. *IEEE Access* 7, 143190–143206. doi: 10.1109/access.2019.2943454

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 25 July 2023
DOI 10.3389/fpls.2023.1181322

Check for updates

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU), United States

REVIEWED BY
Milind Ratnaparkhe,
ICAR Indian Institute of Soybean Research, India
Haozhou Wang,
The University of Tokyo, Japan

*CORRESPONDENCE
Wei Lu
njaurobot@njau.edu.cn
Guangnan Xing
xinggn@njau.edu.cn

RECEIVED 09 March 2023
ACCEPTED 06 July 2023
PUBLISHED 25 July 2023

CITATION
He W, Ye Z, Li M, Yan Y, Lu W and Xing G (2023) Extraction of soybean plant trait parameters based on SfM-MVS algorithm combined with GRNN. *Front. Plant Sci.* 14:1181322. doi: 10.3389/fpls.2023.1181322

Extraction of soybean plant trait parameters based on SfM-MVS algorithm combined with GRNN

Wei He[1], Zhihao Ye[2], Mingshuang Li[3], Yulu Yan[2], Wei Lu[3]* and Guangnan Xing[2]*

[1]College of Engineering, Nanjing Agricultural University, Nanjing, China, [2]Soybean Research Institute, Ministry of Agriculture and Rural Affairs (MARA) National Center for Soybean Improvement, Ministry of Agriculture and Rural Affairs (MARA) Key Laboratory of Biology and Genetic Improvement of Soybean, National Key Laboratory for Crop Genetics & Germplasm Enhancement and Utilization, Jiangsu Collaborative Innovation Center for Modern Crop Production, College of Agriculture, Nanjing Agricultural University, Nanjing, China, [3]College of Artificial Intelligence, Nanjing Agricultural University, Nanjing, China

Soybean is an important grain and oil crop worldwide and is rich in nutritional value. Phenotypic morphology plays an important role in the selection and breeding of excellent soybean varieties to achieve high yield. Nowadays, the mainstream manual phenotypic measurement has some problems such as strong subjectivity, high labor intensity and slow speed. To address the problems, a three-dimensional (3D) reconstruction method for soybean plants based on structure from motion (SFM) was proposed. First, the 3D point cloud of a soybean plant was reconstructed from multi-view images obtained by a smartphone based on the SFM algorithm. Second, low-pass filtering, Gaussian filtering, Ordinary Least Square (OLS) plane fitting, and Laplacian smoothing were used in fusion to automatically segment point cloud data, such as individual plants, stems, and leaves. Finally, Eleven morphological traits, such as plant height, minimum bounding box volume per plant, leaf projection area, leaf projection length and width, and leaf tilt information, were accurately and nondestructively measured by the proposed an algorithm for leaf phenotype measurement (LPM). Moreover, Support Vector Machine (SVM), Back Propagation Neural Network (BP), and Back Propagation Neural Network (GRNN) prediction models were established to predict and identify soybean plant varieties. The results indicated that, compared with the manual measurement, the root mean square error (RMSE) of plant height, leaf length, and leaf width were 0.9997, 0.2357, and 0.2666 cm, and the mean absolute percentage error (MAPE) were 2.7013%, 1.4706%, and 1.8669%, and the coefficients of determination (R2) were 0.9775, 0.9785, and 0.9487, respectively. The accuracy of predicting plant species according to the six leaf parameters was highest when using GRNN, reaching 0.9211, and the RMSE was 18.3263. Based on the phenotypic traits of plants, the differences between C3, 47-6 and W82 soybeans were analyzed genetically, and because C3 was an insect-resistant line, the trait parametes (minimum box volume per plant, number of leaves, minimum size of single leaf box, leaf projection area).The results show that the proposed method can effectively extract the 3D phenotypic structure information of soybean plants and leaves without loss which has the potential using ability in other plants with dense leaves.

KEYWORDS

structure from motion, soybean plant, 3D point cloud, plant phenotype, 3D trait extraction

1 Introduction

Soybean is an important grain and oil crop worldwide and is rich in high-quality protein, unsaturated fatty acids, isoflavones, and other nutrients (Zhang T et al., 2019). The phenotypic morphological characteristics embodied in the growth process play an important role in the selection of excellent soybean varieties (Zhu et al., 2020), and the phenotypic state of plants is the physical manifestation of the genotype (Alonge et al., 2020), which is not only of great significance for the quantitative analysis of genotype-environment interactions (Barker et al., 2019; Van Eeuwijk et al., 2019), but also for breeding activities, such as optimal cultivation, fertilization, and irrigation of plants (Chawade et al., 2019; Li et al., 2021). Phenotypes are prone to changes in response to genetic mutations and environmental influences (Vogt, 2021), which are the main bottlenecks limiting the expansion of genomics in plant sciences, animal biology, and medicine. Different genes determine different insect resistance in plants, affecting plant phenotypes (Tyagi et al., 2020). Therefore, accurate and non-destructive acquisition of soybean phenotypic parameters is essential for the study of soybean plants and breeding of insect-resistant varieties.

Chen et al. (2021). constructed the 3D model of soybean plant can efficiently obtain its geometric characteristics and morphological traits, which is essential for understanding plant growth and plant response to biotic and abiotic stresses, so as to estimate the growth rate of soybean plants and predict the tolerance of stress, it greatly reduces the marginal cost of collecting multiple morphological traits across multiple time points, which has important theoretical significance and practical value for soybean variety selection and breeding, scientific cultivation and fine management (Wang et al., 2022). By means of the 3D model of the plant, the growth situation and specific changes of the plant can be quickly understood, which contributes to screen out excellent varieties with high quality and strong insect resistance, and can also lay the foundation for the genetic improvement of soybean and breed better varieties (Xue et al., 2023).

The traditional methods used to obtain plant phenotypic parameters include manual measurement, two-dimensional (2D) image measurements, and precision instrument measurements. Manual measurements are slow, costly, and subjectively inaccurate (Gage et al., 2019), which can easily damage plants during measurement. When plant phenotypic parameters are measured based on 2D image technology (Das Choudhury et al., 2020; Li et al., 2020; Omari et al., 2020; Kuett et al., 2022), critical spatial and volumetric information, such as thickness, bending, and orientation, is easily lost during data conversion from three-dimensional (3D) to 2D states, and the morphology will also be blocked from different perspectives (Martinez-Guanter et al., 2019). Precision instruments, such as handheld laser scanners (Artec EVA laser scanners and FastSCAN laser scanners) (Ma et al., 2022), 3D laser scanning, and radar technology (FARO Focus3D 120 laser scanning of ground objects) (Junttila et al., 2021; Nguyen et al., 2022), are often used to measure plant phenotypic traits. Although it has a high resolution and can reconstruct the 3D model of the plant with high precision and record the phenotypic information of the plant (Ao et al., 2022), its acquisition speed is slow, the equipment is expensive, and the lack of color information for the obscured parts of plants fails to accurately reflect phenotypic traits. In addition, for automatic analysis of plant phenotypic information, 3D point clouds generated by laser scanners must be correctly extracted from a large amount of 3D data and classified for this purpose. The high cost and limited availability of laser-scanning equipment hinder its wide applications.

Recently, scholars have been increasingly interested in the structure from motion (SFM) algorithm based on multi-view stereo measurement, and a series of exploratory studies have been carried out in the fields of geographical environment and agriculture. The 3D model can be automatically reconstructed according to overlapping 2D digital image sets (Jiang et al., 2020), which has the advantages of being self-calibrated, less constrained by the environment, and functional both indoors and outdoors, and has been widely used in 3D reconstruction (James et al., 2019; Swinfield et al., 2019). Ewertowski et al. (2019) used UAV combined with this technology to quickly and ultra-high-resolution 3D reconstruction of glacier landforms, and drew the terrain related to glaciers in detail. In the field of agriculture, He et al. (2017) used this technology to obtain 3D models of strawberries and used custom software to process point cloud data and obtain seven agronomic traits of strawberries. Huang et al. (2022) used the DoidiltenGAN image enhancement algorithm combined with SFM-MVS algorithm to develop a set of agricultural equipment that could accurately perceive the growth of crops under low light. Hui et al. (2018) used this technology to obtain 3D point clouds for cucumbers with flat leaves, peppers with small leaves, and eggplants with curly leaves. With the help of precision instruments and Geomagic Studio software, they measured five characteristic parameters of the plant, including leaf length, leaf width, and leaf area, and analyzed the errors between them. In (Xu et al., 2019), a UAV was used in combination with this technology to obtain a 3D model of cotton, and a DEM was used to measure four phenotypic traits, such as plant height and canopy coverage. In (Piermattei et al., 2019), this technology was used to obtain 3D point clouds of trees and four parameters, such as DBH and the number of trees. With the rising demand for different types of phenotypic information from 3D point clouds, Rahman et al. (2017) explored future research on volume measurement and modeling using this method to obtain 3D models.

These studies show that the SFM algorithm has good potential in the field of plant phenotype detection. However, at present, the analysis of phenotypic trait parameters of plants is limited, most software is used, and there is a lack of technology for reconstruction and phenotype measurement of plants with various and dense leaves. Therefore, in this study, we combined structure from motion (SFM) with multiple view stereo (MVS) methods to build a platform for acquiring plant sequence images. Using the soybean seedlings with different gene expression patterns of the same soybean plant at the R4 stage as the research object, the point cloud models were obtained by 3D reconstruction using different sequence images, the LPM algorithm was used to quickly perform non-destructive phenotype measurements, and the accuracy of phenotype measurement was evaluated. The feasibility of SFM-MVS technology combined with the LPM algorithm is explored and the phenotype and insect resistance of soybean plants are analyzed.

At present, machine learning (ML) and deep learning (DL) algorithms are widely used in the plant phenotype classification. For machine learning (ML), Tan et al. (2021) used the machine learning (ML), based on tomato cultivation as well as disease datasets to classify plant diseases; Barradas et al. (2021) applied different machine learning (ML) methods such as Decision Tree (DT), Random Forest (RF), and Extreme Gradient Boosting (XGBoost) to classify plants into three drought stress levels; Alam et al. (2020) used random forests (RF) for detection and classification of weeds as well as crops and accurate identification and control of weeds. For deep learning (DL), Ferentinos et al. (2018). made use of Convolutional Neural Networks (CNN) to classify plant disease images; Brugger (2022). analyzed spectral data of plant phenotypes based on deep learning (DL) to forecast plant diseases and categories; Cardellicchio et al. (2023) used YOLOv5 to recognize fruits, flowers and the colors of objects; Azimi et al. (2021) took advantage of deep learning (DL) to classify stress in plant shoots based on plant phenotype images; Zhou et al. (2021) applied advanced deep learning (DL) methods based on convolutional neural networks to carry out the analysis of corn phenotype. The above researches show that DL/ML has favorable potential in the classification of plant phenotype, but the obtained plant morphological traits are comparatively single and there are few studies to predict plant species and analyze insect resistance genotypes based on the morphological traits of leaves, and the related ML/DL models are highly susceptible to the influence of environment, images, data sets, etc. during the implementation of detection. In this paper, we will try to solve the above problems.

To evaluate crops based on soybean plant phenotypic information, the traditional popular machine learning (ML) often uses Shallow Neural networks, such as support vector machine (SVM), back propagation neural network (BP), generalized regression neural network (GRNN), and other models based on small datasets are often applied to construct plant gene-insect resistance models in the field of agricultural engineering (Kamilaris and Prenafeta-Boldú, 2018). Deep learning techniques, such as deep neural networks (DNN) (Du et al., 2019) , convolutional neural networks (CNN) (Cong et al., 2019) , recurrent neural networks (RNN) (Yu et al., 2019), and residual neural networks (Resnet) (Alom et al., 2019), require a large amount of data for modeling and are significantly less effective than shallow neural networks for small data (Chlingaryan et al., 2018). Owing to the difficulty of soybean phenotypic data collection, therefore, we constructed a small data set between plant phenotypes and varieties. Based on this, we used popular shallow neural networks such as Support Vector Machine (SVM), Back Propagation Neural Network (BP) and Generalized Regression Neural Network (GRNN)to build the model respectively to classify its species based on the phenotypic characteristics of soybean leaves.

Therefore, the aim of this study is to accurately extract phenotypic trait parameters from the leaves of plants with different gene expression forms of the same variety using the LPM algorithm based on the application of the SFM algorithm combined with the MVS reconstruction technique in plants. It will construct a triple linkage between genotype-phenotype-insect resistance and establish a prediction and classification model of soybean varieties. This study is organized as follows: (1) A 3D target acquisition system based on the SFM algorithm combined with MVS reconstruction technology is designed and constructed to perform 3D reconstruction of soybean plants with different gene expression forms (ko-Williams82, oe-Williams82, and Williams82) of the same variety and obtain their 3D point cloud models. (2) Point cloud data, such as individual plants, stems, and leaves, are automatically segmented using low-pass filtering, Gaussian filtering, ordinary least squares (OLS) plane fitting, and Laplacian smoothing. (3) Eleven phenotypic parameters of the leaves, including length, width, volume, projection area, projection length, tilt information and so on, are obtained using the LPM algorithm. (4) The reconstruction accuracy of the SFM-MVS algorithm is analyzed using regression evaluation indicators (RMSE, MAPE, R^2), and the association between genotype, phenotype, and insect resistance is constructed by combining the plant penetrance parameters of different gene expression forms. (5) Three models, SVM, BP, and GRNN, are constructed to compare the prediction and classification models of soybean species based on six characteristic phenotypic parameters of leaves.

2 Materials and methods

2.1 Experimental materials and data acquisition

Three soybean varieties, ko-Williams82, oe-Williams82, and Williams82 (hereinafter referred to as C3, 47-6, and W82, respectively) were selected from the Baima Base of Nanjing Agricultural University. There were 15 plants of each variety (planted in three replicates, each in a separate row with five plants of each variety in a row), and a total of 45 soybean plant samples were collected. The soybean row spacing was 40 cm and the plant spacing was 80 cm. For the convenience of data processing in the later stage, the experimental samples were planted with potted plants (diameter of 27 cm; height of 21 cm) to avoid occlusion between plants. The soil used for soybean planting was first dried in the sun, then the dried soil was first crushed, and then the stones and weeds in the soil were removed through a 6 mm mesh screen to ensure the homogeneity of the soil. Finally, the sieved soil and nutrient soil (organic matter content >15%, total N, P, and K content >0.88%, ph7~7.5) were divided into 3:1 evenly mixed, loaded quantitatively into a plastic pot with a diameter of 30 cm, and water added to make the absolute water content of the soil 30%. Five soybean seeds were placed in each pot at a sown depth of 3.0 cm. The soybean plants were placed in a net chamber and provided normal water and fertilizer management during soybean growth. When the soybean grew to R4 stage, the density of one spot bug per plant was used for insect treatment. After 10 days of damage, dynamic non-destructive measurement and manual comparison verification of plant height, leaf length, leaf width, and other parameters of soybean plants were carried out, and the association between soybean plant genotype, phenotype, and insect resistance was established.

A smartphone (iPhone 11) was used as the acquisition device to capture the soybean plant for 40 s. The resolution was set to 1080p HD, 60fps before video acquisition to ensure the universality of the video acquisition device. To avoid the influence of smart phone mirror shooting on 3D reconstruction, an electric turntable (diameter of 26 cm) with a speed of 0.05 r/s and a load bearing of 40 KG was used as the plant bearing platform. The smartphone was placed on a scaffold with a height of 45 cm at a distance of 25 cm from the plant, and the data at different angles of the plant were collected by tilting down 30° at a horizontal height of approximately 30 cm above the plant. The carrying platform was rotated for two weeks for video shooting, and 300 multi-view images were extracted by frame in JPG format with 1080×1920 resolution. The back and bottom of the platform were covered with a black fleece to ensure a stable and reliable recording environment and to minimize noise interference (Figure 1).

The specific steps of the manual measurement of soybean plant height, leaf width, and leaf length are as follows. Four workers measured the height of the same soybean plant using a scale ruler as the reference line along the basin and measured the leaf length (from leaf base to leaf tip, excluding petiole) and leaf width (the widest part on the leaf that is perpendicular to the main vein) of all the leaves of each soybean plant using a standard calculation paper with a straight ruler. The average of the readings of the four workers was taken as the final manually measured value of the phenotypic parameters of the soybean plant.

The software used for the experiment was Free Studio, the 3D reconstruction open-source software Visual SFM, and MATLAB 2022a. The electric turntable worked continuously for 40 s at a speed of 0.05 r/s to obtain the image video of the soybean plant. Three hundred multi-view images were extracted from the video obtained by frame. To ensure a large amount of accurate point cloud data, the ROI were selected from the multi-view images of the plant, and the point cloud data were generated by 3D reconstruction. The point cloud data were sampled and denoised; low-pass filtering, point cloud clustering, OLS fitting, and Laplacian smoothing were used. Parameters, such as plant height, the number of leaves, leaf length, leaf width, minimum bounding box volume of a single plant, minimum bounding box volume of a single leaf, the volume of a leaf, leaf projection area, projection length, projection width, and angle were automatically measured using the maximum traversal and greedy projection triangle algorithms. The accuracy and robustness of the SFM reconstruction of soybean plants were evaluated and compared with the manual measurement of plant height, leaf length, and leaf width.

2.2 Overall process of SFM-MVS method for reconstructing 3D model of soybean plants

In this study, the SFM-MVS method was used to reconstruct the 3D models of soybean plants. A workflow diagram is shown in Figure 2. It consists of seven steps: (1) capturing multi-view images of soybean plants; (2) selecting the Plant ROI; (3) finding key points from multi-view images and reconstructing the 3D point cloud of the plant; (4) filtering and segmentation algorithms to separate leaves and stems; (5) reconstructing the smooth surface of the leaf point cloud using the plane fitting algorithm and the Laplacian smoothing algorithm; (6) extracting and evaluating plant structural phenotype parameters based on the distance maximum traversal algorithm and the greedy projection triangulation algorithm; and (7) establishing the identification of soybean varieties based on phenotypic information.

2.3 Extraction of ROI from soybean plants

This study proposes an improved detection and matching strategy to accurately obtain the key feature points of multi-view images and improve the efficiency of feature matching (Figure 3). The proportion of the region of interest (ROI) is increased by cropping the original image, and the scale of the image is reduced to reduce the number of calculations for feature detection.

The preliminary segmentation of soybean plant regions in multi-view images based on the ROI algorithm is a key part of the 3D reconstruction. The multi-view image sequence is cropped based on the ROI of each image, effectively reducing the resolution of the image and increasing the proportion of the soybean plant in the whole image. The rate of generation of dense point clouds was increased by 81.62% by the SFM-MVS algorithm for the 3D reconstruction of soybean plants after soybean plant ROI extraction.

2.4 3D model reconstruction of soybean plants

We used VisualSVM software to conduct the standard sfm-mvs workflow and obtained the plant point clouds. The process of 3D model reconstruction, as shown in Figure 4. The main steps in soybean plant 3D model reconstruction are feature point extraction and matching, sparse point cloud reconstruction, and dense point cloud reconstruction.

2.5 Processing of soybean plants point cloud data

As a result of the many dense leaves of soybean plants (Figure 5A), the reconstructed data were large and interspersed with a number of noisy background point clouds (Figure 5B). Point cloud data sampling, denoising, optimization, coordinate

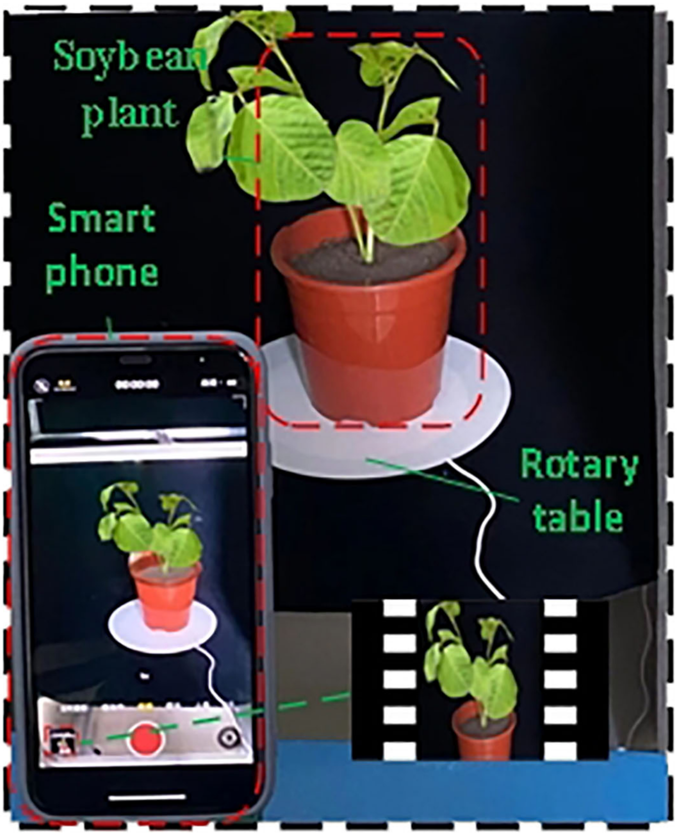


FIGURE 1
3D object acquisition platform.

correction, and other processes are required because the soybean 3D point cloud model is inconsistent with the actual plant in the standard 3D space direction and scale (Figure 5C).

2.5.1 Sampling of point cloud data

Owing to the large redundancy, long reconstruction time, and low efficiency of 3D point cloud data reconstructed using Visual SFM software, a point cloud simplification algorithm based on voxelized grid downsampling was used. Voxelized grid downsampling creates a minimum 3D voxel grid based on the point cloud data (Han et al., 2017), divides the point cloud data into a 3D voxel grid, selects a data point as the center of gravity point of the grid, and retains the data point closest to the center of gravity of the small grid. This method is simple, efficient, and does not require the establishment of a complex topological structure to simplify point cloud data, reduce operation time, and improve the program running speed (Liang et al., 2020). As shown in Figure 5B, the number of point clouds was reduced to 11% of that presented in Figure 5A, and the soybean plant phenotype did not show any change, which did not affect the extraction of its phenotypic shape parameters.

2.5.2 Point cloud denoising

Owing to the influence of a series of external factors, such as data sampling equipment, external environment, and experience of experimental operators, noise points and outliers in the reconstruction process have adverse effects on trait extraction, feature matching, and surface reconstruction (Li and Cheng, 2018). A low-pass filtering algorithm was used to locally fit the soybean, and the appropriate threshold (Points/Radius was set to 0.098264, Maxerror was set to 2) was set to remove the points that deviated from the fitting plane. The background noise and most of the edge noise were removed by setting the RGB of the background (the main background noise in this study was the point cloud of the soil and basin along the color). The denoising effect of the 3D point cloud of the soybean plant is shown in Figure 5C, where the number of point clouds was reduced to 89% of the number of point clouds of a single plant after sampling. As shown in Figure 5B, the reduced points were background noise points.

2.5.3 Coordinate correction of point cloud data

(1) To accurately extract the phenotypic trait parameters of soybean plants, coordinate correction is required for the 3D point cloud of soybean, and the proportional coordinates are calculated using the potted plant as the reference. The length of the potted plant in the point cloud data was calculated using the Euclidean distance algorithm and converted to obtain the transformation coefficients to obtain the true coordinates of the soybean plant. The calculation formula is as follows:

$$(x, y, z) = \alpha(x', y', z') \tag{1}$$

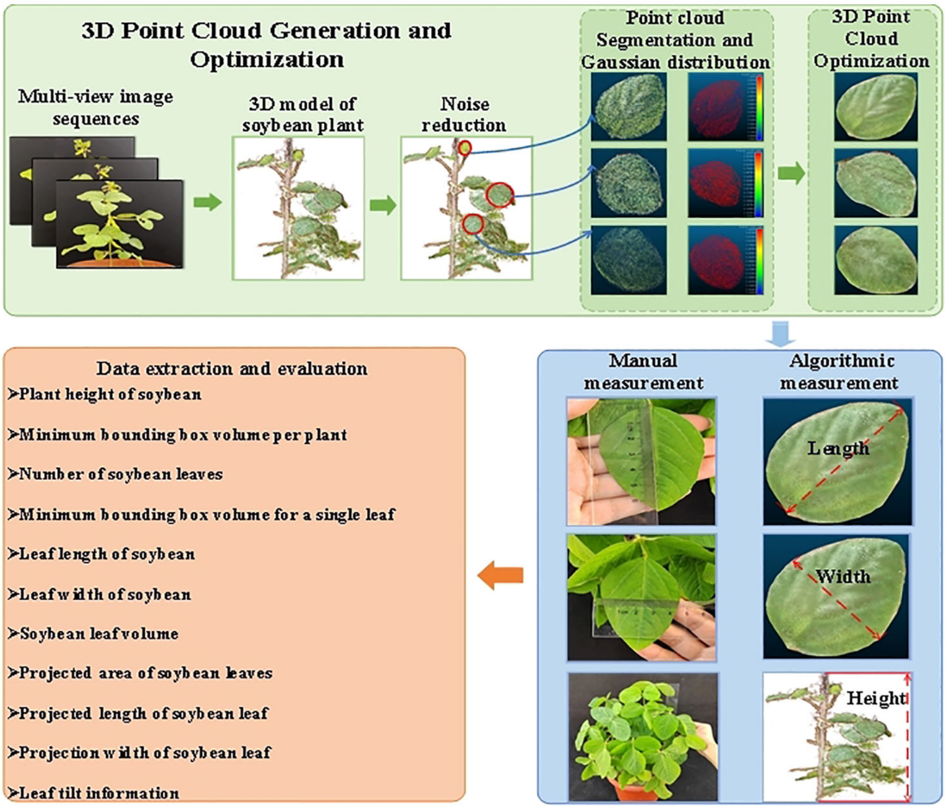

3D Point Cloud Generation and Optimization
Multi-view image sequences
3D model of soybean plant
Noise reduction
Point cloud Segmentation and Gaussian distribution
3D Point Cloud Optimization
Data extraction and evaluation
➢Plant height of soybean
➢Minimum bounding box volume per plant
➢Number of soybean leaves
➢Minimum bounding box volume for a single leaf
➢Leaf length of soybean
➢Leaf width of soybean
➢Soybean leaf volume
➢Projected area of soybean leaves
➢Projected length of soybean leaf
➢Projection width of soybean leaf
➢Leaf tilt information
Manual measurement
Algorithmic measurement
Length
Width
Height

FIGURE 2
Workflow of 3D reconstruction and accuracy evaluation.

where (x, y, z) is the length of reference in the point cloud, $(x',$ $y', z')$ is the real length of reference, and α is the transformation coefficient of point cloud coordinates.

(2) The random sample consensus algorithm (RANSAC) is used to detect the ground and obtain the normal vector of the ground $\vec{m}$, and the rotation angle θ is obtained by combining the normal vector $\vec{n}(0, 0, 1)$ of the Z-axis. The rotation matrix can be obtained by using the Rodriguez rotation formula, and the calculation formula is as follows:

$$\vec{m} \cdot \vec{n} = m * n * \cos\theta \tag{2}$$

$$\theta = \cos^{-1}\left(\frac{\vec{m} \cdot \vec{n}}{m * n}\right) \tag{3}$$

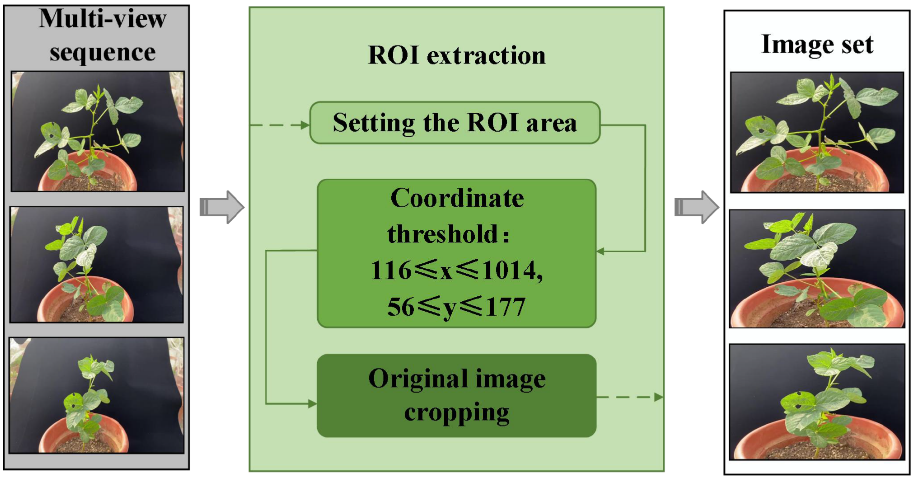


FIGURE 3
Clipping of the ROI.

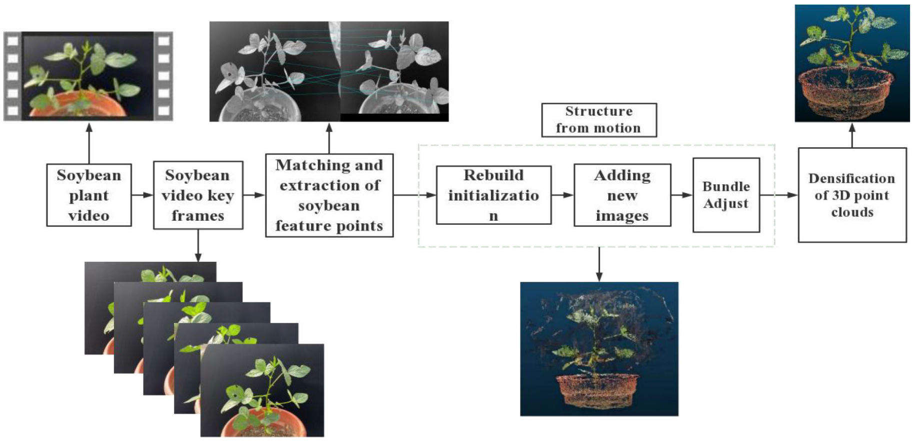


FIGURE 4
3D model reconstruction process.

$$R_{rot} = \vec{E} * \cos\theta + (\vec{m} \cdot \vec{n}) * \vec{d} * R(\theta) + (\vec{m} * \vec{n}) * \sin\theta \tag{4}$$

$$R_{rot} = \begin{bmatrix} \cos\theta + d_1 R(\theta) & d_1 d_2 (R(\theta) - d_3 \sin\theta) & d_2 \sin\theta + d_1 d_3 R(\theta) \\ d_3 \sin\theta + d_1 d_2(\theta) & \cos\theta + d_2^2 R(\theta) & -d_1 \sin\theta + d_1 d_2 R(\theta) \\ -d_2 \sin\theta + d_1 d_3 R(\theta) & d_1 \sin\theta + d_2 d_3 R(\theta) & \cos\theta + d_3^2 R(\theta) \end{bmatrix} \tag{5}$$

where defined $R(\theta) = 1 - \cos\theta$, respectively, m and n are respectively the lengths of $\vec{m}$ and ethe $\vec{n}$, $\vec{E}$ is the third-order identity matrix, θ is the rotation angle, and $\vec{d}(d_1, d_2, d_3)$ is the unit vector of $\vec{m} * \vec{n}$.

2.5.4 Point cloud segmentation

The 3D point cloud segmentation of soybean plants mainly aims to segment and extract the leaves and stems of soybean plants, as shown in Figure 6. A gap exists between any two leaves, which is a prerequisite for individual leaf segmentation. A point cloud clustering algorithm was used to segment different parts of the leaves, a cylindrical fit to the stalk of the soybean plant based on a random sampling consistency algorithm, and a statistical method to remove noise and extraneous points from the root part of the leaves was used.

2.5.5 Point cloud optimization

After the point cloud segmentation of leaves and stalks, white noise generated by surface reflection or occlusion around leaves was removed based on the difference between the color of the noise and the characteristics of the leaf point cloud. The KD-Tree was used to determine the point cloud data and the distance between the fields, and the point cloud density was obtained by statistical analysis. Clutter was eliminated using the data analysis method, and the calculation formula is as follows:

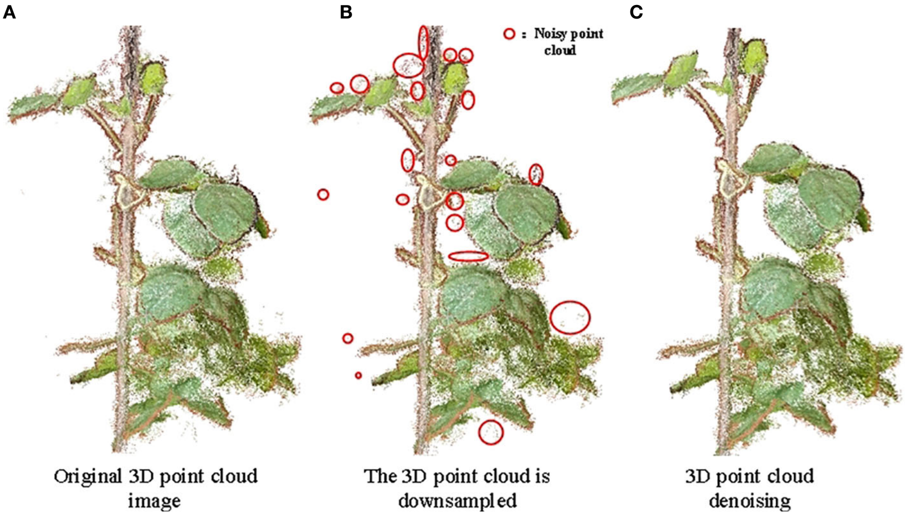

A
B
C
○ : Noisy point cloud
Original 3D point cloud image
The 3D point cloud is downsampled
3D point cloud denoising

FIGURE 5
Down-sampling and denoising effect of soybean point cloud. **(A)** Original 3D point cloud image; **(B)** The 3D point cloud is downsampled; **(C)** 3D point cloud denoising.

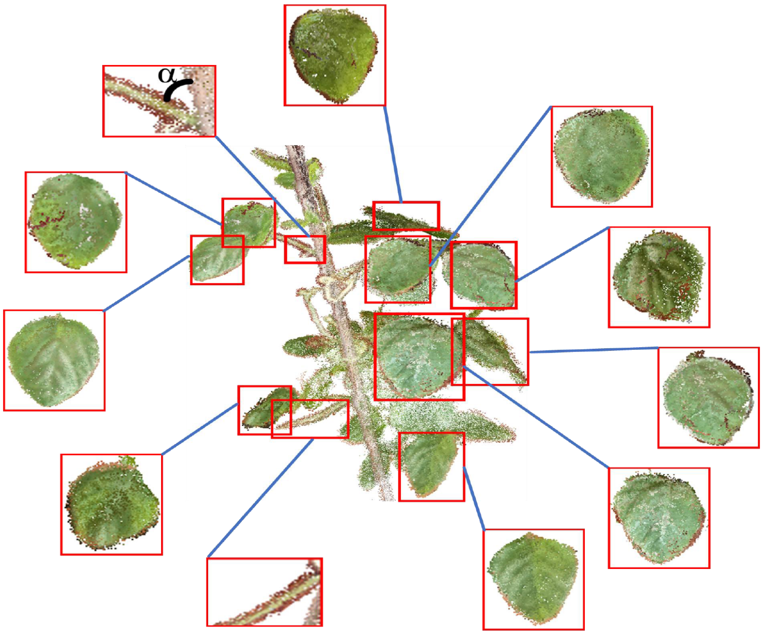


FIGURE 6
Effect of point cloud segmentation.

$$d_i = \sqrt{\frac{(x_{ij} - x_i)^2 + (y_{ij} - y_i)^2 + (z_{ij} - z_i)^2}{k}} \tag{6}$$

$$\overline{d_i} = \frac{\sum_{i=1}^{n} d_i}{n} \tag{7}$$

$$\sigma = \frac{\sum_{i=1}^{n} (d_i - \overline{d_i})^2}{n} \tag{8}$$

where, d_i is the distance between soybean point cloud and other K adjacent areas, $\overline{d_i}$ is the average value of the d_i, σ standard deviation of soybean.

To better realize the effect of Gaussian filtering, scalar fields were used to establish the Z-coordinate axis and draw the chromatographic diagram of the point cloud in Figure 7A. The Gaussian filter algorithm was used to set the covariance of the Gaussian filtering, draw the Gaussian distribution and filtering result diagram of the soybean point cloud, which are shown in Figures 7B, C.

The OLS plane fitting method was used to find the best matching function by minimizing the square error (Rannik et al., 2020) for the plane fitting of soybean leaves. The Laplacian smoothing algorithm was used to smooth the edges and surfaces of the soybean leaves after the initial fitting. A statistical filtering

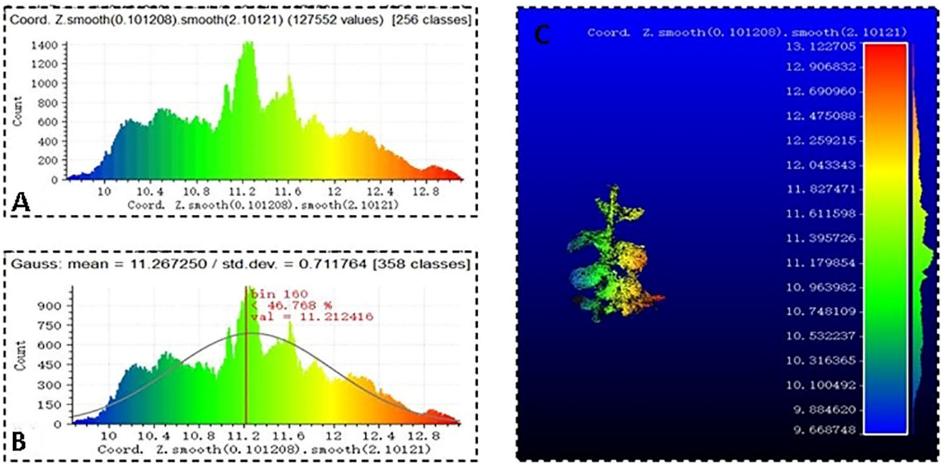


FIGURE 7
Point cloud Gaussian filtering, **(A)** soybean point cloud chromatogram, **(B)** soybean Gaussian distribution, and **(C)** filtering result.

algorithm was used to optimize the soybean stalks. A 3D soybean point cloud model was obtained by splicing the optimized point cloud leaf and stem models.

2.6 The LPM algorithm was used to extract soybean plants traits

Based on the 3D point cloud of the soybean model, the LPM algorithm is proposed in this study to calculate plant height, leaf number, length and width, minimum bounding box volume of a single plant, minimum bounding box volume of a single leaf and leaf volume, projection area, projection length, and width. The extraction process of the trait parameters is shown in Figure 8. First, soybean plant point cloud is displayed, the height of soybean plant and minimum volume of bounding box per plant were measured. Then, the phenotypic parameters of leaves were extracted after segmentation. The specific parameters were calculated as follows:

2.6.1 Height of soybean plants

Plant height is an important indicator of plant growth in various environments (Xiao et al., 2020). The point clouds of individual soybean plants (Figure 8A) were extracted, and all points were traversed. After coordinate correction, the growth direction of the soybean was consistent with the z-axis direction. Therefore, the maximum value of the Z-axis coordinates between soybean and potted plants was selected, and the absolute value of the difference was the height of a single soybean plant (Figure 8B).

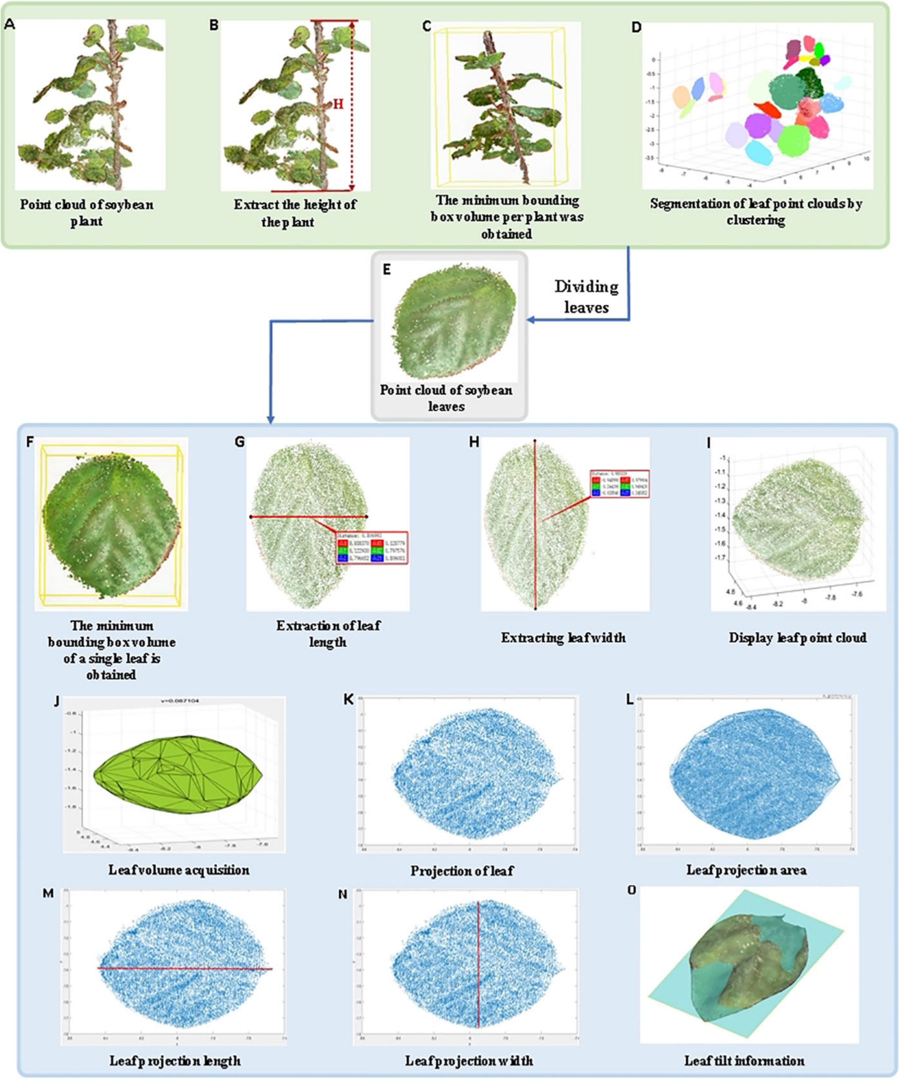

A
Point cloud of soybean plant
B
H
Extract the height of the plant
C
The minimum bounding box volume per plant was obtained
D
Segmentation of leaf point clouds by clustering
Dividing leaves
E
Point cloud of soybean leaves
F
The minimum bounding box volume of a single leaf is obtained
G
Extraction of leaf length
H
Extracting leaf width
I
Display leaf point cloud
J
Leaf volume acquisition
K
Projection of leaf
L
Leaf projection area
M
Leaf projection length
N
Leaf projection width
O
Leaf tilt information

FIGURE 8
Extraction of soybean plant trait parameters. **(A)** Point cloud of soybean plant; **(B)** Extract the height of the plant; **(C)** The minimum bounding box volume per plant was obtained; **(D)** Segmentation of leaf point clouds by clustering; **(E)** Point cloud of soybean leaves; **(F)** The minimum bounding box volume of a single leaf is obtained; **(G)** Extraction of leaf length; **(H)** Extracting leaf width; **(I)** Display leaf point cloud; **(J)** Leaf volume acquisition; **(K)** Projection of leaf; **(L)** Leaf projection area; **(M)** Leaf projection length; **(N)** Leaf projection width; **(O)** Leaf tilt information.

2.6.2 Minimum volume of bounding box per plant

The individual soybean plants were corrected to the main direction, and the cuboid composed of yellow lines was the bounding box. The maximum x, y, and z coordinate values and the minimum x, y, and z coordinate values of the point cloud of the individual soybean plant after correction were determined, and eight vertices were obtained. The cuboid volume formed by the connection of the eight vertices corresponds to the minimum bounding box volume of the individual plant (Figure 8C).

2.6.3 Number of soybean leaves

The non-stem point cloud was extracted to remove noise and external points, and the point cloud clustering algorithm was used to segment soybean leaves into different parts of a single plant (different colors represent different classes), where the number of different classes clustered was the number of leaves (Figure 8D).

2.6.4 Minimum bounding box volume of a single leaf

The individual soybean plants were corrected to the main direction, and any parts of the leaves were cut (Figure 8E). The cuboid, which is composed of yellow lines, is the bounding box. The maximum x, y, and z coordinates and the minimum x, y, and z coordinates of the point cloud of the corrected individual soybean plants were determined, and eight vertices were obtained. The volume of the cuboid formed by the connection of these eight vertices was the minimum bounding box volume of a single plant (Figure 8F).

2.6.5 Length of soybean leaves

The length of soybean leaves were calculated by the distance along the surface of the leaf, and any segmented leaf was extracted. The Euclidean distance algorithm was used to obtain the distance between the leaf base and leaf tip as the leaf length (Figure 8G).

2.6.6 Width of soybean leaves

The width of soybean leaves were calculated by the distance along the surface of the leaf, and any segmented leaf was extracted. The Euclidean distance algorithm was used to obtain the maximum distance perpendicular to the leaf length as the leaf width (Figure 8H).

2.6.7 Leaf volume of soybean

After extraction and segmentation, any soybean leaf is displayed (Figure 8I), and Gaussian filtering is used to de-noise the point cloud, and the envelope of its 3D point cloud is extracted. Each point cloud was divided into discrete grids, and the volume of the corresponding cell of each grid was calculated and summed to obtain the soybean leaf volume (Figure 8J).

2.6.8 Projected area of soybean leaves

The segmented arbitrary soybean leaves were projected onto the oxy-plane, and the corresponding projected leaf point cloud was generated (Figure 8K). The projected leaves were triangulated using a greedy projection algorithm (Zhang Y et al., 2019), and the projected soybean leaves after triangulation were composed of small triangles. The leaf projection area of a single leaf was calculated based on the Helen formula and area summation formula (Figure 8L). The formula used is given by

$$S_i = \sqrt{p_j(p_i - a_j)(p_j - b_j)(p_j - c_j)} \tag{9}$$

$$S_{2D} = \sum_{j=0}^{m} S_j \tag{10}$$

where, p_j is half of the perimeter of the triangulated triangle, a_j, b_j and c_j are the lengths of each side of the triangulated triangle, m is the total number of triangulated triangles, j is the index number of triangulated triangles, S_j is the projection area of a single planar triangulated facet, and S_{2D} is the total projection area of a single leaf.

2.6.9 Projection length of soybean leaves

The segmented soybean leaves were projected onto the oxy plane to generate the corresponding projected leaf point cloud, and the maximum and minimum values of the length-direction coordinates were calculated. The absolute value of the difference was the default length of the soybean leaf projections (Figure 8M).

2.6.10 Projection width of soybean leaves

The segmented soybean leaves were projected onto the oxy plane to generate the corresponding projected leaf point cloud, and the maximum and minimum values of the width-direction coordinates were calculated. The absolute value of the difference was the default width of the soybean leaf projections (Figure 8N).

2.6.11 Tilt information of leaves

The growth situation and environmental problems of soybeans can be determined based on the tilt information of soybean leaves. RANSAC plane fitting was used to obtain the plane, fitting variance RMSE, and tilt matrix, which can judge the tilt direction from a series of point cloud information using an iterative method (Figure 8O).

2.7 Modeling based on plant phenotype prediction

In this study, for three soybean varieties (C3, 47-6, W82) in R4 stage, because it is difficult to obtain the information of leaves and only a small data set is available, we used popular shallow neural networks such Support Vector Machine (SVM), Back Propagation Neural Network (BP) and Generalized Regression Neural Network (GRNN) to construct the model and select the optimal one.

Support Vector Machine (SVM) (Deng et al., 2019) is based on statistical theory and its learning model algorithm, which determines the optimal classification hyperplane in the high-dimensional feature space of data by solving optimization problems. The least-squares support vector machine (LS-SVM) overcomes the computational burden of its constrained optimization programming based on SVM to handle complex data classification more effectively.

Back Propagation Neural Network (BP) (Ju and Feng, 2019) neural network is a multi-layer feedforward network trained by an error backpropagation algorithm. The phenotypic data of plant leaves were used as the input of the BP neural network, and the output was the predicted value of the plant varieties.

Generalized Regression Neural Network (GRNN) (Dai et al., 2019) has strong nonlinear mapping ability and learning speed. In terms of classification and fitting, the GRNN model performed better when the accuracy of the plant phenotypic parameter data was poor.

Since model prediction was made based on leaf morphological traits and the light source maps the leaf vertically, the data of leaf length and width are highly similar to the data of leaf projection length and width. Therefore, Six experimental parameters (minimum bounding box volume of a single leaf, leaf volume, projection length of soybean leaves, projection width of soybean leaves, projected area of soybean leaves and leaf tilt information) are preferably selected. The input datatype for training (e.g., X is (447 x 6) array that records 6 traits of 447 leaves, Y is (447 x 1) array that records the cultivars of corresponding, use integer as labels) to construct the models of soybean sample variety prediction. For each prediction model, 80% samples are randomly selected as the training set and 20% samples are used as the test set to detect the prediction effect.

2.8 Accuracy evaluation

The soybean plant height, leaf length, and leaf width measured by the algorithm were compared with manual measurement values to evaluate the accuracy of the proposed method. The accuracy was measured using the mean absolute percentage error (MAPE), root mean square error (RMSE), and determination coefficient (R^2) to evaluate the accuracy of the SFM algorithm. Correlation coefficients of calibration (Rc)、Root mean square error of calibration (RMSEC)、Correlation coefficients of prediction (Rp) and Root mean square error of prediction (RMSEP) are often used for evaluating the accuracy of models.

Mean absolute percentage error (MAPE) (Chen et al., 2020) is often used to evaluate the prediction of performance, which intuitively reflects the difference between the real value and the predicted value, usually in the range up to 100%. Root mean square error (RMSE) (Hodson, 2022) is used to measure the deviation between the predicted value and true value, and is more sensitive to outliers in the data. Determination coefficient (R2) (Piepho, 2019) is an important statistic that reflects the goodness of fit of the model. The value ranges from 0 to 1, and closer to 1 means better; Correlation coefficients of calibration (Rc) (Wang et al., 2020) as the correlation coefficient of determination for calibration, commonly used to evaluate model results, and with the value closer to 1 being better; Root mean square error of calibration (RMSEC) (Hacisalihoglu et al., 2022) is often used as an evaluation of quantitative models; Correlation coefficients of prediction (Rp) (Wang et al., 2020) as the correlation coefficient of determination for the prediction set, with the value closer to 1 means better; Root mean square error of prediction (RMSEP) (Cominotte et al., 2020) is commonly used to verify the prediction error of the model internally or externally, and is the most critical parameter for evaluating the goodness of a model.

3 Results

3.1 Results and analysis of LPM algorithm

In this study, a total of 45 soybean samples from three soybean varieties (C3, 47-6, W82) in the R4 stage were used for 3D reconstruction using the SFM algorithm, and the plant height and leaf point clouds of soybean plants were automatically segmented, measured, and analyzed. In the 3D point cloud of the soybean plant, the plant trait parameters measured by the algorithm were proportionally converted, and the automatically measured plant height, leaf length, and leaf width were compared with the manually measured values. Figure 9 shows the results.

As shown in Figure 9A, R^2=0.9775, MAPE = 2.7013%, RMSE = 0.9997 cm, and the accuracy of the plant height measurement by the algorithm was 97.2987%. In addition, R^2=0.9785, MAPE = 1.4706%, and RMSE = 0.2357 cm, and the accuracy of the leaf length measurement was 98.5294%, as shown in Figure 9B. As shown in Figure 9C, R^2 = 0.9487, MAPE= 1.8669%, and RMSE = 0.2666 cm, and the accuracy of leaf width measurement by the algorithm was 98.1331%. According to Figure 9, the results show that the proposed method has high accuracy, and the algorithm measurements are in good agreement with human measurements.

3.2 Prediction results of plant varieties

In this study, three modeling methods, such as BP, SVM, and GRNN were used to establish soybean plant variety prediction models. Soybean leaf phenotypic parameters and the soybean plant variety were used as model inputs and the output, respectively. Among them, RMSEC is often used as an evaluation of quantitative models; RMSEP is often used to validate the prediction error of a model internally or externally; Rc as the correlation coefficient of determination for calibration; Rp is used as the correlation coefficient of determination of the prediction set. The modeling results based on the six leaf phenotypic parameters are listed in Table 1.

By modeling the leaf phenotypic parameters in Table 1 to predict the types of soybean plants, the GRNN model had the highest prediction accuracy. The training set Rc of soybean plants was 0.9744, and the prediction set Rp was 0.9211.

4 Discussion

Zareef et al. (2019) used Partial Least Squares Regression (PLSR) based on the phenolic compounds of Congo black tea to predict and construct the model. The prediction accuracy of

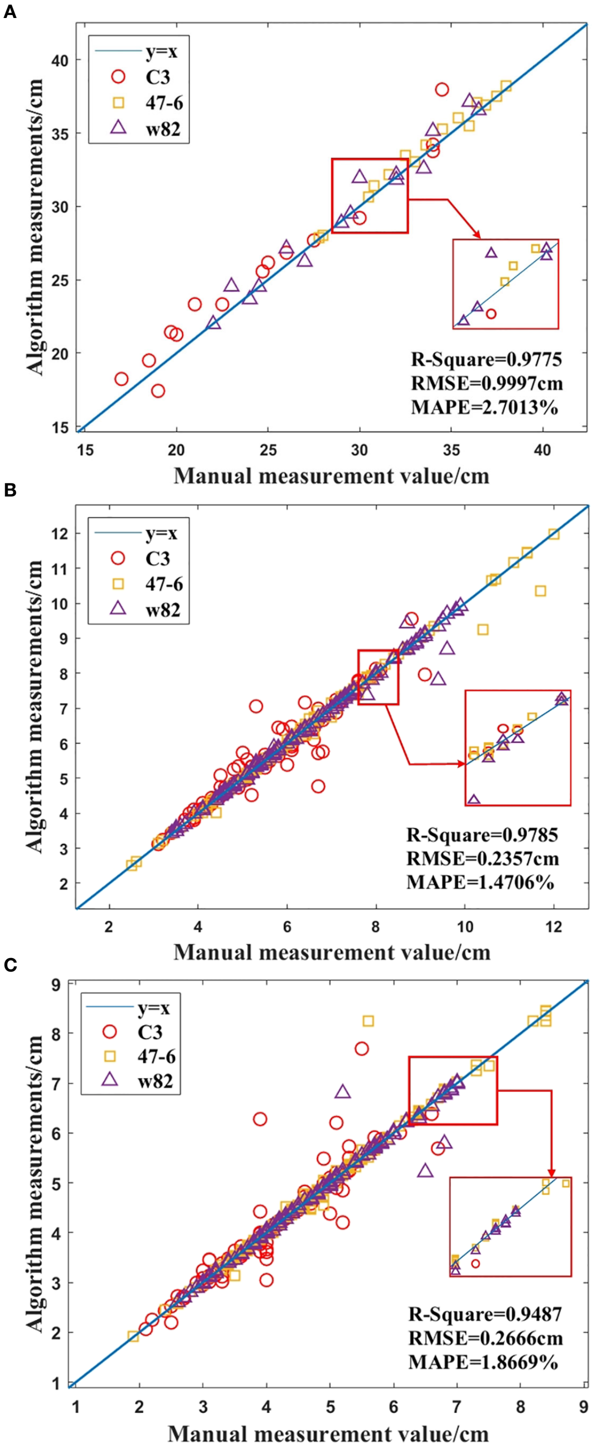


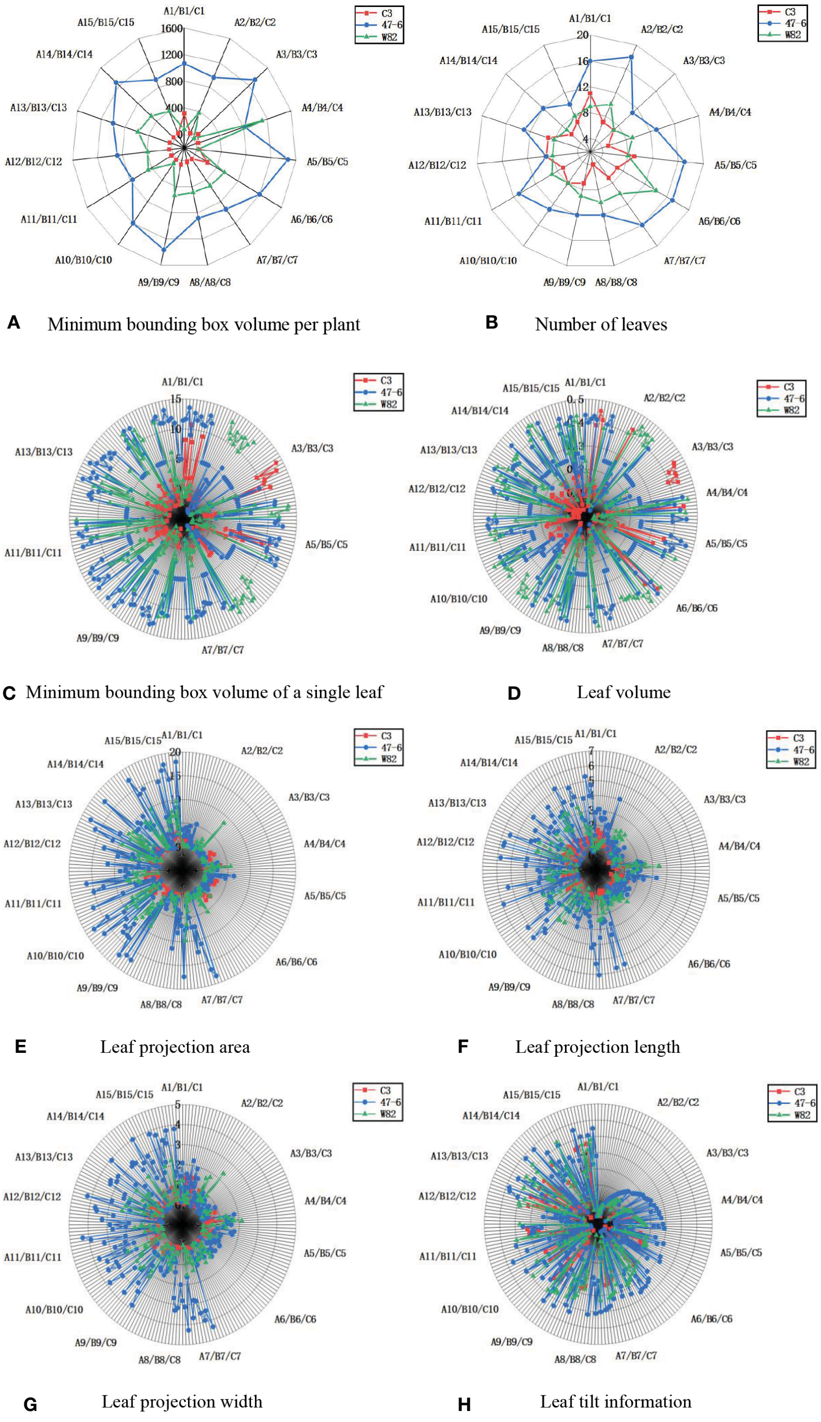
FIGURE 9
Comparison of manual and algorithmic measurements of soybean plant traits, **(A)** Height of the plant, **(B)** Length of the leaf, **(C)** Width of the leaf.

Gallic acid was 0.9111, and the prediction accuracy of Rutin was 0.8255; Hasan et al. (2021) applied six commonly ML methods (SVM, Adaboost, Logistic Regression, etc.), the gene models of Roaceae, rice and Arabidopsis were predicted and constructed, and the prediction accuracy was 0.918,0.827,0.635, respectively; Yoosefzadeh-Najafabadi et al. (2021) took advantage of three common ML (MLP, SVM, RF) based on hyperspectral reflectance data to predict and construct a soybean seed yield model, and the accuracy of the model was 0.87. The above methods use multiple models to classify and predict the phenotypes and compounds of multiple experimental objects quickly and efficiently, but the accuracy is relatively low.

The LPM algorithm used in this paper is combined with GRNN to construct a soybean prediction model, and the

TABLE 1 Modeling results of leaf phenotypic parameters.

Model	Rc	RMSEC	Rp	RMSEP
LS-SVM	0.6934	0.5979	0.6536	0.6995
BPNN	0.7781	0.6419	0.5716	0.9528
GRNN	0.9744	18.3263	0.9211	18.9024



A Minimum bounding box volume per plant

B Number of leaves

C Minimum bounding box volume of a single leaf

D Leaf volume

E Leaf projection area

F Leaf projection length

G Leaf projection width

H Leaf tilt information

FIGURE 10
Measurement results of soybean plant trait parameters, **(A)** Minimum bounding box volume per plant, **(B)** Number of leaves, **(C)** Minimum bounding box volume of a single leaf, **(D)** Leaf volume, **(E)** Leaf projection area, **(F)** Leaf projection length, **(G)** Leaf projection width, **(H)** Leaf tilt information.

accuracy of model can reach 0.9211. In the paper, the 3D model of soybean plant can be reconstructed quickly and accurately by using motion restoration structure algorithm and multi-view stereo vision algorithm; The LPM algorithm can effectively measure the phenotypic parameters of 11 plant three-dimensional models, and constructed the relationship between phenotype and insect resistance; The optimal model GRNN was established to accurately predict and identify plant varieties based on the morphological traits of leaves.

In terms of individual plant character parameters (minimum bounding box volume per plant, number of leaves, minimum bounding box volume per leaf, leaf volume, leaf projection area, leaf projection width, leaf projection length, and leaf tilt information), the soybeans of the C3 variety were lower than that of the 47-6 and W82 varieties, as shown in Figure 10. Soybean plant variety 47-6 were higher than soybean of variety W82 in terms of four trait parameters (minimum enclosing box volume per plant, number of leaves, leaf projected width, and leaf projected area). Soybean of varieties 47-6 and W82 were higher than soybean of variety W82 in four trait parameters (minimum enclosing box volume per plant, number of leaves, minimum enclosing box volume per leaf, and leaf projection area). There were no highly significant differences between the 47-6 and W82 varieties in terms of four trait parameters (leaf projection length, leaf volume, leaf projection width, and leaf tilt information).

C3, 47-6, and W82 are different gene expression forms of the same variety, where 47-6 (oe-Williams82) is a certain gene overexpression strain and C3 (ko-Williams82) is a gene knockout strain. Differences in gene expression may be the reason for the changes in the overall parameters, and the differences in gene expression will lead to changes in the surface hairs of the soybean. These hairs of soybean pods of the 47-6 overexpressed variety were sparse, and the pods were easily fed on by stink bugs. The stink bugs bite the soybean pods through the mouth, resulting in the normal development of soybean seeds (Chen et al., 2018) and the formation of aborted seeds. Here, the sink and source relationship is confusing. Therefore, the plant will use more nutrients to promote the vegetative growth and growth of its node, make the plant taller, and increase the volume of the minimum bounding box per plant and the number of leaves. However, pod feeding of *M. obstatus* did not affect changes in leaf morphology-related information, such as leaf projection length, leaf volume, leaf projection width, and leaf tilt information. C3 is an insect-resistant line, which is considerably slightly damaged by the bug. Thus, the trait parameters of C3 are significantly less than 47-6, and gene knockout affects the changes in leaf morphology-related information parameters. Plant phenotypic traits can be divided into physiological, morphological, and component traits (Danilevicz et al., 2022). Among the three major targets of breeding, such as the yield, quality, and resistance, the resistance target (biotic stress or abiotic stress) is particularly important and indicates the core productivity to ensure stable yield. Among them, changes in morphological and structural traits, such as plant height and leaf area, are the most intuitive reflections of plant resistance and they play an important role in the study of insect resistance (Nelson et al., 2018).

5 Conclusion

The soybean plant 3D structure was successfully obtained by SfM, and a good correction (R2>0.94) and small RMSE (<0.24) were observed with manual measured. Compared to SVM and BPNN, the GRNN showed the highest accuracy (0.9211) of the cultivar classification tasks.

In this paper, we mainly focus on the 3D reconstruction of soybean plants (ko-Williams82, oe-Williams82, and Williams82), and analyze the relationship between phenotypic traits and insect resistance genes. In the later stage, a whole set of machines will be developed to expand the number of soybean varieties and monitor the growth changes of soybean plants in real-time to further enhance the practicability and realize more comparisons of soybeans between species and genotypes to select superior insect-resistant varieties.

Data availability statement

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

Author contributions

WH proposed the conceptualization and methodology, and wrote the paper. ZY programmed the software. ML compared the performance of the algorithms. YY designed and carried out the experiments. WL and GX improved the methodology and conceived the experiments. All authors reviewed the manuscript. All authors contributed to the article and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation of China (32071896), the National Key R&D Program of China (2021YFD1201604), and Jiangsu JCICMCP Program.

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

Alam, M., Alam, M. S., Roman, M., Tufail, M., Khan, M. U., Khan, M. T., et al. (2020). "Real-time machine-learning based crop/weed detection and classification for variable-rate spraying in precision agriculture," in *2020 7th International Conference on Electrical and Electronics Engineering (ICEEE).* (IEEE), 273–280.

Alom, M. Z., Yakopcic, C., Nasrin, M., Tarek Taha, M. T.M., and Vijayan, K. (2019). Breast cancer classification from histopathological images with inception recurrent residual convolutional neural network. *J. digital Imaging* 32 (4), 605–617. doi: 10.1007/s10278-019-00182-7

Alonge, M., Wang, X., Benoit, M., Soyk, S., Pereira, L., Zhang, L., et al. (2020). Major impacts of widespread structural variation on gene expression and crop improvement in tomato. *Cell* 182 (1), 145–161. e23. doi: 10.1016/j.cell.2020.05.021

Ao, Z., Wu, F., Hu, S., Sun, Y., Su, Y., Guo, Q., et al. (2022). Automatic segmentation of stem and leaf components and individual maize plants in field terrestrial LiDAR data using convolutional neural networks. *Crop J.* 10 (5), 1239–1250. doi: 10.1016/j.cj.2021.10.010

Azimi, S., Kaur, T., and Gandhi, T. K. (2021). A deep learning approach to measure stress level in plants due to Nitrogen deficiency. *Measurement* 173, 108650. doi: 10.1016/j.measurement.2020.108650

Barker, H. L., Holeski, L. M., and Lindroth, R. L. (2019). Independent and interactive effects of plant genotype and environment on plant traits and insect herbivore performance: a meta-analysis with Salicaceae. *Funct. Ecol.* 33 (3), 422–435. doi: 10.1111/1365-2435.13249

Barradas, A., Correia, P. M. P., Silva, S., Mariano, P., Pires, M. C., Matos, A. R., et al. (2021). Comparing machine learning methods for classifying plant drought stress from leaf reflectance spectra in Arabidopsis thaliana. *Appl. Sci.* 11 (14), 6392. doi: 10.3390/app11146392

Brugger, A. (2022). *Deep Phenotyping of disease resistance based on hyperspectral imaging and data mining methods in high throughput* (Universitäts-und Landesbibliothek Bonn).

Cardellicchio, A., Solimani, F., Dimauro, G., Petrozza, A., Summerer, S., Cellini, F., et al. (2023). Detection of tomato plant phenotyping traits using YOLOv5-based single stage detectors. *Comput. Electron. Agric.* 207, 107757. doi: 10.1016/j.compag.2023.107757

Chawade, A., van Ham, J., Blomquist, H., Bagge, O., Alexandersson, E., and Ortiz, R. (2019). High-throughput field-phenotyping tools for plant breeding and precision agriculture. *Agronomy* 9 (5), 258. doi: 10.3390/agronomy9050258

Chen, J. H., Bi, R., Huang, J.-M., Cui, J., and Shi, S.-S. (2018). Differential analysis of the effects of different bugs on soybean growth and yield. *Soybean Sci.* 37 (04), 585–589. doi: 10.11861/j.issn.1000-9841.2018.04.0585

Chen, X., Xu, X., Yang, Y., Wu, H., Tang, J., and Zhao, J. (2020). Augmented ship tracking under occlusion conditions from maritime surveillance videos. *IEEE Access* 8, 42884–42897. doi: 10.1109/ACCESS.2020.2978054

Chen, Y., Zhang, M., and Bhandari, B. (2021). 3D printing of steak-like foods based on textured soybean protein. *Foods* 10 (9), 2011. doi: 10.3390/foods10092011

Chlingaryan, A., Sukkarieh, S., and Whelan, B. (2018). Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. *Comput. Electron. Agric.* 151, 61–69. doi: 10.1016/j.compag.2018.05.012

Cominotte, A., Fernandes, A. F. A., Dorea, J. R., Rosa, G. J.M., Ladeira, M. M., van Cleef, E. H. C. B., et al. (2020). Automated computer vision system to predict body weight and average daily gain in beef cattle during growing and finishing phases. *Livestock Sci.* 232, 103904. doi: 10.1016/j.livsci.2019.103904

Cong, I., Choi, S., and Lukin, M. D. (2019). Quantum convolutional neural networks. *Nat. Phys.* 15 (12), 1273–1278. doi: 10.1038/s41567-019-0648-8

Dai, Y., Guo, J., Yang, L., and You, W. (2019). A new approach of intelligent physical health evaluation based on GRNN and BPNN by using a wearable smart bracelet system. *Proc. Comput. Sci.* 147, 519–527. doi: 10.1016/j.procs.2019.01.235

Danilevicz, M. F., Gill, M., Anderson, R., Batley, J., Bennamoun, M., Bayer, P. E., et al. (2022). Plant genotype to phenotype prediction using machine learning. *Front. Genet.* 13. doi: 10.3389/fgene.2022.822173

Das Choudhury, S., Maturu, S., Samal, A., Stoerger, V., and Awada, T. (2020). Leveraging image analysis to compute 3D plant phenotypes based on voxel-grid plant reconstruction. *Front. Plant Sci.* 11, 521431. doi: 10.3389/fpls.2020.521431

Deng, W., Yao, R., Zhao, H., Yang, X., and Li, G. (2019). A novel intelligent diagnosis method using optimal LS-SVM with improved PSO algorithm. *Soft Comput.* 23 (7), 2445–2462. doi: 10.1007/s00500-017-2940-9

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. (2019). "Gradient descent finds global minima of deep neural networks," in *International conference on machine learning.* (PMLR), 1675–1685.

Ewertowski, M. W., Tomczyk, A. M., Evans, D. J. A., Roberts, D. H., and Ewertowski, W. (2019). Operational framework for rapid, very-high resolution mapping of glacial geomorphology using low-cost unmanned aerial vehicles and structure-from-motion approach. *Remote Sens.* 11 (1), 65. doi: 10.3390/rs11010065

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis.Comput. *Electron. Agric.* 145, 311–318. doi: 10.1016/j.compag.2018.01.009

Gage, J. L., Richards, E., Lepak, N., Kaczmar, N., Soman, C., Chowdhary, G., et al. (2019). In-field whole-plant maize architecture characterized by subcanopy rovers and latent space phenotyping. *Plant Phenome J.* 2 (1), 1–11. doi: 10.2135/tppj2019.07.0011

Hacisalihoglu, G., Armstrong, P. R., Mendoza, P. T. D., and Seabourn, B. W. (2022). Compositional analysis in sorghum (Sorghum bicolor) NIR spectral techniques based on mean spectra from single seeds. *Front. Plant Sci.* 13, 995328. doi: 10.3389/fpls.2022.995328

Han, X. F., Jin, J. S., Wang, M.-J., Jiang, W., Gao, L., and Xiao, L. (2017). A review of algorithms for filtering the 3D point cloud. *Signal Process.: Image Commun.* 57, 103–112. doi: 10.1016/j.image.2017.05.009

Hasan, M. M., Basith, S., Khatun, M. S., Lee, G., Manavalan, B., and Kurata, H. (2021). Meta-i6mA: an interspecies predictor for identifying DNA N 6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework. *Briefings Bioinf.* 22 (3), bbaa202. doi: 10.1093/bib/bbaa202

He, J. Q., Harrison, R. J., and Li, B. (2017). A novel 3D imaging system for strawberry phenotyping. *Plant Methods* 13 (1), 1–8. doi: 10.1186/s13007-017-0243-x

Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscie. Model. Dev.* 15 (14), 5481–5487. doi: 10.5194/gmd-15-5481-2022

Huang, Y., Liu, Y., Han, T., Xu, S., and Fu, J. (2022). Low illumination soybean plant reconstruction and trait perception. *Agriculture* 12 (12), 2067. doi: 10.3390/agriculture12122067

Hui, F., Zhu, J., Hu, P., Meng, L., Zhu, B., Guo, Y., et al. (2018). Image-based dynamic quantification and high-accuracy 3D evaluation of canopy structure of plant populations. *Ann. Bot.* 121 (5), 1079–1088. doi: 10.1093/aob/mcy016

James, M. R., Chandler, J. H., Eltner, A., Fraser, C., Miller, P. E., Mills, J. P., et al. (2019). Guidelines on the use of structure-from-motion photogrammetry in geomorphic research. *Earth Surface Processes Landforms* 44 (10), 2081–2084. doi: 10.1002/esp.4637

Jiang, S., Jiang, C., and Jiang, W. (2020). Efficient structure from motion for large-scale UAV images: A review and a comparison of SfM tools. *ISPRS J. Photogrammetry Remote Sens.* 167, 230–251. doi: 10.1016/j.isprsjprs.2020.04.016

Ju, X., and Feng, Y. (2019). Ultrasonic scanning image segmentation based on BP neural network. *Modern Ind. Econ. Inf.* 9 (8), 23–24. doi: 10.16525/j.cnki.14-1362/n.2019.08.09

Junttila, S., Hölttä, T., Lindfors, L., Issaoui, A. E., Vastaranta, M., Hyyppä, H., et al. (2021). Why trees sleep?-explanations to diurnal branch movement. doi: 10.21203/rs.3.rs-365866/v1

Kamilaris, A., and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Comput. Electron. Agric.* 147, 70–90. doi: 10.1016/j.compag.2018.02.016

Kuett, L., Catena, R., Özcan, A., Plüss, A.Cancer Grand Challenges IMAXT Consortium, , Schram, P., et al. (2022). Three-dimensional imaging mass cytometry for highly multiplexed molecular and cellular mapping of tissues and the tumor microenvironment. *Nat. Cancer* 3 (1), 122–133. doi: 10.1038/s43018-021-00301-w

Li, Q., and Cheng, X. (2018). Comparison of different feature sets for TLS point cloud classification. *Sensors* 18 (12), 4206–4422. doi: 10.3390/s18124206

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

Li, D., Quan, C., Song, Z., Li, X., Yu, G., Li, C., et al. (2021). High-throughput plant phenotyping platform (HT3P) as a novel tool for estimating agronomic traits from the lab to the field. *Front. Bioeng. Biotechnol.* 8, 623705. doi: 10.3389/fbioe.2020.623705

Liang, X. Y., Zhou, F. G., Chen, H., Liang, B., Xu, X. , et al. (2020). Three-dimensional maize plants reconstruction and traits extraction based on structure from motion. *Trans. Chin. Soc. Agric. Machinery* 51 (06), 209–219. doi: 10.6041/j.issn.1000-1298.2020.06.022

Ma, X., Wei, B., Guan, H., and Yu, S. (2022). A method of calculating phenotypic traits for soybean canopies based on three-dimensional point cloud. *Ecol. Inf.* 68, 101524. doi: 10.1016/j.ecoinf.2021.101524

Martinez-Guanter, J., Ribeiro, Á, Peteinatos, G., Pérez-Ruiz, M., Gerhards, R., Bengochea-Guevara, J. M., et al. (2019). Low-cost three-dimensional modeling of crop plants. *Sensors* 19 (13), 2883. doi: 10.3390/s19132883

Nelson, R., Wiesner-Hanks, T., Wisser, R., and Balint-Kurti, P. (2018). Navigating complexity to breed disease-resistant crops. *Nat. Rev. Genet.* 19 (1), 21–33. doi: 10.1038/nrg.2017.82

Nguyen, V. T., Fournier, R. A., Côté, J. F., and Pimont, F. (2022). Estimation of vertical plant area density from single return terrestrial laser scanning point clouds acquired in forest environments. *Remote Sens. Environ.* 279, 113115. doi: 10.1016/j.rse.2022.113115

Omari, M. K., Lee, J., Faqeerzada, M. A., Park, E., and Cho, B.-K. (2020). Digital image-based plant phenotyping: a review. *Korean J. Agric. Sci.* 47 (1), 119–130. doi: 10.7744/kjoas.20200004

Piepho, H. P. A. (2019). coefficient of determination (R2) for generalized linear mixed models. *Biometrical J.* 61 (4), 860–872. doi: 10.1002/bimj.201800270

Piermattei, L., Karel, W., Wang, D., Wieser, M., Mokroš, M., Surový, P., et al. (2019). Terrestrial structure from motion photogrammetry for deriving forest inventory data. *Remote Sens.* 11 (8), 950. doi: 10.3390/rs11080950

Rahman, A., Mo, C., and Cho, B. K. (2017). 3-D image reconstruction techniques for plant and animal morphological analysis - A review. *J. Biosyst. Eng.* 42 (4), 339–349. doi: 10.5307/JBE.2017.42.4.339

Rannik, Ü, Vesala, T., Peltola, O., Novick, K. A., Aurela, M., Järvi, L., et al. (2020). Impact of coordinate rotation on eddy covariance fluxes at complex sites. *Agric. For. Meteorol.* 287, 107940. doi: 10.1016/j.agrformet.2020.107940

Swinfield, T., Lindsell, J. A., Williams, J. V., Harrison, R. D., Habibi, A., Gemita, E., et al. (2019). Accurate measurement of tropical forest canopy heights and aboveground carbon using structure from motion. *Remote Sens.* 11 (8), 928. doi: 10.3390/rs11080928

Tan, L., Lu, J., and Jiang, H. (2021). Tomato leaf diseases classification based on leaf images: a comparison between classical machine learning and deep learning methods. *AgriEngineering* 3 (3), 542–558. doi: 10.3390/agriengineering3030035

Tyagi, S., Kesiraju, K., Saakre, M., Rathinam, M., Raman, V., Pattanayak, D., et al. (2020). Genome editing for resistance to insect pests: an emerging tool for crop improvement. *ACS omega* 5 (33), 20674–20683. doi: 10.1021/acsomega.0c01435

Van Eeuwijk, F. A., Bustos-Korts, D., Millet, E. J., Boer, M. P., Kruijer, W., Thompson, A., et al. (2019). Modelling strategies for assessing and increasing the effectiveness of new phenotyping techniques in plant breeding. *Plant Sci.* 282, 23–39. doi: 10.1016/j.plantsci.2018.06.018

Vogt, G. (2021). Epigenetic variation in animal populations: sources, extent, phenotypic implications, and ecological and evolutionary relevance. *J. Biosci.* 46 (1), 1–47. doi: 10.1007/s12038-021-00138-6

Wang, Y. J., Jin, G., Li, L. Q., Liu, Y., Kalkhajeh, Y. K., Ning, J.-M., et al. (2020). NIR hyperspectral imaging coupled with chemometrics for nondestructive assessment of phosphorus and potassium contents in tea leaves. *Infrared Phys. Technol.* 108, 103365. doi: 10.1016/j.infrared.2020.103365

Wang, F., Ma, X., Liu, M., and Wei, B. (2022). Three-dimensional reconstruction of soybean canopy based on multivision technology for calculation of phenotypic traits. *Agronomy* 12 (3), 692. doi: 10.3390/agronomy12030692

Xiao, S., Chai, H., Shao, K., Shen, M., Wang, Q., Wang, R., et al. (2020). Image-based dynamic quantification of aboveground structure of sugar beet in field. *Remote Sens.* 12 (2), 269. doi: 10.3390/rs12020269

Xu, R., Li, C., and Paterson, A. H. (2019). Multispectral imaging and unmanned aerial systems for cotton plant phenotyping. *PloS One* 14 (2), e0205083. doi: 10.1371/journal.pone.0205083

Xue, C., Qiu, F., Wang, Y., Li, B., Zhao, K. T., Chen, K., et al. (2023). Tuning plant phenotypes by precise, graded downregulation of gene expression. *Nat. Biotechnol.*, 1–7. doi: 10.1038/s41587-023-01707-w

Yoosefzadeh-Najafabadi, M., Earl, H. J., Tulpan, D., Sulik, J., and Eskandari, M. (2021). Application of machine learning algorithms in plant breeding: predicting yield from hyperspectral reflectance in soybean. *Front. Plant Sci.* 11, 624273. doi: 10.3389/fpls.2020.624273

Yu, Y., Si, X., Hu, C., and Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* 31 (7), 1235–1270. doi: 10.1162/neco_a_01199

Zareef, M., Chen, Q., Ouyang, Q., Arslan, M., Hassan, M., Ahmad, W., et al. (2019). Rapid screening of phenolic compounds in congou black tea (Camellia sinensis) during in *vitro* fermentation process using portable spectral analytical system coupled chemometrics. *J. Food Process. Preservation* 43 (7), e13996. doi: 10.1111/jfpp.13996

Zhang, T., Wu, T., Wang, L., Jiang, B., Zhen, C., Yuan, S., et al. (2019). A combined linkage and GWAS analysis identifies QTLs linked to soybean seed protein and oil content. *Int. J. Mol. Sci.* 20 (23), 5915. doi: 10.3390/ijms20235915

Zhang, Y., Wu, H., and Yang, W. (2019). Forests growth monitoring based on tree canopy 3D reconstruction using UAV aerial photogrammetry. *Forests* 10 (12), 1052. doi: 10.3390/f10121052

Zhou, S., Chai, X., Yang, Z., Wang, H., Yang, C., and Sun, T. (2021). Maize-IAS: a maize image analysis software using deep learning for high-throughput plant phenotyping. *Plant Methods* 17 (1), 48. doi: 10.1186/s13007-021-00747-0

Zhu, R., Sun, K., Yan, Z., Yan, X., Yu, J., Shi, J., et al. (2020). Analysing the phenotype development of soybean plants using low-cost 3D reconstruction. *Sci. Rep.* 10 (1), 1–17. doi: 10.1038/s41598-020-63720-2

TYPE Methods
PUBLISHED 25 August 2023
DOI 10.3389/fpls.2023.1170221

Development of effective model for non-destructive detection of defective kiwifruit based on graded lines

Feiyun Wang[†], Chengxu Lv, Lizhong Dong, Xilong Li, Pengfei Guo and Bo Zhao*[†]

National Key Laboratory of Agricultural Equipment Technology, Chinese Academy of Agricultural Mechanization Sciences Group Co., Ltd, Beijing, China

The accurate detection of external defects in kiwifruit is an important part of postharvest quality assessment. Previous studies have not considered the problems posed by the actual grading environment. In this study, we designed a novel approach based on improved Yolov5 to achieve real-time and efficient non-destructive detection of multiple defect categories in kiwifruit. First, a kiwifruit image acquisition device based on grading lines was developed to enhance the image acquisition. Subsequently, a kiwifruit dataset was constructed based on the external defect characteristics and a new data enhancement method was proposed to augment the kiwifruit samples. Thereafter, the SPD-Conv and DW-Conv modules were combined to improve Yolov5s, with EIOU as the loss calculation function. The results demonstrated that the improved model training loss value was 0.013 lower, the convergence was accelerated, the number of parameters was reduced, and the computational effort was increased. The detection accuracies of the samples in the test set, which included healthy, leaf-rubbing damaged, healed cuts or scarred, and sunburned samples, were 98.8%, 98.7%, 97.6%, and 95.9%, respectively, with an overall detection accuracy of 97.7%. The detection time was 8.0 ms, thereby meeting real-time sorting demands. The average detection accuracy and model size of SSD, Yolov5s, Yolov7, and Yolov5-Ours were compared. When the confidence threshold was 0.5, the detection accuracy of Yolov5-Ours was 10% and 6.4% higher than that of SSD and Yolov5s, respectively. In terms of the model size, Yolov5-Ours was approximately 6.5- and 4-fold smaller than SSD and Yolov7, respectively. Thus, Yolov5-Ours achieved the highest accuracy, adaptability, and robustness for the detection of all kiwifruit categories as well as a small volume and portability. These results can provide technical support for the non-destructive detection and grading of agricultural products in the future.

KEYWORDS

kiwifruit, grading line, SPD-Conv, DWConv, real time, non-destructive detection

Check for updates

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU),
United States

REVIEWED BY
Ruoyu Zhang,
Shihezi University, China
Naveen Kumar Mahanti,
Dr.Y.S.R. Horticultural University, India

*CORRESPONDENCE
Bo Zhao
✉ zhaoboshi@126.com

[†]These authors have contributed equally to this work

RECEIVED 20 February 2023
ACCEPTED 01 August 2023
PUBLISHED 25 August 2023

CITATION
Wang F, Lv C, Dong L, Li X, Guo P and Zhao B (2023) Development of effective model for non-destructive detection of defective kiwifruit based on graded lines. *Front. Plant Sci.* 14:1170221. doi: 10.3389/fpls.2023.1170221

1 Introduction

Kiwifruit is characterized by a soft texture, sweet and sour taste, and richness in amino acids and minerals. The detection and grading of kiwifruit are key aspects of postharvest processing and provide important support for value-added commercialization (Fu et al., 2018; Li et al., 2022).

In China, the grading of kiwifruits from different cities is primarily conducted by manual sorting at present, which is inefficient and subjective. Existing sorting equipment, such as mechanical size grading and weight grading, cannot identify the external defects of the fruit. Thus, computer vision is being applied increasingly to agricultural products with the developments in image processing technology (Liu et al., 2020; Tian et al., 2021).

Traditional image processing methods usually achieve fruit recognition and detection by combining the extraction of shallow information, such as the color, size, and texture of the target, using techniques such as segmentation and discriminative models. Cui et al. (2012) proposed the use of a near-infrared light source for image acquisition and realized the extraction of scratch, decay, and sun-burning defects using segmentation. Yang et al. (2021) used the K-means clustering algorithm to segment the surface of kiwifruit and reject defective fruits according to the darker color of surface defects, such as fruit scars and disease spots, compared with those of normal fruits. Subsequent studies (Zhou et al., 2012; Liu and Gai, 2020) used an image segmentation algorithm to extract the contours of the fruit in an image to meet the detection and grading needs. Li et al. (2020) used hyperspectral techniques for deformed kiwifruit detection and compared three methods: the partial least-squares linear discriminant model, back-propagation neural network (BPNN), and least-squares support vector machine. The experimental results showed that the BPNN model achieved the highest accuracy at 97.56%. Fu et al. (2016) used a camera with a weight sensor on a grading line that was equipped for kiwifruit shape grading through a stepwise multiple linear regression method. The grading accuracy when using a linear combination of the cross-sectional diameter length was 98.3%. However, traditional image processing techniques, which generally extract feature targets manually, are only applicable to specific scene studies, have weaker robustness, and are susceptible to environmental influences during the extraction process.

Deep convolutional neural networks (CNNs) are superior to traditional methods and have been applied to the class classification and defect detection of fruits. Fan et al. (2020) improved the parameters and number of connections in a CNN model to detect the surface defects of apples in real time, with an accuracy of 92%. Lu et al. (2022) used the Attention-YOLOv4 model to detect the ripeness of different-colored apples. Zhang et al. (2020) improved the VGG16 model by converting it into a fully convolutional network and combining it with a spectral projection image to segment the mechanical damage and calyx regions of blueberries. Their method achieved an accuracy of 81.2%. Similarly, Wang et al. (2018) combined hyperspectral images with deep learning methods, and used the AlexNet and ResNet models to detect internal mechanical damage in blueberries. Their results showed that the deep learning models could maintain a higher accuracy than that of machine learning methods while reducing the calculation time significantly. Yu et al. (2018) proposed a combined model consisting of an autoencoder and a fully connected neural network to predict the hardness and soluble solid contents of Korla fragrant pears, resulting in a correlation coefficient of 0.89. Momeny et al. (2020) combined maximum pooling with mean pooling in a CNN to classify self-built regular and irregular cherry databases with an accuracy of 99.4%. Luna et al. (2019) created a dataset of healthy and defective tomatoes and evaluated the accuracy of their model using VGG16. A high accuracy rate of 98.75% was achieved. Azizah et al. (2017) used a four-fold cross-validation method to classify CNN mangosteen with an accuracy of 97.5%. Jahanbakhshi et al. (2020) proposed an improved CNN model for healthy and damaged sour lemon detection, achieving an accuracy of 100%. Xue et al. (2018) improved the YOLOv2 model using the Tiny-yolo-dense network to detect unripe mangoes with an accuracy of 97.02%. CNNs have achieved high detection accuracy, application flexibility, and good performance rates in many fruit quality detection studies. However, the detection of small objects with a low resolution remains challenging. This is because small objects with a low resolution provide few learning features and often coexist with larger undetectable objects.

Therefore, in this study, a kiwifruit dataset was constructed according to an image acquisition device based on grading lines for the detection of external kiwifruit defects. The widely used Yolov5s (Li et al., 2023) was selected as the base model. The network structure was improved and the loss function was optimized to achieve non-destructive and efficient external detection of kiwifruit. The results of this study can provide technical support for kiwifruit quality grading.

2 Materials and methods

2.1 Dataset production

2.1.1 Sample source

Kiwifruit samples were obtained from the Zhouzhi (108.20 °E, 34.17 °N) and Meixian counties (107.76 °E, 34.29 °N) in Shaanxi Province. The kiwifruit varieties Xu Xiang and Cui Xiang were selected as the subjects of the study, and multiple batches were acquired in the field and online from November 2021 to November 2022. A total of 1,020 original samples were obtained, including 320 healthy samples, 240 leaf-rubbing damaged samples, 240 sunburned samples, and 220 healed cuts or scarred samples. The various sample types are presented in Figure 1.

2.1.2 Image acquisition

Image acquisition was performed using an MV-EM200C camera (Microvision, Xi'an, China) with a model BT-23C0814MP5 industrial lens, an image resolution of 1,600 × 1,200 pixels, and an acquisition frame rate of 39.93 fps. The image acquisition device was constructed based on a grading line (Li et al., 2018), as illustrated in Figure 2, and mainly included the

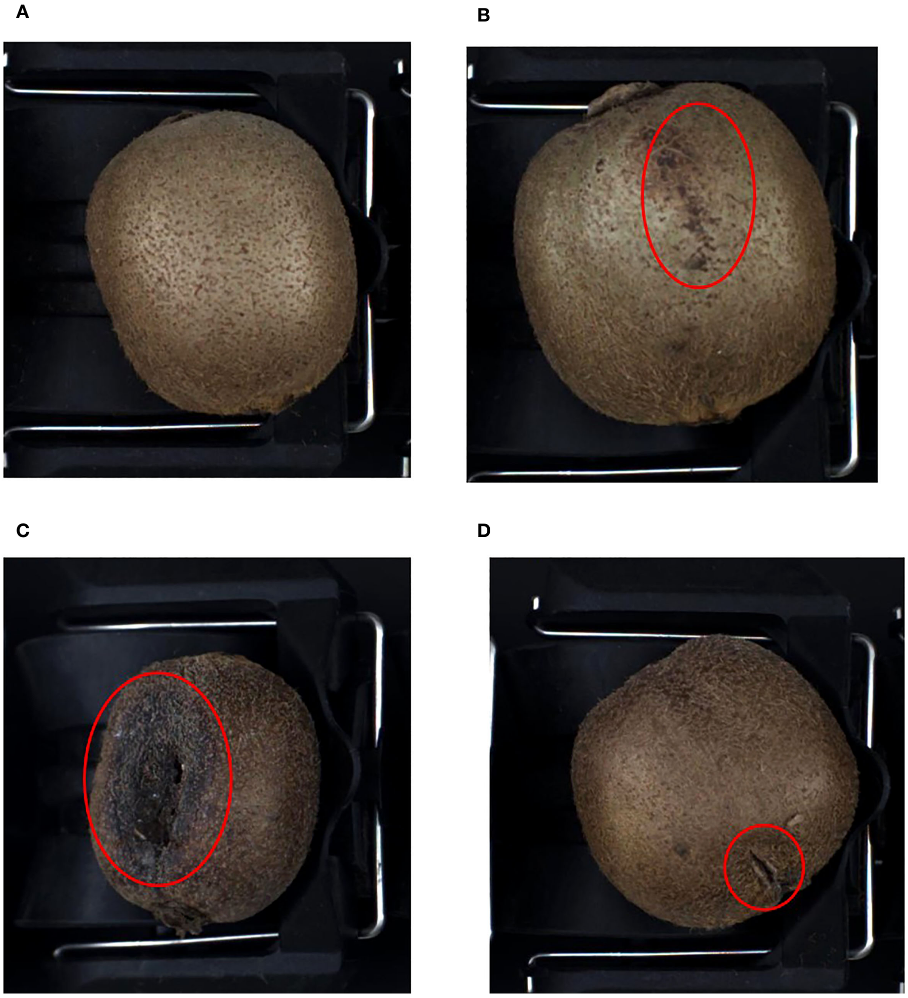


FIGURE 1
Kiwifruit samples. **(A)** Healthy, **(B)** leaf-rubbing damaged, **(C)** sunburned, **(D)** healed cuts or scarred.

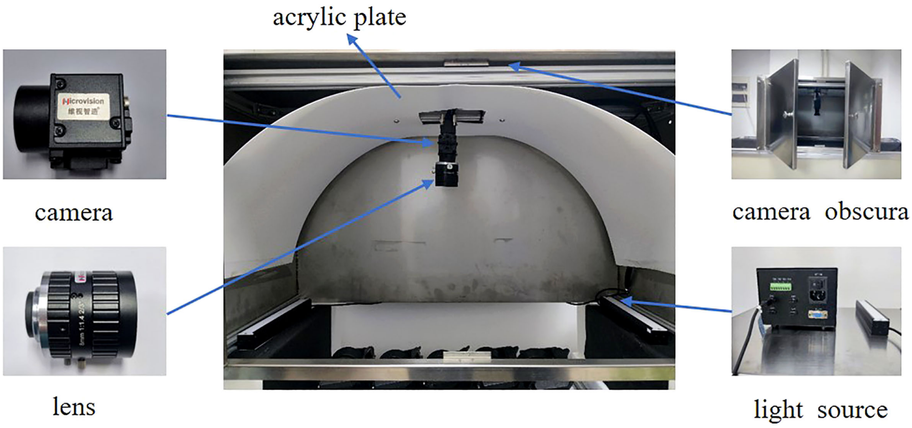


FIGURE 2
Acquisition device diagram.

camera, lens, camera obscura, light sources, and acrylic plate. The camera height was adjusted to 32 cm above the tray level to capture the information of the three trays completely in a single image for the grading application scenario. When the grading line moved, the roller tray could turn the kiwifruit, and three samples in a single image could be obtained to acquire the full surface information of the kiwifruit. The light source was emitted from the bottom and reflected on the kiwifruit surface through a half-cylinder acrylic plate, which helped to reduce the problems of uneven light exposure and reflection at different locations owing to direct radiation. When the graded line speed was adjusted to 3–5 pcs/sec, the pallet information was captured by a counter-light sensor, which was passed to the isolation plate, thereby driving the camera to trigger synchronously. Thus, the quality of the images captured by the device was improved. The captured images contained 1–3 unequal samples, with a total of 2,220 images captured, as shown in Figure 3.

2.1.3 Data processing

First, the collected images were divided into training (1,332), validation (444), and test (444) sets by batch at a 3:1:1 ratio. A multi-data-enhanced fusion method based on an adjustable range was implemented to enhance the robustness of the model under background differences in the kiwifruit images. The training set data were randomly combined using six methods: contrast, brightness, and rotation angle adjustment, mirroring, Gaussian noise addition, and filtering. The training dataset was enhanced seven times, resulting in a total of 10,656 images. The specific parameters are listed in Table 1. The experiment was conducted using a dataset in the Pascal Voc format and the dataset was labeled using labelImg. Four categories were labeled: “Kiwifruit,” “Leaf-rubbing damaged,” “Sunburned,” and “Healed cuts or scarred,” with the latter three categories corresponding to each defect type. The sample labeled “Kiwifruit” was used to locate the kiwifruit, but a single sample labeled “Kiwifruit” was considered as healthy.

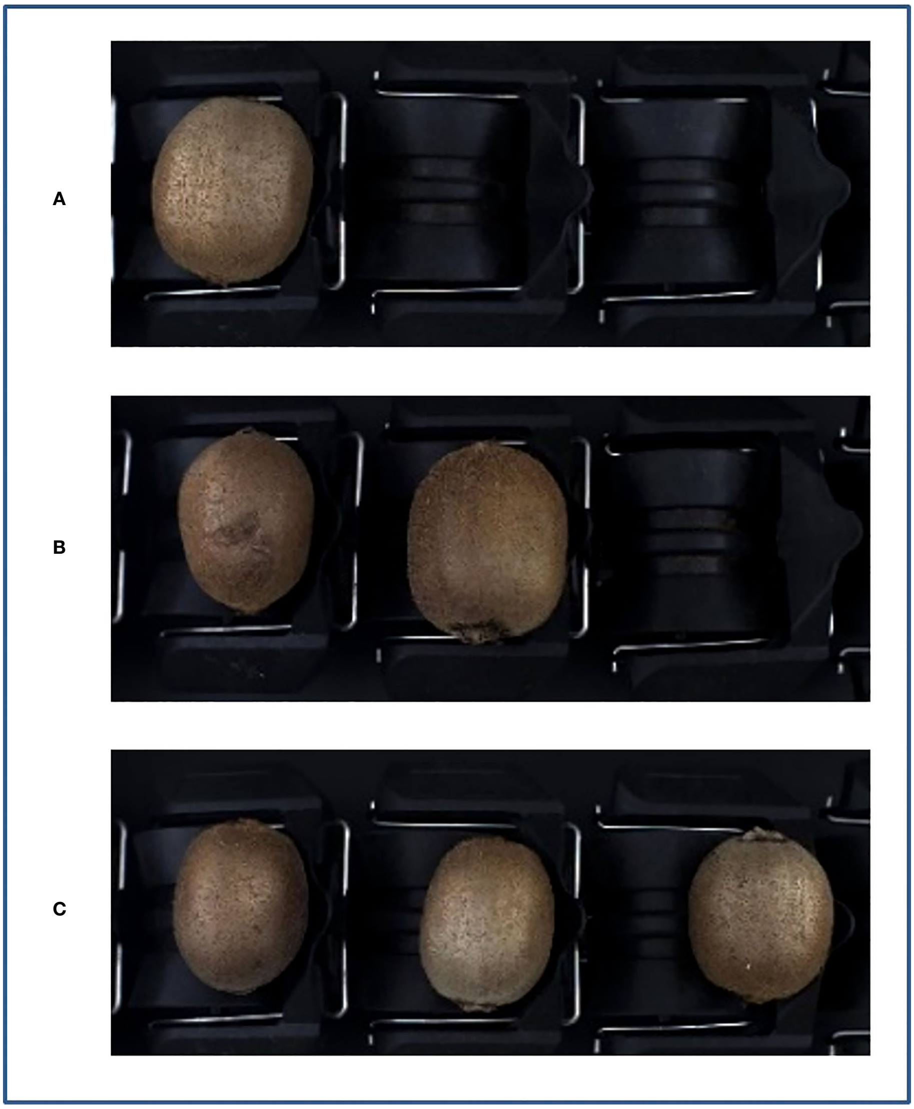


FIGURE 3
Image acquisition. **(A)** Single sample, **(B)** two samples, **(C)** three samples.

TABLE 1 Data enhancement methods.

Methods	Parameter range
Mirroring	/
Contrast ratio	(0.8, 1.2)
Gaussian noise	/
Filtering	/
Rotation angle	(-20°, 20°)
Brightness	(0.8, 1.2)

/, non-random variation.

2.2 Model construction

2.2.1 Experimental environment

The experimental operating platform was a Dell Precision 7920 Tower workstation (Dell, Round Rock, TX, USA) with an Ubuntu 18.04 64-bit operating system. The central processor of the workstation was an Intel Xeon Silver 4216 @ 2.10 GHz (X2; Intel, Santa Clara, CA, USA) with 128 G of running memory. The GPU was an NVIDIA GeForce RTX 3090 (Nvidia, Santa Clara, CA, USA) with a 24 G display memory. A deep learning framework with a GPU was used to accelerate the dynamic neural network Pytorch version 1.11, Anaconda 3.7 environment manager, and Python version 3.8.

2.2.2 Model structure

The structure of Yolov5-Ours, which was based on Yolov5s, is depicted in Figure 4. It included four parts: the input, backbone, neck, and prediction.

(a) Input: The input was a three-channel RGB image of kiwifruit, and the image size was uniformly adjusted from 1,600 × 1,200 to 640 × 640 at the acquisition time using adaptive picture scaling.

(b) Backbone: The backbone consisted of CBL, DWCBL, SPD-Conv, C3, and SPP. CBL consisted of convolutional and BN layers and leaky ReLU. The image size at the input was 640 × 640 × 3, and the output was 320 × 320 × 32 after slicing by the first CBL. DWCBL consisted of depth-wise separable convolution (DWConv) and BN layers and a Leaky ReLU. The DWConv layer with SPD-Conv (consisting of spatial-depth (SPD) and step-free convolutional layers) was implemented as the improved structure (the numbered part marked in Figure 4). The improved structure is described in detail in Section 2.3. C3 consisted of a CBL, residual structure, and convolutional layer connection, which could solve

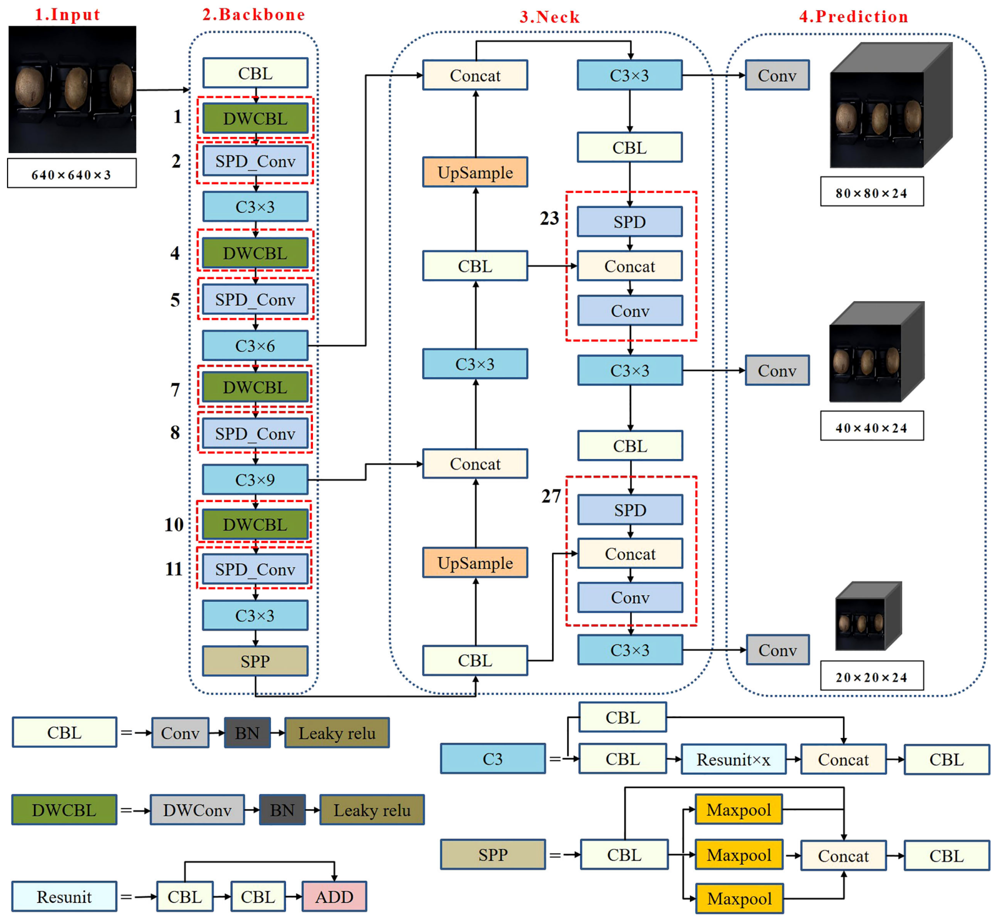


FIGURE 4
Yolov5-Ours network structure.

the problem of gradient repetition in the backbone network of the large CNN framework. Furthermore, it integrated the gradient changes into the feature map from beginning to end, thereby reducing the number of model parameters and computation values (Li et al., 2019) to ensure the speed and accuracy of the inference. SPP concatenated the different scales of the feature maps to expand the extraction of kiwifruit features using the maximum down-sampling of different convolutional kernels.

(c) Neck: FPN+PAN (Lin et al., 2017; Liu et al., 2018) was used. The FPN structure fuses and passes the feature information on the upper layers from top to bottom by up-sampling. The PAN structure is a bottom-up feature pyramid. The FPN+PAN structure was fused with feature layers from different backbone layers to improve the feature fusion capabilities further.

(d) Prediction: Output feature maps with sizes of 80 × 80, 40 × 40, and 20 × 20 were used to localize the kiwifruit defects. The training loss values were calculated using the loss calculation function and were iteratively updated to obtain the best model.

2.3 Structure optimization

2.3.1 SPD-Conv module

The convolution and pooling layers that are used in conventional methods lead to the loss of fine-grained information and insufficient learned features in the image. This results in small and low-resolution kiwifruit defect features that cannot be learned effectively during the convolution process. To address this problem, we incorporated the convolutional structure of SPD-Conv (Sunkara and Luo, 2022) into Yolov5s instead of the convolutional and pooling layers. When the feature size of the kiwifruit was a feature mapping X with a size of $M \times M \times C$, to achieve a two-fold down-sampling operation, the scale value S was selected as 2 in Equation (1). Subsequently, the SPD layer was subjected to spatial sub-mapping $f_{0,0}$、$f_{0,1}$、$f_{1,0}$、$f_{1,1}$ by slicing. These spatial sub-mappings were spliced in the channel dimension to acquire the dimensional mapping $X'(\frac{M}{S=2},\frac{M}{S=2},4C)$, and a step-free convolutional layer after SPD was added to obtain the final mapping $X''(\frac{M}{2},\frac{M}{2},C')$. The SPD layer preserved the information in the channel dimension when down-sampling was performed in the feature layer by retaining all information in the channel dimension when down-sampling the feature layer. The step-free layer retained the feature discriminant information in the convolution and adjusted the number of output channels. As illustrated in Figure 4, SPD-Conv was used as a substitute for four convolutional layers with a step size of 2 to down-sample the feature map in the backbone. Similarly, two alternative operations were executed in the neck.

$$
\begin{aligned}
&f_{0,0} = X[0:M:S, 0:M:S], \cdots f_{0,S-1} = X[0:M:S,\ S-1:M:S]\\
&f_{1,0} = X[1:M:S, 0:M:S], \cdots f_{1,S-1} = X[1:M:S,\ S-1:M:S]\\
&\vdots\\
&f_{S-1,0} = X[S-1:M:S, 0:M:S], \cdots f_{S-1,S-1} = X[S-1:M:S,\ S-1:M:S]
\end{aligned}
\tag{1}
$$

2.3.2 DWConv

The number of model calculation parameters and calculation amount increased following the structural improvement described in Section 2.3.1. We used DWConv (Chollet, 2017) instead of conventional convolution to solve this problem. The four regular convolutions in the backbone were replaced with DWConv, as indicated in Figure 4. As illustrated in Figure 5, the basic implementation process of DWConv consisted of depth-wise and point-wise convolution. Each convolution kernel of the depth-wise convolution convolved a single channel to make the number of input feature map channels the same as that of the output feature map channels. The point-wise convolution generated a new output feature map by linearly weighting the number of input feature map channels in the depth direction. DWConv effectively reduced the volume and computation of the parameters compared with conventional convolution for the same input and output cases.

2.4 Loss function

The target detection regression loss function IOU (Yu et al., 2016) cannot evaluate the distance information of the two frames when the prediction and target frames do not intersect. Thus, the gradient information cannot be passed back to the model, which results in the model not being learned and trained further.

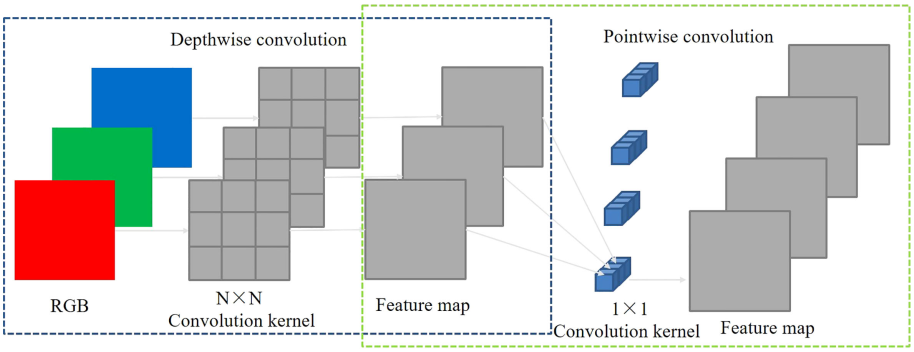


FIGURE 5
Schematic of DWConv.

Moreover, when the prediction and target frames intersect, the model cannot reflect the overlapping method of both frames. GIOU (Rezatofighi et al., 2019) introduces the minimum outer rectangle concept into the prediction and target frames. Although it solves the problems of IOU, errors, difficult convergences, and horizontal and vertical instability occur when the prediction and target frames have inclusion relations. DIOU (Xu et al., 2023) improves the penalty term in GIOU to calculate the distance between the minimized center point of the prediction and target frames to accelerate the convergence. However, DIOU does not consider the aspect ratio in the regression process. CIOU adds the influence factor to the penalty term based on DIOU and considers the prediction frame aspect ratio as fitting the target frame aspect ratio. However, the aspect ratio that is described by CIOU is a relative value and may be ambiguous. EIOU (Zhang et al., 2022) replaces the aspect ratio with the width-height difference value based on CIOU and introduces the focal loss to solve the problem of imbalance between difficult and easy samples. Therefore, EIOU was used as the loss calculation function in this study. The implementation process is illustrated in Figure 6 and the loss function value is calculated using Equation (2).

$$\begin{aligned} L_{EIOU} &= L_{IOU} + L_{dis} + L_{asp} \\ &= 1 - IOU + \frac{d^2(b^P, b^{gt})}{(w^c)^2 + (h^c)^2} + \frac{d^2(w^P, w^{gt})}{(w^c)^2} + \frac{d^2(h^P, h^{gt})}{(h^c)^2}, \end{aligned} \tag{2}$$

where L_{IOU} is the overlap loss, L_{dis} is the center distance loss, and L_{asp} is the scale loss. Furthermore, b^P and b^{gt} are the coordinates of the center points of the prediction and target frames, respectively, whereas $d(b^P, b^{gt})$ is the Euclidean distance between the frames. w^c and h^c are the width and height of the smallest outer rectangle of the prediction and target frames, respectively. Moreover, IOU is the ratio of the intersection of the prediction and target frames to the union, $d(w^P, w^{gt})$ is the difference between the widths of the prediction and target frames, and $d(h^P, h^{gt})$ is the difference between the lengths of the prediction and target frames.

2.5 Evaluation indicators

To evaluate the effectiveness of the external defect detection model for kiwifruit, multiple metrics were used, including the rate of precision and recall, number of parameters (Params) and FLOPs (Li et al., 2021), model size, average precision (AP) of a single sample, and average precision (mAP) of all categories. The precision and recall are determined by Equations (3) and (4), respectively.

$$P = TP/(TP + FP) \times 100\,\% \tag{3}$$

$$R = TP/(TP + FN) \times 100\,\%, \tag{4}$$

where P is the precision rate; that is, the proportion of predicted targets that are the same as the labeled targets, and R is the recall rate; that is, the proportion of correctly predicted positive samples to all labeled positive samples. TP represents the predicted positive and actual positive samples, FP represents the predicted positive and actual negative samples, and FN represents the predicted negative and actual positive samples.

The curve for PR was plotted with R and P as the horizontal and vertical coordinates, respectively, and the area enclosed by the curve was calculated to obtain AP. The calculation of mAP is shown in Equations (5) and (6).

$$AP = \int_0^1 P(R)\mathrm{d}R \times 100\,\% \tag{5}$$

$$mAP = \frac{1}{C}\sum_{c \in C} AP(c) \times 100\,\%, \tag{6}$$

where c is a single category and C is all categories.

3 Results and discussion

3.1 Model training results

A stochastic gradient descent optimizer with a momentum of 0.937 and a weight decay of 0.0005 was selected to evaluate the performance of the proposed network. The number of training warm-up rounds, total number of rounds, and training batches were set to 3, 200, and 32, respectively. The training learning rate was set linearly from 0.003 to 0.01 following the warm-up phase and decayed linearly to a final value of 0.0001 after 200 iterations.

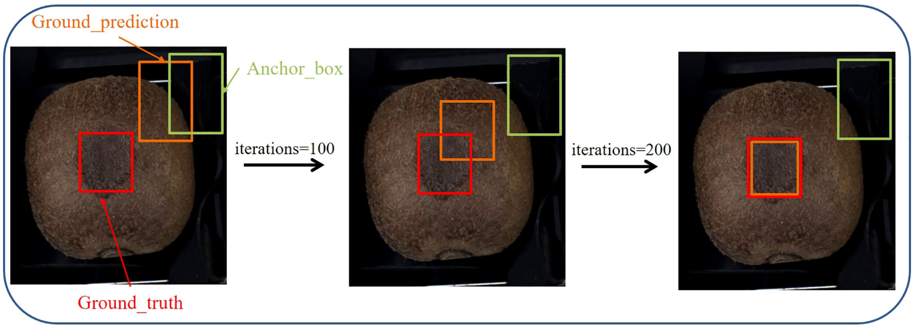


FIGURE 6
Schematic of EIOU implementation.

The loss value is a metric that is used to measure the effectiveness of network training. Figure 7 shows the loss values of Yolov5s and Yolov5-Ours in the training set. The loss value of Yolov5-Ours decreased rapidly to approximately 0.08 from the beginning of the iterations, and then steadily with an increase in iterations. The initial loss value of Yolov5s was larger than that of Yolov5-Ours; the loss value decreased more slowly and appeared to fluctuate with the increase in iterations. After 200 iterations, the loss value of Yolov5-Ours was 0.050 and that of Yolov5s was 0.063. Thus, Yolov5-Ours reduced the loss value by 0.013 compared to Yolov5s.

The AP of the training detection provides an important indication of whether the model has learned the features effectively. Figure 8 depicts the average class detection accuracies of Yolov5s and Yolov5-Ours in the training set. From the beginning of the iterations, the detection mAP increased while Yolov5s and Yolov5-Ours learned the kiwifruit defect features. Yolov5-Ours reached convergence at 100 iterations and the detection mAP was slightly higher than that of Yolov5s. After 200 iteration rounds, both Yolov5s and Yolov5-Ours reached stability, and both had better detection mAPs for kiwifruit defects, but that of Yolov5-Ours was slightly higher than that of Yolov5s. The Yolov5-Ours model achieved a detection accuracy of 99.4% for healthy kiwifruit, 99.3% for leaf-rubbing damaged kiwifruit, 97.7% for healed cuts or scarred kiwifruit, and 99.2% for sunburned kiwifruit during the validation phase on 444 kiwifruit images.

The number of parameters and computations were visualized in terms of the spatial and temporal complexity for the model size and speed, respectively. Spatial complexity refers to the consumption of computer hardware memory resources, whereas temporal complexity is the model computation time. The number of parameters and amount of computation during the training process of Yolov5s, Yolov5s+SPD-Conv, and Yolov5-Ours were determined, as indicated in Table 2. The number of parameters of Yolov5s+SPD-Conv increased by 1.54 M and the computation amount increased by 17.5 G compared to Yolov5s. The number of parameters of Yolov5-Ours decreased by 1.56 M and the computation amount decreased by 15.1 G compared to Yolov5s+SPD-Conv. These results demonstrate the effectiveness of the model improvement described in Section 2.3.1.

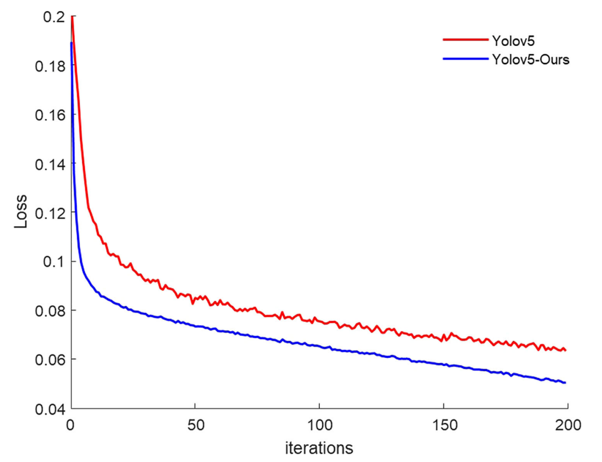


FIGURE 7
Training loss value.

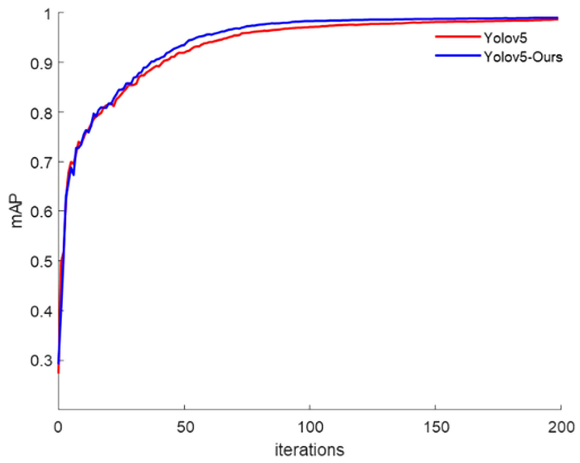


FIGURE 8
Average accuracy of training categories.

3.2 Model testing results

The 444 test set images contained 1,151 kiwifruit samples, including 326 healthy, 268 leaf-rubbing damaged, 284 healed cuts or scarred, and 273 sunburned samples. The samples in each category were tested using Yolov5-Ours with optimal weights. As indicated in Table 3, the precision rates for the four categories were all higher than 99% and the recall rates were all higher than 95%. The average detection precisions of the healthy, leaf-rubbing damaged, healed cuts or scarred, and sunburned samples were 98.8%, 98.7%, 97.6%, and 95.9%, respectively, at a confidence threshold of 0.5, whereas the detection mAP of all categories was 97.7%. Moreover, the detection time of the image was only 8.0 ms, thereby meeting the real-time sorting requirements of the grading line. As shown in a partial plot of the results (Figure 9), Yolov5-Ours could effectively detect all categories at a confidence level higher than 0.8 for each category, which suggests that the model is highly adaptable and robust for each category of kiwifruit.

3.3 Model comparison

The sample mAP and model sizes of SSD, Yolov5s, Yolov7, and Yolov5-Ours were compared to validate the performance of Yolov5-Ours further. As shown in Table 4, the mAP of the

TABLE 2 Number of parameters and calculated values.

Model	Params (M)	FLOPs (G)
Yolov5s	7.03	15.9
Yolov5s+SPD-Conv	8.57	33.4
Yolov5-Ours	7.01	18.3

TABLE 3 Test results.

Category	P (%)	R (%)	AP@0.5 (%)	mAP@0.5 (%)	Image (ms)
Healthy	99.8	97.1	98.8	97.7	8.0
Leaf-rubbing damaged	99.7	96.7	98.7		
Healed cuts or scarred	99.5	98.3	97.6		
Sunburned	1.0	95.1	95.9		

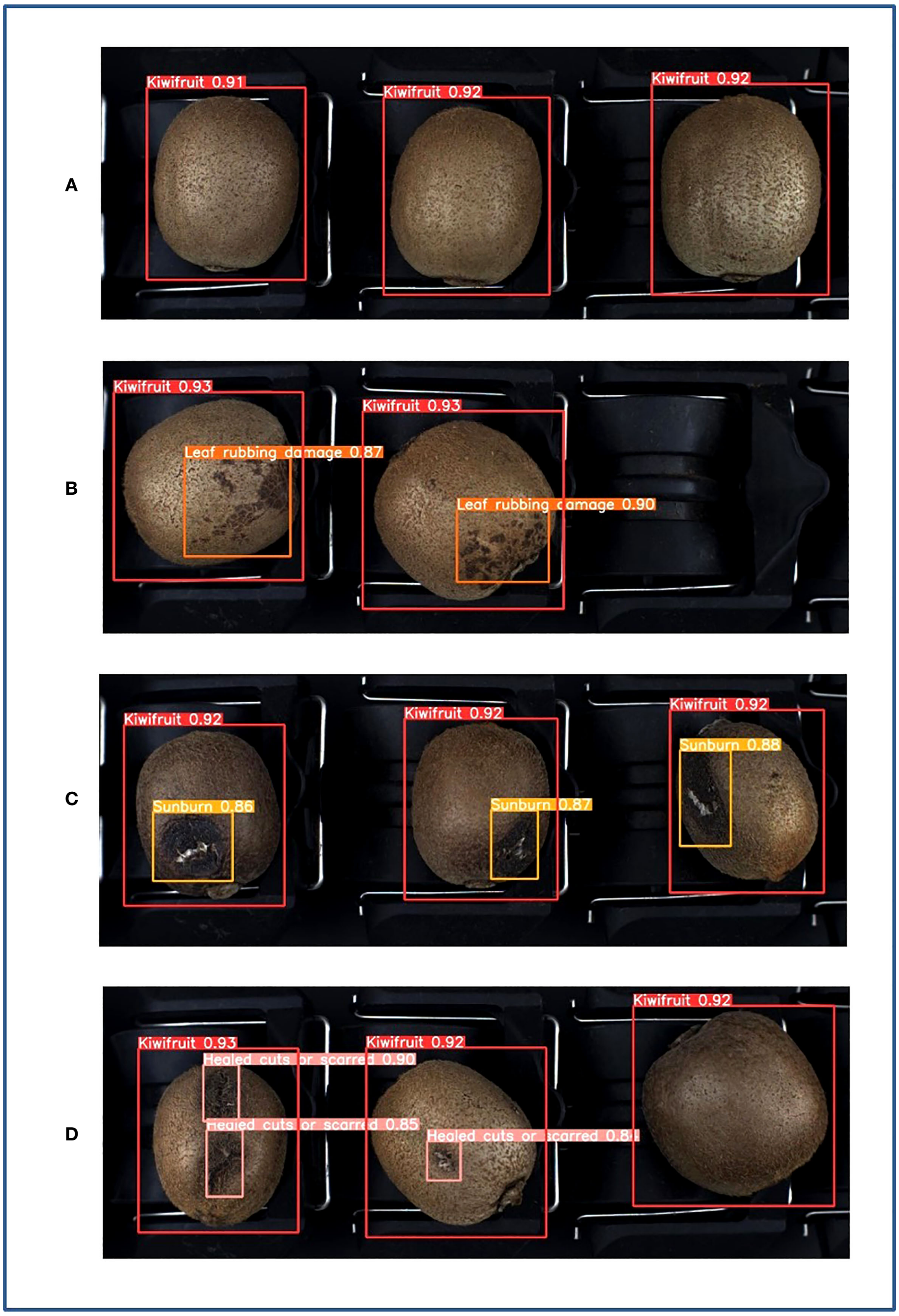


FIGURE 9
Test results. **(A)** Healthy, **(B)** leaf-rubbing damaged, **(C)** sunburned, **(D)** healed cuts or scarred.

TABLE 4 Model comparison.

Model	mAP@0.5 (%)	mAP@0.8 (%)	Weight (MB)
SSD	87.7	72.8	108.1
Yolov5s	91.3	78.2	14.4
Yolov7	98.8	91.5	74.8
Yolov5-Ours	97.7	88.3	14.4

samples was compared at confidence threshold values of 0.5 and 0.8. When the confidence level was 0.5, the mAP of Yolov5-Ours was 1.1% lower than that of Yolov7, but 10% and 6.4% higher than those of SSD and Yolov5s, respectively. When the confidence level was 0.8, the mAP of Yolov5-Ours was 88.3%, 15.5%, and 10.1% higher than those of SSD and Yolov5s, but 3.2% lower than that of Yolov7. The model size of Yolov5-Ours was the same as that of Yolov5s, which was approximately 6.5- and 4-fold smaller than those of SSD and Yolov7, respectively.

SSD is mainly divided into the backbone network and multi-scale prediction network. The backbone network adopts the VGG16 model, which is used to realize the initial extraction of image features. The multi-scale feature detection network extracts the feature layers that are obtained from the backbone network at different scales, so that different feature maps can detect different-sized features. Finally, the detection results are regressed. Yolov7 introduces model reparameterization into the network structure, includes a new label assignment method, and incorporates multiple tricks for efficient training compared to Yolov5. Yolov7 achieves higher computational efficiency and accuracy than Yolov5, and can achieve better detection accuracy with the same computational resources. However, Yolov5 is much faster than Yolov7 in terms of the inference speed, because the faster computational efficiency of Yolov7 leads to more memory-occupied resources. Yolov5-Ours improves the detection of small feature defects on the surface of kiwifruit by adding the SPD-Conv module based on Yolov5s and reduces the parameters using DWConv, which means that the model size does not increase even with higher detection accuracy. In summary, the results verified that Yolov5-Ours balances the model size and accuracy and achieves efficient performance in kiwifruit defect detection.

4 Conclusions

We developed and validated the effectiveness of a non-destructive detection method for kiwifruit defects. We applied the target detection technique to multiple healthy and defective kiwifruits and improved several aspects, including the data acquisition and methodology, to detect kiwifruit defects in various categories efficiently. First, a kiwifruit image acquisition device was constructed and improved to solve the problem of uneven light exposure in the image, thereby improving the image quality. Subsequently, a kiwifruit database was established. To avoid the problem of overfitting, the training dataset was increased seven-fold using a new data enhancement method. We proposed Yolov5-Ours based on Yolov5s, in which we fused SPD-Conv and DWConv and improved the loss calculation function. The average detection accuracy of healthy, leaf-rubbing damaged, healed cuts or scarred and sunburned samples was 97.7%. The single-frame image detection was run in 8.0 ms, thereby meeting the classification line-sorting requirements. The results validated the effectiveness of Yolov5-Ours in terms of both the accuracy and model size.

The external kiwifruit defects of sunburned and healed cuts or scarred affect the flesh of the kiwifruit, and effective detection can increase the commercial value of the kiwifruit. Leaf-rubbing damaged kiwifruit only has defects in the skin and the flesh of the kiwifruit is normal, and correct detection can increase the reuse of iso-extracted fruits. Consequently, the proposed method can facilitate the effective detection of kiwifruit defects, provide a theoretical basis for online real-time detection and grading, and serve as a framework for future non-destructive defect detection in agricultural products.

This study also has some shortcomings. Only three major kiwifruit defects were selected for detection and sorting. We plan to expand the categories of kiwifruit defects for detection in the future, which will make the study more applicable to actual kiwifruit sorting.

Data availability statement

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

Author contributions

FW and BZ designed the study, performed the experiments, analyzed the data, and wrote the manuscript. CL supervised the project and helped to design the research. LD, XL, and PG performed the experiments. All authors have contributed to the manuscript and approved the submitted version.

Funding

This study was funded by the Beijing Nova Program (No. 20220484066).

Conflict of interest

Authors are employed by company Mechanization Sciences Group Co., Ltd.

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

Azizah, L. M., Umayah, S. F., Riyadi, S., Damarjati, C., and Utama, N. A. (2017). "Deep learning implementation using convolutional neural network in mangosteen surface defect detection," in *2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia. 242–246.

Chollet, F. (2017). "Xception: deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. 1800–1807.

Cui, Y. J., Li, P. P., Ding, X., and Su, S. (2012). Detection of surface defects in graded kiwifruit fruits. *J. Agric. Mech. Res.* 34, 139–142.

Fan, S. X., Li, J. B., Zhang, Y. H., Tian, X., Wang, Q. Y., 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

Fu, L. S., Feng, Y. L., Elkamil, T., Liu, Z. H., Li, R., and Cui, Y. J. (2018). Image recognition method of multi-cluster kiwifruit in field based on convolutional neural networks. *Trans. CSAE* 34, 205–211.

Fu, L. S., Sun, S. P., Li, R., and Wang, S. J. (2016). Classification of kiwifruit grades based on fruit shape using a single camera. *Sensors* 16, 1012. doi: 10.3390/s16071012

Jahanbakhshi, A., Momeny, M., Mahmoudi, M., and Zhang, Y. D. (2020). Classification of sour lemons based on apparent defects using stochastic pooling mechanism in deep convolutional neural networks. *Sci. Hortic.* 263, 109–133. doi: 10.1016/j.scienta.2019.109133

Li, D., Hu, J., Wang, C. H., Li, X. T., She, Q., Zhu, L., et al. (2021). "Involution: inverting the inherence of convolution for visual recognition," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA. 12316–12325. doi: 10.48550/arXiv.2103.06255

Li, J., Huang, B. H., Wu, C. P., Su, Z., Xue, L., Liu, M. H., et al. (2022). Effect of hardness on the mechanical properties of kiwifruit peel and flesh. *Int. J. Food. Prop.* 25 (1), 1697–1713. doi: 10.1080/10942912.2022.2098972

Li, Y. S., Qi, Y. N., Mao, W. H., Zhao, B., Lv, C. X., Ren, C., et al. (2018). Automatic weighing and grading system for balsam pears. *Agric. Eng.* 8, 63–68.

Li, X., Wang, W. H., Hu, X. L., and Yang, J. (2019). "Selective kernel networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA. 510–519.

Li, K. S., Wang, J. C., Jalil, H., and Wang, H. (2023). A fast and lightweight detection algorithm for passion fruit pests based on improved YOLOv5. *Comput. Electron. Agric.* 204, 107534. doi: 10.1016/j.compag.2022.107534

Li, J., Wu, C. P., Liu, M. H., Chen, J. Y., Zheng, J. H., Zhang, Y. F., et al. (2020). Hyperspectral imaging for shape feature detection of kiwifruit. *Spectro. Spectral Anal.* 40, 2564–2570.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. 936–944.

Liu, H. Z., Chen, L. P., Mu, L. T., Gao, Z. B., and Cui, Y. J. (2020). A K-means clustering-based method for kiwifruit flower identification. *J. Agric. Mech. Res.* 42, 22–26.

Liu, Z. C., and Gai, X. H. (2020). Design of kiwifruit grading control system based on machine vision and PLC. *Chin. J. Agric. Chem.* 41, 131–135.

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*, Salt Lake City, UT, USA. 8759–8768.

Lu, S., Chen, W., Zhang, X., and Manoj, K. (2022). Canopy-attention-YOLOv4-based immature/mature apple fruit detection on dense-foliage tree architectures for early crop load estimation. *Comput. Electron. Agric.* 193, 106696. doi: 10.1016/j.compag.2022.106696

Luna, R. G. D., Dadios, E. P., Bandala, A. A., and Vicerra, R. R. P. (2019). "Tomato fruit image dataset for deep transfer learning-based defect detection," in *2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, Bangkok, Thailand. 356–361.

Momeny, M., Jahanbakhshi, A., Jafarnezhad, K., and Zhang, Y. D. (2020). Accurate classification of cherry fruit using deep CNN based on hybrid pooling approach. *Postharvest. Biol. Technol.* 166, 111204. doi: 10.1016/j.postharvbio.2020.111204

Rezatofighi, H., Tsoi, N., Gwak, J. Y., Sadeghian, A., Reid, I., and Savarese, S. (2019). "Generalized intersection over union: a metric and a loss for bounding box regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA. 658–666.

Sunkara, R., and Luo, T. (2022). "No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects," in *Computer vision and pattern recognition*. (Cham: Springer Nature Switzerland), 443–459. doi: 10.48550/arXiv.2208.03641

Tian, Y. W., Wu, W., Lu, S., and Deng, H. B. (2021). Application of deep learning in fruit quality detection and grading classification. *Food. Sci.* 42, 260–270.

Wang, Z. D., Hu, M. H., and Zhai, G. T. (2018). Application of deep learning architectures for accurate and rapid detection of internal mechanical damage of blueberry using hyperspectral transmittance data. *Sensors* 18, 1126. doi: 10.3390/s18041126

Xu, B., Cui, X., Ji, W., Yuan, H., and Wang, J. (2023). Apple grading method design and implementation for automatic grader based on improved YOLOv5. *Agriculture* 13, 124. doi: 10.3390/agriculture13010124

Xue, Y. J., Huang, N., Tu, S. Q., Mao, L., Yang, A. Q., Zhu, X. M., et al. (2018). An improved YOLOv2 identification method for immature mangoes. *Trans. CSAE* 34, 173–179.

Yang, T., Ma, J. J., and Lei, J. (2021). A grading method for kiwifruit based on surface defect identification. *Hubei Agric. Sci.* 60, 145–148.

Yu, J., Jiang, Y., Wang, Z., Cao, Z. M., and Huang, T. (2016). "Unitbox: an advanced object detection network," in *Proceedings of the 24th ACM international conference on Multimedia*. 516–520. doi: 10.1145/2964284.2967274

Yu, X. J., Lu, H. D., and Wu, D. (2018). Development of deep learning method for predicting firmness and soluble solid content of postharvest Korla fragrant pear using Vis/NIR hyperspectral reflectance imaging. *Postharvest. Biol. Technol.* 141, 39–49. doi: 10.1016/j.postharvbio.2018.02.013

Zhang, M., Jiang, Y., Li, C., and Yang, F. Z. (2020). Fully convolutional networks for blueberry bruising and calyx segmentation using hyperspectral transmittance imaging. *Biosyst. Eng.* 192, 159–175. doi: 10.1016/j.biosystemseng.2020.01.018

Zhang, Y. F., Ren, W. Q., Zhang, Z., Jia, Z., Wang, L., and Tan, T. N. (2022). Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* 506, 146–157. doi: 10.1016/j.neucom.2022.07.042

Zhou, Y. T., Bing, F., Wang, W. J., and Tian, L. S. (2012). Research on kiwifruit defect detection based on image processing. *Comput. Knowl. Technol.* 8, 3979–3981.

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 08 December 2023
DOI 10.3389/fpls.2023.1139666

OPEN ACCESS

EDITED BY
Long He,
The Pennsylvania State University (PSU),
United States

REVIEWED BY
Rui Liu,
Ningxia University, China
Pengpeng Sun,
Northwest A&F University, China
Muthusamy Ramakrishnan,
Nanjing Forestry University, China
Anupun Terdwongworakul,
Kasetsart University, Thailand

*CORRESPONDENCE
Yu Zhang
✉ yuzhang2015@hainanu.edu.cn
Jing Chen
✉ jingchen_haiko@163.com
Mengxing Huang
✉ huangmx09@hainanu.edu.cn
Chengxu Sun
✉ hnsuncx@qq.com

RECEIVED 07 January 2023
ACCEPTED 22 November 2023
PUBLISHED 08 December 2023

CITATION
Liu Q, Zhang Y, Chen J, Sun C, Huang M, Che M, Li C and Lin S (2023) An improved Deeplab V3+ network based coconut CT image segmentation method. *Front. Plant Sci.* 14:1139666. doi: 10.3389/fpls.2023.1139666

An improved Deeplab V3+ network based coconut CT image segmentation method

Qianfan Liu[1], Yu Zhang[1*], Jing Chen[2*], Chengxu Sun[3*], Mengxing Huang[4*], Mingwei Che[4], Chun Li[4] and Shenghuang Lin[2]

[1]School of Computer Science and Technology, Hainan University, Haikou, China, [2]Central South University Xiangya School of Medicine Affiliated Haikou Hospital, Haikou, China, [3]Coconut Research Institute, Chinese Academy of Tropical Agricultural Sciences, Wenchang, Hainan, China, [4]School of Information and Communication Engineering, Hainan University, Haikou, China

Due to the unique structure of coconuts, their cultivation heavily relies on manual experience, making it difficult to accurately and timely observe their internal characteristics. This limitation severely hinders the optimization of coconut breeding. To address this issue, we propose a new model based on the improved architecture of Deeplab V3+. We replace the original ASPP(Atrous Spatial Pyramid Pooling) structure with a dense atrous spatial pyramid pooling module and introduce CBAM(Convolutional Block Attention Module). This approach resolves the issue of information loss due to sparse sampling and effectively captures global features. Additionally, we embed a RRM(residual refinement module) after the output level of the decoder to optimize boundary information between organs. Multiple model comparisons and ablation experiments are conducted, demonstrating that the improved segmentation algorithm achieves higher accuracy when dealing with diverse coconut organ CT(Computed Tomography) images. Our work provides a new solution for accurately segmenting internal coconut organs, which facilitates scientific decision-making for coconut researchers at different stages of growth.

KEYWORDS

coconut, CT images, semantic segmentation, DASPP, CBAM, RRM

1 Introduction

As a plant native to tropical environments, coconuts not only serve as distinctive landscape trees for tourism, but also contribute significantly to the local economy as a pillar industry. The various structures within coconuts are essential materials in other industries and closely linked to people's lives (Arumugam and Hatta, 2022). As a result, the development of the coconut industry has garnered high attention and research efforts worldwide. However, the unique growth environment of coconuts, coupled with factors such as extensive farming practices, limited processing enterprises, weak risk resilience, low technological content, and backward deep processing capabilities, have led to insufficient raw materials and severe homogeneity issues in coconut products. Currently, the global coconut market is facing a severe supply-demand imbalance, with a significant shortage of high-quality coconuts. Consequently, the cultivation of superior coconut seeds has become a research hotspot in order to provide higher-quality seedlings and resources for the coconut industry. Real-time monitoring of the internal structural growth during the cultivation process has become the key to addressing this issue. Currently, growers can only resort to destructive methods, such as cutting open coconuts for observation and documentation, which not only hampers the normal growth of the coconut but is also unsuitable for large-scale cultivation research. However, the use of X-ray imaging methods can be effectively applied in this scenario.

Computed tomography (CT) imaging, widely used in clinical medicine, provides clear visualization of internal structures in the human body, aiding doctors in obtaining crucial information for diagnosing organs or tissues. It holds significant importance in quantitative pathological assessment, treatment planning, and disease progression monitoring. By applying this method to agricultural research, utilizing the penetrating characteristics of X-rays, we can obtain clear internal organ images of coconuts without disrupting their normal physiological structure and growth (Zhang et al., 2023).

For image segmentation tasks, traditional segmentation methods suffer from poor robustness, low efficiency, and low accuracy. With the development of deep learning techniques, image segmentation can be achieved without relying on manually designed features, as neural networks can automatically learn the features required for segmentation tasks. Therefore, methods based on deep learning have become the primary choice for researchers in various image segmentation tasks (Suk et al., 2023). However, existing deep learning-based image segmentation algorithms have significant limitations when it comes to organ segmentation tasks in coconut CT images, failing to meet the high-precision segmentation requirements in agriculture. In response to these issues, this paper proposes corresponding improvement methods and validates the effectiveness and superiority of the proposed methods through ablation experiments and comparative experiments. The model proposed in this paper can obtain higher-precision semantic information when facing coconut CT images, facilitating a more detailed analysis and evaluation of coconut development and growth.

Our work has made the following main contributions:

1. We conducted non-destructive observations of coconuts at different stages and with different characteristics through CT scanning. We obtained internal images of coconuts at multiple time periods and multiple categories. Based on the growth conditions of coconuts, we classified and labeled the internal organs of coconuts, establishing a CT-based coconut organ image dataset named "CIDCO." These data were used for training and testing the network model we constructed and also provided image resources for coconut research.
2. To achieve precise segmentation of the internal structure of coconuts, we proposed an improved image segmentation method based on the modified Deeplab V3+ network. Through model comparison, we demonstrated that the improved network achieves higher segmentation accuracy and can be effectively applied to coconut image segmentation and growth development research.

The structure of this paper is as follows: In Section 2, we introduce and analyze relevant research on non-destructive observations and image segmentation for agricultural applications. In Section 3, we summarize the research methods used in this work. Section 4 presents the experiments we designed and compares the results with other models. In Section 5, we provide a summary of the entire work and discuss future directions and ideas.

2 Related work

The use of non-destructive methods to acquire images of target objects has been receiving increasing attention and gradually being applied in various research fields. CT, ultrasound, infrared laser, nuclear magnetic resonance, and other methods have been used for image scanning. For example, Yu et al. (2022) employed electron microscopy CT for non-destructive observation of coconut variations, aiming to explore growth and development. Li et al. (2020) conducted terahertz imaging to observe changes in leaf water content in their research on crop water status monitoring and diagnosis. These studies demonstrate the feasibility of obtaining images of target objects through non-destructive means. Regarding image segmentation, traditional methods include threshold determination, region-based similarity aggregation, edge operator calculations, and energy-minimizing active contour-based approaches to accomplish various segmentation tasks. For instance, Thorp and Dierig (2011) presented a color image segmentation method to monitor the flowering status of Lesquerella. This method converts the RGB color space to the HSI color space and utilizes histogram equalization to enhance image contrast. Then, threshold segmentation is used to separate the flower parts from the background, and morphological operations and region-growing algorithms are employed to remove noise and connect discontinuous flower parts. Finally, the number of flowers is counted based on the segmentation results, achieving automatic monitoring of Lesquerella flowering. Xiang (2018) introduced an image segmentation method for nighttime identification of the entire tomato plant. This method first converts the image to the HSV color space and then separates the plant from the background using threshold segmentation. However, these traditional methods perform reasonably well when dealing with images with simple linear features. But once other factors increase, they can greatly affect the segmentation results. With the rise of deep neural networks, various neural network methods have been quickly applied to various image segmentation tasks. Deep learning-based methods fundamentally transform semantic segmentation into an image per-pixel classification problem. Van De Looverbosch et al. (2021) proposed a non-destructive internal defect detection method for pears using deep learning techniques. X-ray CT scanning is employed to acquire images, and semantic segmentation techniques are used for internal defect detection and recognition. Ni et al. (2020) utilized deep learning techniques to segment and extract features from blueberry fruit images in order to better predict the harvest period and yield of blueberry fruits. This research provides a new method for accurately predicting fruit harvest and yield. Sun et al. (2021) employed semantic segmentation networks and shape-constrained level set methods to detect and segment images of apple, peach, and pear flowers. The research results demonstrate that this approach can more accurately detect and segment the contours of flowers. Turgut et al. (2022) proposed a deep learning architecture called RoseSegNet for plant organ segmentation. This model, based on attention mechanisms, can identify different organs of a rose, including petals, stamens, and leaves, providing a new tool for botanical research. Singh et al. (2022) proposed a method for semantic segmentation of cotton structures from aerial images using deep convolutional neural networks. This research achieved automatic identification and segmentation of cotton bolls from the sky using deep convolutional neural networks. This method can improve cotton harvesting efficiency, reduce costs, and provide new technological support for modern agriculture. The introduction of deep learning networks has brought faster and more accurate solutions to image segmentation tasks. However, due to the unique characteristics of coconuts, there is still limited research on the application of high-precision semantic segmentation models in coconut CT images. Therefore, our focus is on addressing this issue.

3 Method

3.1 Coconut data collection and scanning

Considering the suitable average temperature for coconuts to be maintained between 24 to 27°C, with ample precipitation and an annual sunlight guarantee of more than 2000 hours, and in order to obtain richer raw material resources in large-scale cultivation areas, after careful consideration, the experimental fields of Wenchang Coconut Research Institute and the coconut plantation in Leiming Town, Ding'an County were selected as the collection sites. The experimental fields adopted a triangular planting pattern to achieve higher yields per unit area, mainly consisting of green coconuts, red coconuts, and yellow coconuts, covering an age range of 3 to 12 months. The coconut trees in the plantation are approximately 20 years old, with a height of 10 meters and 30 leaves. The majority of coconuts produced are green coconuts at the stage of 7 to 12 months. Refer to Figure 1 for illustration.

In the aforementioned field conditions, a total of 104 coconuts were collected, categorized into different groups based on color, type, and age. The coconuts were numbered according to their growth months in sequential order. Using the anatomical scanning of the human body as the reference position, they were scanned using a Siemens 256 dual-source CT machine. X-rays were used to obtain cross-sectional images in three directions: axial, coronal, and sagittal. This process resulted in complete multi-angle sliced images of each coconut. Considering that a single image may contain more than one complete target coconut, additional coconuts with varying representations were also included in the CT scan images. The number of images obtained from each coconut scan ranged from 170 to 220, with approximately one-fourth of the images capturing the complete structural information. An example of the coconut scanning process is shown in Figure 2.

Each image is labeled in the format of "color_month_id" to facilitate quick and accurate searching. The labeled images are then stored and organized according to the major coconut varieties, with

FIGURE 1
Coconut collection area situation.

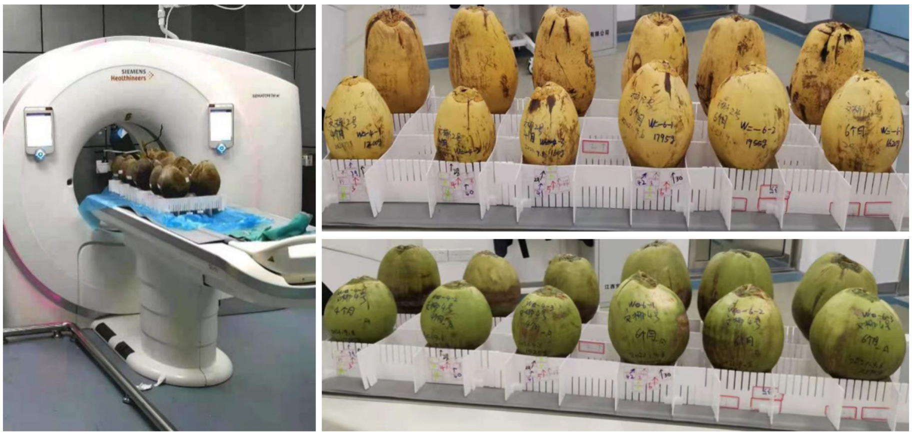


FIGURE 2
Example of coconut scan.

corresponding annotation folders created. Coconut researchers and project members were involved in the annotation process. The four main organs of the coconut that are most relevant to its development and growth are the absorber, solid endosperm (coconut meat), liquid endosperm (coconut water), and embryo. These four organs were annotated, with the background represented in black by default. The absorber was annotated in yellow, the solid endosperm in red, and the liquid endosperm in blue. Coconut CT images can be seen in Figure 3, and the corresponding annotation results are shown in Figure 4.

3.2 Design of segmentation model

Given the limitations of the original Deeplab V3+ network, such as insufficient utilization of inter-level feature information leading to unclear segmentation boundaries and lack of detailed feature map information, resulting in poor final results, we propose a new semantic segmentation model for coconut CT images. The improved model builds upon the advantages of the original framework's encoder-decoder architecture and enhances the feature recognition and capture capabilities through module replacement and addition.

After the input of the task image, the Deeplab V3+ model first uses a deep convolutional network (DCNN) to extract features from the input image, dividing them into two categories: high-level semantic features and low-level semantic features. Some of the low-level features directly enter the decoder, while other information enters the encoder stage. At this point, the Atrous Spatial Pyramid Pooling (ASPP) module is introduced to capture coconut organ features and requires a sufficiently large receptive field. However, increasing the dilation rate leads to sparser pixel sampling compared to traditional convolution, resulting in more loss of detail information. As a result, the original ASPP module experiences attenuation in the effectiveness of dilated convolutions, and the effectiveness of atrous convolutions gradually decreases, ultimately affecting the model's capabilities.

Furthermore, the original network employs a 4x upsampling in the decoder stage. For coconut organs, large-scale upsampling adversely affects edge segmentation. Moreover, the fusion with only low-level features from the base network may result in the loss of some information, thus affecting the final segmentation accuracy.

To address these issues, the Dense Atrous Spatial Pyramid module is used to replace the original ASPP module. The input-output dense connections are established between each atrous convolution layer, allowing for the coverage of multi-scale range

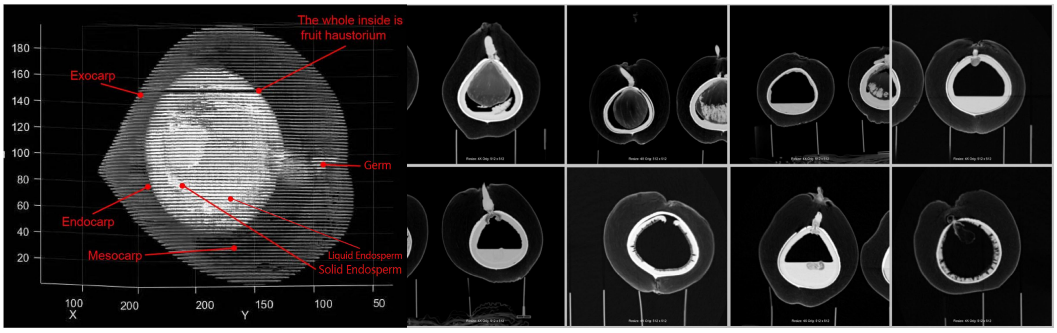


FIGURE 3
Example of CT image of coconut.

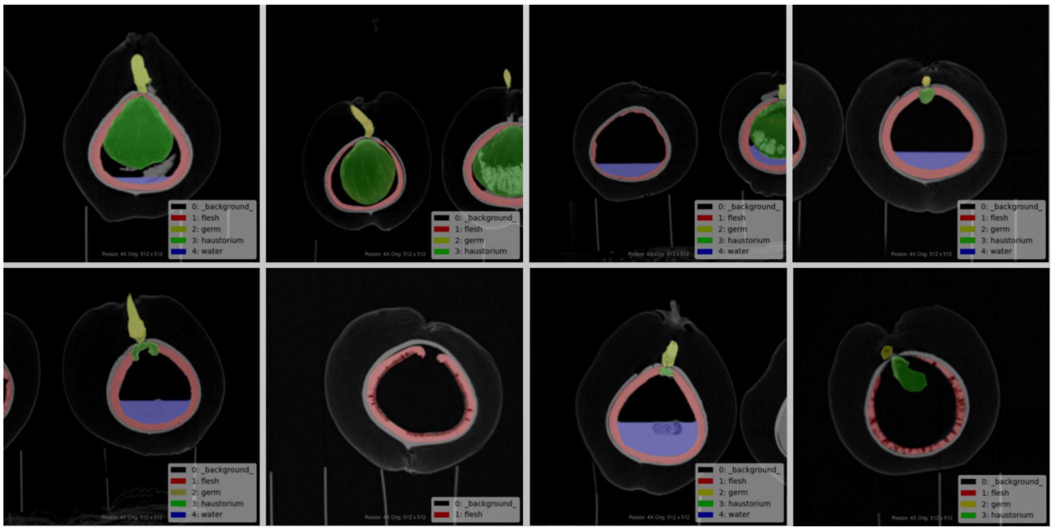


FIGURE 4
Example of the corresponding labeled diagram.

feature information using appropriate dilation rates. Additionally, a convolutional attention module is introduced to enhance effective feature information, suppress irrelevant information responses, and improve feature extraction and representation capabilities. Finally, a residual refinement module is embedded after the decoder to map the significant information transmitted from the upper layers, optimizing organ boundaries and improving segmentation accuracy. The improved model is illustrated in Figure 5.

3.3 Principle of the improvement module

3.3.1 DASPP module

DASPP stands for “Dense Atrous Spatial Pyramid Pooling.” In the structure of the DASPP module, atrous convolutions are combined into a cascaded fusion operation. The dilation rate increases layer by layer, with layers having lower dilation rates placed in the lower-level parts and layers with higher dilation rates placed in the higher-level parts. The subsequent layers share information with the preceding layers, using their features for information sharing. This dense connectivity allows for more intensive pixel utilization. Each atrous layer concatenates the input with the output of the previous lower-level layer as its input, ultimately producing a feature map generated by multi-scale atrous convolutions.

Compared to traditional ASPP, DASPP utilizes dense connections to establish interconnections between layers with different dilation rates. Each set can be considered as a convolutional kernel of a different scale, representing different receptive fields. This change brings about a denser feature pyramid and a larger receptive field, allowing for better recognition and integration of semantic features of target organs

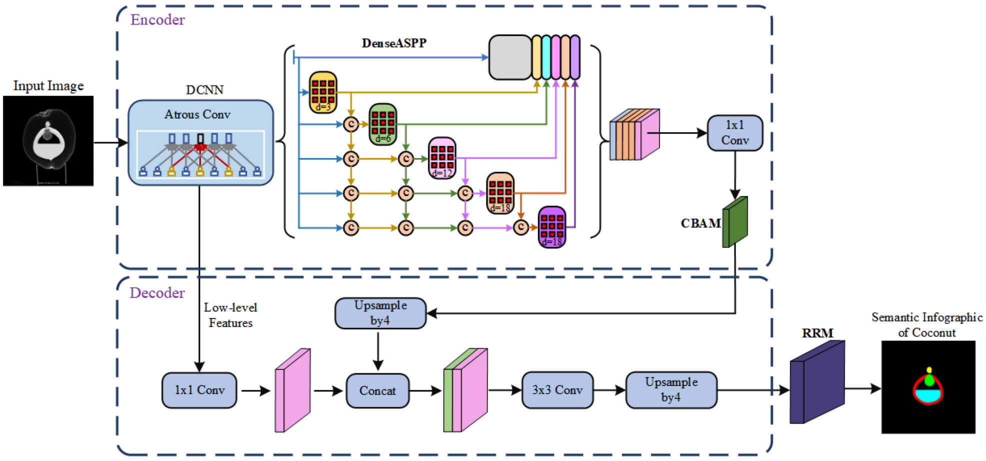


FIGURE 5
Diagram of the improved model structure.

of various scales. The structure of the module is illustrated in Figure 6.

3.3.2 CBAM attention mechanism

CBAM is a lightweight and versatile module for feed-forward convolutional neural networks. It concentrates attention resources more on the key target areas in coconut image, allocating different weights to information and background. It enhances the network's expressive power without significantly affecting its efficiency and facilitates information propagation. CBAM consists of two main parts: Channel Attention Module and Spatial Attention Module. The input features pass through the Channel Attention Module and the Spatial Attention Module sequentially, resulting in recalibrated features.

In the Channel Attention Module, both average pooling and max pooling are applied to the features. The pooled features are then fed into a shared multi-layer perceptron with shared weights. The output of the MLP is multiplied element-wise with the original feature map after a sigmoid operation. In the Spatial Attention Module, the feature map outputted by the Channel Attention Module serves as the input. Two pooling operations are performed along the channel dimension, resulting in feature maps of size h * w * 1 each time. The feature maps from the two poolings are then concatenated along the channel dimension, resulting in a feature map of size h * w * 2. This feature map undergoes a convolution operation with a kernel size of 7 * 7 and a convolutional kernel count of 1 (channel compression). The result is then passed through a sigmoid function and finally subjected to matrix multiplication. The working principle of the entire CBAM module is illustrated in Figure 7.

3.3.3 RRM module

The Residual Refinement Module (RRM) is a commonly used module in deep neural networks that incorporates the idea of an excellent encoder-decoder architecture (Qin et al., 2019). Its main purpose is to refine the details in the optimized results that deviate from the ground truth by learning to integrate features from both high and low layers. The RRM consists of four stages each for the encoder and decoder. Each stage involves a convolution operation to extract image features. Each layer has a set of 64 3×3 convolutional filters to capture specific feature information. Batch normalization and ReLU activation functions are applied after each convolution. The bridge connection layer follows the same structure.

Upon receiving the fused feature map from the original network's decoder, the encoder utilizes non-overlapping max pooling for downsampling to preserve global texture information. The decoder employs up-sampling with bilinear interpolation to restore the fine features to the original size. Finally, the module outputs the result of the saliency feature map. This design enables the continuous capture of detailed information at different scales and enhances the completeness of boundary semantic features. The structure of RRM is depicted in Figure 8.

3.4 CT image segmentation method based on improved Deeplab V3+ network

After making improvements to the network model, and based on the established dataset, the two main components are integrated into the entire segmentation method. The logical flow of the process is designed as shown in Figure 9. The diamond boxes represent the results obtained before and after algorithm training and testing, while the rectangular boxes represent the operations during the training and testing process.

A self-built dataset of coconut CT images is used, including the original images and the corresponding ground truth segmentation images. The types and quantities of images can be selected and divided into training and testing sets as needed. For network model training, the original coconut CT images are used as inputs to the entire model, with the ground truth segmentation images as the supervision. The training process is end-to-end. After training, the improved Deeplab V3+ model for coconut CT image segmentation is obtained.

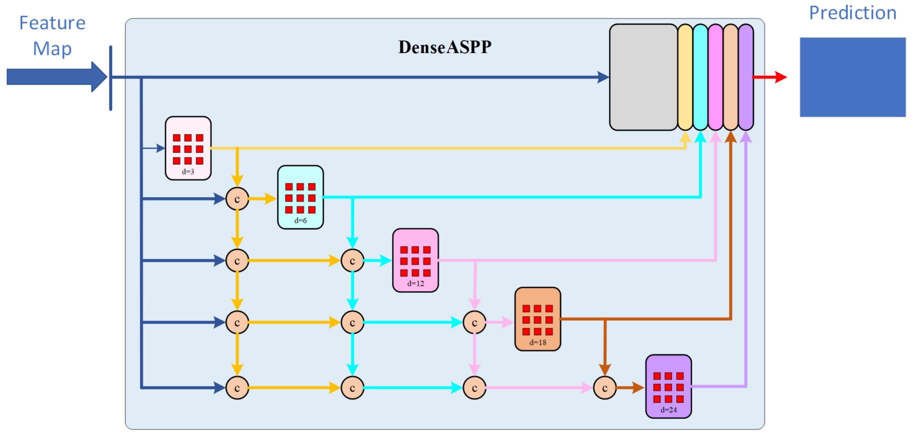


FIGURE 6
Diagram of DASPP module.

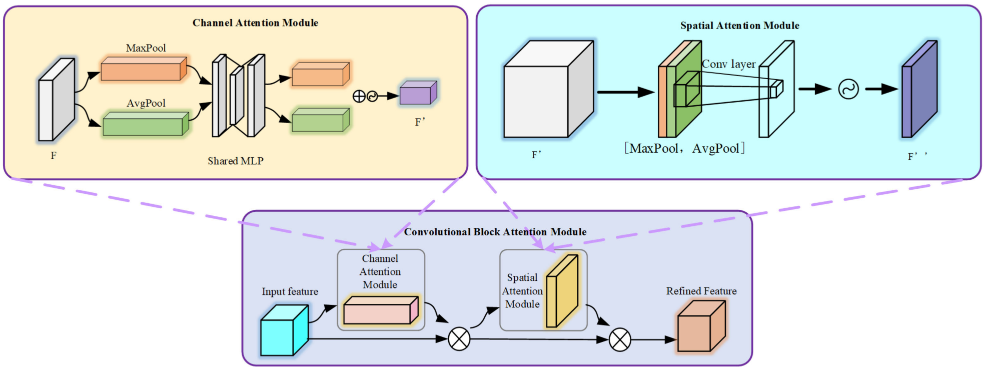


FIGURE 7
The structure of CBAM attention mechanism.

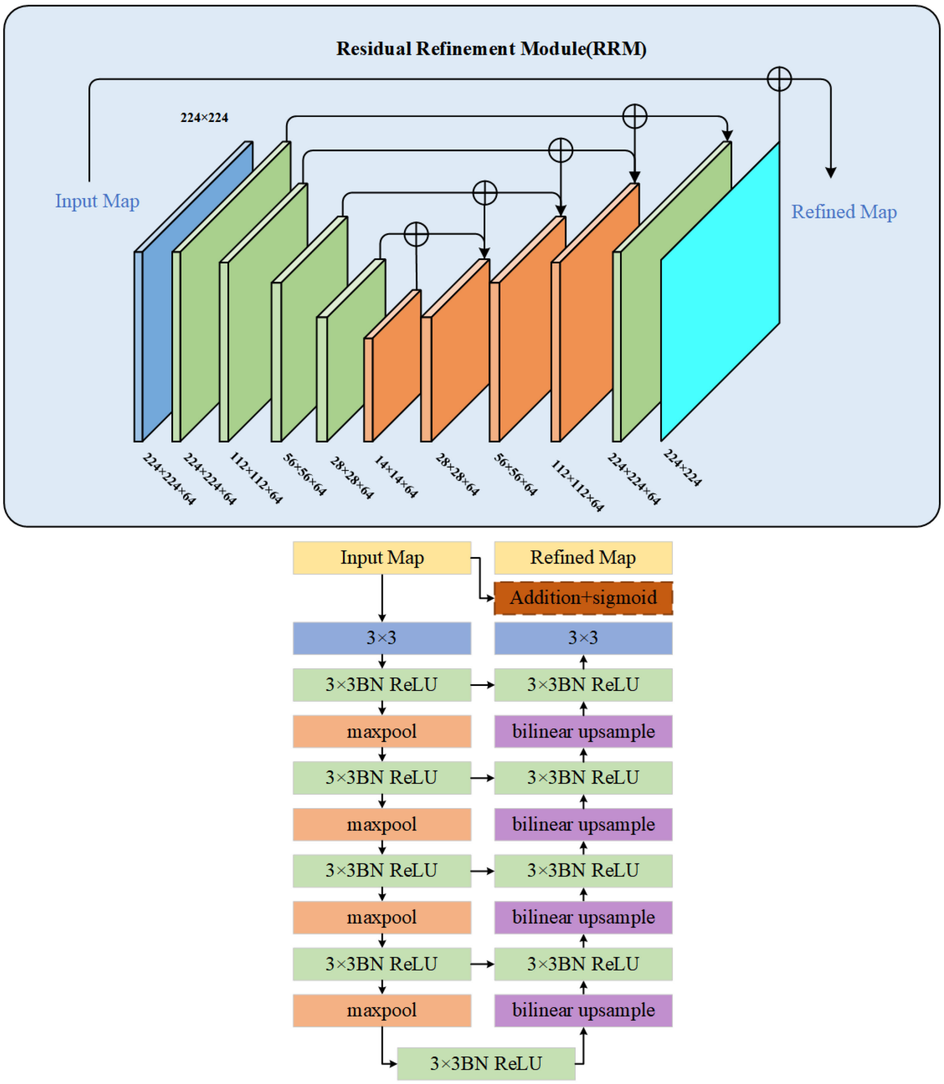


FIGURE 8
Structure diagram of RRM module.

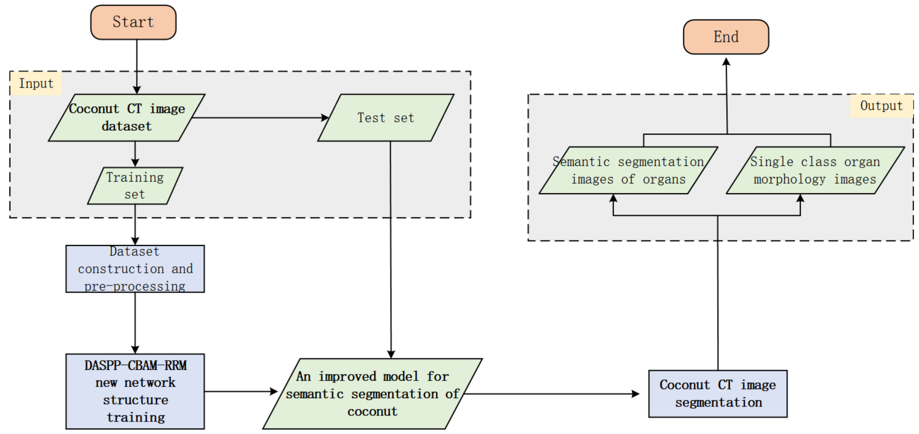

Start
Input
Coconut CT image dataset
Test set
Training set
Dataset construction and pre-processing
DASPP-CBAM-RRM new network structure training
An improved model for semantic segmentation of coconut
Coconut CT image segmentation
Semantic segmentation images of organs
Single class organ morphology images
Output
End

FIGURE 9
Flow chart of CT image segmentation based on improved Deeplab V3+ network.

Then, in the segmentation stage, a given original coconut CT image from the testing set is used. With the trained improved segmentation model, specific target organs can be segmented from the image. If it is necessary to view a specific organ separately, the pixel color values can be traversed to detect and extract the target region. Since the CT scanner has a fixed scale set when generating slice images, the values obtained from the semantic image can be transformed according to the scale to obtain actual quantified data of the target organ. Subsequently, segmentation experiments and validations will be conducted using this method.

4 Experiment

4.1 Experimental environment

The CT images were captured using a dual-source CT scanner (Somatom Definition Flash, Siemens, Germany). Each time, the coconut was placed uniformly with the top facing upwards and the bottom placed on a fixed mold. They were sequenced according to the month of growth, and positions were marked with a marker on both the fixed mold and the coconut to ensure data uniformity and completeness throughout long-term scanning. The CT scan parameters were as follows: slice thickness/increment = 0.6mm/75%, tube voltage 120kV, tube current 250mAs, field of view (FOV) 400mm×400mm, gantry rotation speed 0.5s/rotation.

Model training was conducted on a Dell workstation with the Ubuntu 20.04 operating system. It includes 24G of video memory, an RTX3090 graphics card, an Inter i7 CPU, and was developed on the Pycharm platform. The version of Pytorch used was torch1.10, with cuda version 11.4. The model was trained using our own constructed Coconut CT Imaging Dataset (CIDCO).Since the previously established coconut CT dataset was categorized and stored in separate folders according to coconut variety and growth stage, to ensure comprehensive training data, images were randomly selected from each category. Five categories were chosen for semantic segmentation: absorber, solid endosperm, liquid endosperm, embryo, and background. Due to the large differences in the internal organs of coconuts at different developmental stages, some organ categories were missing.Taking into account the prevention of an excessive number of images with the same stage and same features, in order to maintain a relatively balanced number of categories in the experimental dataset, the number of pictures containing various organs was adjusted flexibly. In the end, a total of 1470 images were confirmed as experimental data and were divided into a training set and a test set at a ratio of 8:2.

4.2 Training parameters and evaluation metrics

The improved semantic segmentation algorithm adopts a fully supervised learning approach during training. All methods are conducted on the same hardware. The hardware environment for this experiment consists of a workstation based on a 64-bit Ubuntu 20.04 operating system, Intel i7-1050H CPU, 16GB of RAM, 24G of video memory, and an NVIDIA GeForce GTX3090 graphics card. The software environment includes the Pytorch 1.1.0 framework, CUDA version 11.4, Python 3.6, and the Pycharm development platform. The input images are uniformly adjusted to a size of 256×256 pixels. The hyperparameters for the training of the coconut CT image segmentation model are as follows: The Adam optimizer is used with a learning rate of 0.0001, a training batch size of 4, momentum set to 9, a weight in the loss function of 0.7, and the loss function being a combination of Dice loss and focal loss. The total number of training epochs is set to 150.

To validate the effectiveness and robustness of the improved network model, we use IoU (Intersection over Union) and PA (Pixel Accuracy) to measure the segmentation results of individual organs. mIoU (mean Intersection over Union), mPA (mean Pixel Accuracy), and F1_score are used to evaluate the model's overall semantic segmentation capability for coconut CT images. These are commonly used evaluation metrics in semantic segmentation tasks.IoU refers to the ratio of the intersection and union of the model's prediction results and actual values for a single category of a coconut organ. PA refers to the proportion of correctly predicted pixels in a single organ category to the total number of pixels. mIoU represents the average of the ratios of intersections and unions of prediction results and actual values for each category of coconut organs. mPA is calculated by first computing the *PA* for each organ class of the coconut, and then taking the average of the PA_s for all classes.F1_Score represents a comprehensive score for the correctness of the final results. Thus, the larger the value of these indicators, the better the segmentation effect of the model. Their calculation formulas are as per Equations 1–7, where *TP* represents the number of correct detections, *FP* is the number of false detections, *FN* is the number of undetected quantities, *k* represents the number of categories, p_{ii} indicates the number of correctly classified pixels; p_{ij} is the number of pixels of class *i* predicted as class *j*, Precision(i) represents the precision of class *i*, Recall(i) represents the recall rate of class *i*, and r_i represents the proportion of the number of samples of class *i* in the total samples.

$$\text{Precision} \quad = \frac{TP}{TP + FP} \qquad \text{(Eq. 1)}$$

$$\text{Recall} = \text{Sensitivity} = TPR = \frac{TP}{TP + FN} \qquad \text{(Eq. 2)}$$

$$F1\_S\,\text{core} = \frac{2^{*}\ \text{Precision} \ * \ \text{Recall}}{\text{Precision} \ + \ \text{Recall}} \qquad \text{(Eq. 3)}$$

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}} \qquad \text{(Eq. 4)}$$

$$mPA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}} \qquad \text{(Eq. 5)}$$

$$IoU = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\left(\sum_{j=0}^{k} p_{ji} + \sum_{j=0}^{k} p_{ij}p_{ii}\right)} \qquad \text{(Eq. 6)}$$

$$mIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \qquad \text{(Eq. 7)}$$

4.3 Ablation study and model comparison

4.3.1 Module ablation study

To verify the effectiveness of our proposed improvements, we designed an ablation study in which we run the model on the same dataset, subtracting one of the three modules from the improved model. 'All' represents the complete modules that we have added. The training process uses the same parameter configuration, and the final results are shown in Table 1.

According to the data in the table, the network structure improved by the three modules shows the best overall performance. When focusing on individual organs, the improved new network has a higher pixel accuracy than the other comparative modules. When faced with complete organ images showing different features, the model's mIoU, mPA, and F1_Score all outperform structures missing a module. For the task of semantic segmentation of coconut organs, focusing on the entire target area's features and supplementing with local boundary information is the optimal solution. Thus, it is confirmed that this point of improvement can significantly enhance the robustness and accuracy of the segmentation method.

4.3.2 Comparison of segmentation results from different models

In the same dataset, we compare our proposed model with commonly used segmentation models to verify our model's excellent segmentation capability. We selected five models, namely Basnet, Unet, Transfuse, MANet, and Deeplab v3+, using IoU, PA, mIoU, mPA, and F1_Score as evaluation metrics. We compare and analyze the results from both qualitative and quantitative perspectives, as shown in Figure 10 and Table 2.

From Table 2, it is clear that the improved model performs better than the majority of models in terms of Intersection over Union (IoU) and Pixel Accuracy (PA) when facing segmentation of individual organ classes. This is especially apparent for liquid endosperm and embryos. Other models are only comparable to the improved model in one or two data points. For the semantic segmentation of the entire image, the improved model has a clear advantage in terms of mean Intersection over Union (mIoU), mean Pixel Accuracy (mPA), and F1_Score. These three metrics show that the values have improved compared to the comparison models, proving the effectiveness of the improvement method proposed in this chapter. Apart from quantitative results,

TABLE 1 Module ablation data table.

Keep the module	Background		Solid Endosperm		Embryo	Haustorium		Liquid Endosperm		mIoU	mPA	F1_Score
	IoU	PA	IoU	PA	IoU PA	IoU	PA	IoU	PA			
DASPP+CBAM	0.99	0.99	0.82	0.92	0.74 0.85	0.84	0.88	0.71	0.93	82.46	91.86	90.09
RRM+CBAM	0.99	0.99	0.82	0.92	0.75 0.85	0.85	0.90	0.72	0.91	82.99	92.02	90.43
DASPP+RRM	0.99	0.99	0.82	0.92	0.67 0.72	0.85	0.90	0.62	0.94	79.43	89.98	87.93
ALL(D+C+R)	0.99	0.99	0.82	0.93	0.75 0.85	0.84	0.89	0.72	0.92	83.10	92.05	90.50

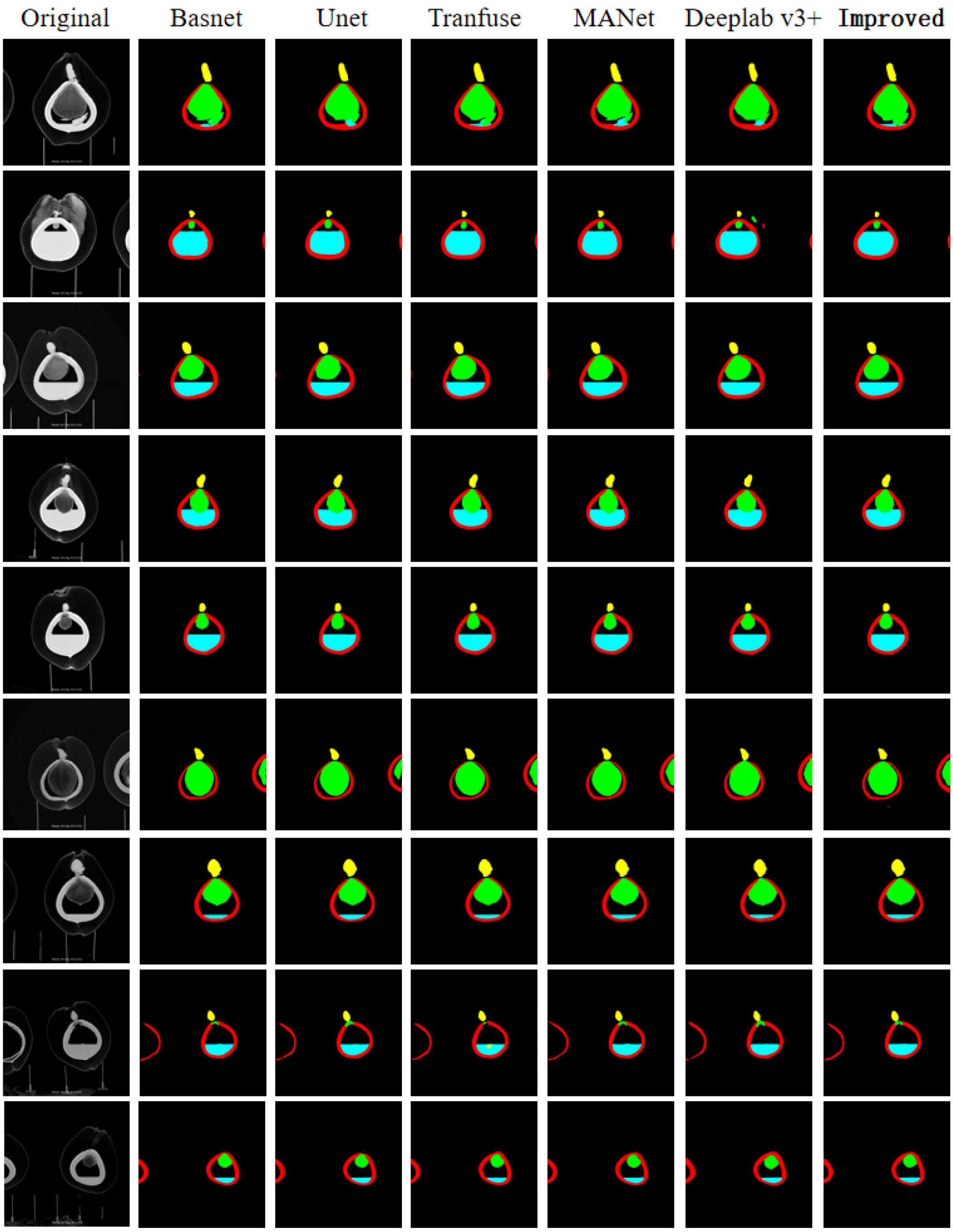


FIGURE 10
Semantic segmentation effect of different models.

Figure 10 shows the segmentation effects of each model at the image level, demonstrating that the improved model still has a higher accuracy in segmentation at a qualitative level.

4.4 Organ extraction

Considering that in actual scenarios, it may be necessary to extract a particular organ for analysis, we set up an organ extraction and data quantification section. After inputting the images to be operated on into the model, we obtain the semantic images of coconuts. We then create a corresponding number of blank images of the same size, traverse all pixels in the semantic image, and follow the principle of point-to-point correspondence in the target organ based on the RGB value in the semantic image to make the corresponding points in the blank image the same value. This way, we can obtain the image of the target organ alone. In terms of determining the growth and development quality of the coconut, quantitative data of the organs is one of the reference pieces of information, in addition to making judgements in the form of two-

TABLE 2 Model comparison table.

Network model	Background		Solid Endosperm		Embryo		Haustorium		Liquid Endosperm		mIoU	mPA	F1_Score
	IoU	PA	IoU	PA	IoU	PA	IoU	PA	IoU	PA			
Basnet	0.99	0.99	0.84	0.93	0.39	0.41	0.84	0.89	0.48	0.93	71.30	83.69	81.00
Unet	0.99	0.99	0.82	0.91	0.46	0.50	0.85	0.89	0.51	0.91	72.80	84.57	82.55
Tranfuse	0.99	0.99	0.83	0.92	0.46	0.54	0.84	0.89	0.55	0.88	73.94	84.92	83.49
MANet	0.99	0.99	0.83	0.92	0.65	0.76	0.85	0.90	0.74	0.94	81.75	90.82	89.54
Deeplab v3+	0.99	0.99	0.79	0.92	0.65	0.70	0.84	0.90	0.64	0.88	78.47	88.34	87.36
Improved model	0.99	0.99	0.82	0.93	0.75	0.85	0.84	0.89	0.72	0.92	83.10	92.05	90.50

dimensional images. Whether it's the complete semantic image of the coconut or a particular organ that has been extracted, data can still be obtained through the RGB value of the pixel points. For example, the height of the embryo can be determined because, in the semantic image, the embryo is characterized by the color green. One can start from the top of the image and gradually traverse downwards in the form of a horizontal line. When the RGB value of a pixel point becomes (0, 255, 0), it is marked as point A. Then, using the same method, traverse from the bottom of the image upwards, and when you encounter a pixel point with the same value, mark it as point B. The distance between points A and B is the height of the embryo. When dealing with an embryo with a significant curvature, it can be rotated to be relatively parallel to the y-axis, and then the point traversal method can be used. Figure 11 shows an example of the extracted image results.

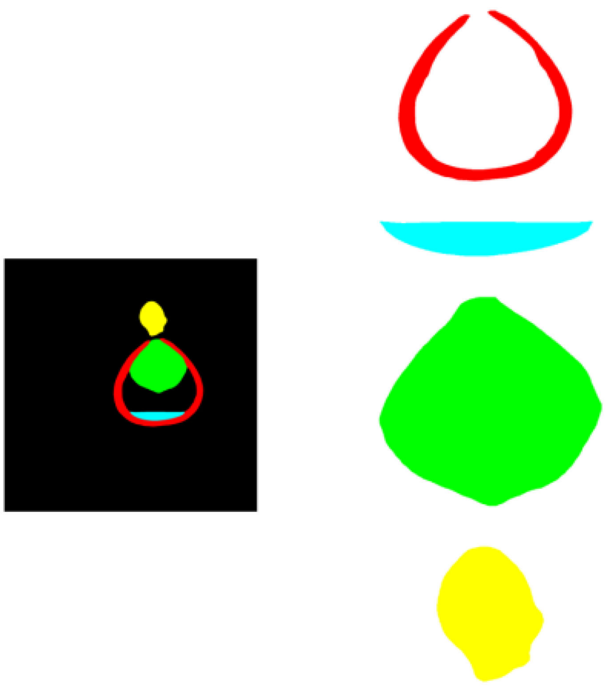


FIGURE 11
Example of single class organ extraction.

5 Conclusion and prospects

This chapter starts from the perspective of the black box phenomenon present in the development process of the coconut fruit. We used CT non-destructive observation to acquire images of coconuts at various stages and of various varieties, thus establishing a CT image dataset for coconuts. This work fills the gap in image resources for coconuts. On this basis, we addressed the issue of traditional semantic segmentation models not performing well on coconut CT images. We replaced the original Atrous Spatial Pyramid Pooling (ASPP) block with a Dense Atrous Spatial Pyramid Pooling (DASPP) module, resolving information loss due to sparse sampling. Then, we added the Convolutional Block Attention Module (CBAM) to the network, enabling it to better capture the features of coconut organs and reduce the interference of irrelevant redundant information. Finally, a residual refinement module was embedded after the decoder to enhance the boundary information between closely connected organs. This allows the network to acquire richer global feature information and optimize boundary details, thereby improving the semantic segmentation accuracy of coconut CT images. During the model training process, we used multi-state feature coconut images to improve the model's robustness. Finally, detailed model comparisons and ablation experiments were carried out. The results of the evaluation indicators and the semantic segmentation effect images both quantitatively and qualitatively demonstrate the improved model's high-precision segmentation ability on coconut CT images. Furthermore, individual organ morphology and quantitative data can be obtained from the semantic segmentation images to increase reference information during the development process of the coconut. This is beneficial in assisting decision-makers to make scientific judgments on the development status and growth stage of the coconut.

In our future research work, we will analyze the high-precision organ morphology and quantitative data obtained from the segmentation model to further mine the laws of coconut growth and development. At the same time, we will incorporate image morphology changes to construct a visualized standard development process for the coconut, thereby making more precise predictions of coconut intelligent development. Furthermore, we aim to deploy our model on mobile devices to provide more reference information and decision support for optimizing coconut breeding. This will aid coconut cultivators in better managing their cultivation practices, with the goal of achieving and continuously surpassing targets for high yield and high-quality coconuts.

Data availability statement

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

Author contributions

QL was in charge of building the modified Deeplab V3+ model and writing the paper, YZ was in charge of the model comparison and ablation experiments. JC and CS were in charge of building the data set, JC was in charge of scanning the CT images, CS was in charge of providing all kinds of coconut. MH was in charge of the coordination of the whole workflow and thesis guidance. MC and CL were in charge of the image tagging. SL was responsible for assisting JC to perform CT scans and record data. All authors contributed to the article and approved the submitted version.

Funding

This work was supported in part by Major Science and Technology Project of Haikou (Grant: 2020-009), in part by the Key R&D Project of Hainan province (Grant: ZDYF2021SHFZ243), in part by National Natural Science Foundation of China (Grant: 62062030).

Acknowledgments

The author thanks the experts from Coconut Research Institute, Hainan University and Haikou Municipal for their help and experience sharing in the whole work, I am also grateful to the reviewers and editors for their valuable comments on improving this paper.

Conflict of interest

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

Publisher's note

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

References

Arumugam, T., and Hatta, M. A. M. (2022). Improving coconut using modern breeding technologies: Challenges and opportunities. *Plants* 11, 3414. doi: 10.3390/plants11243414

Li, B., Wang, R., Ma, J., and Xu, W. (2020). Research on crop water status monitoring and diagnosis by terahertz imaging. *Front. Phys.* 8. doi: 10.3389/fphy.2020.571628

Ni, X., Li, C., Jiang, H., and Takeda, F. (2020). Deep learning image segmentation and extraction of blueberry fruit traits associated with harvestability and yield. *Horticulture Res.* 7, 110. doi: 10.1038/s41438-020-0323-3

Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., and Jagersand, M. (2019). Basnet: Boundary-aware salient object detection. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* (Long Beach, CA, USA). 7479–7489. doi: 10.1109/CVPR.2019.00766

Singh, N., Tewari, V., Biswas, P., Dhruw, L., Pareek, C., and Singh, H. D. (2022). Semantic segmentation of in-field cotton bolls from the sky using deep convolutional neural networks. *Smart Agric. Technol.* 2, 100045. doi: 10.1016/j.atech.2022.100045

Suk, H.-I., Liu, M., Cao, X., and Kim, J. (2023). Advances in deep learning methods for medical image analysis. *Front. Radiol.* 2. doi: 10.3389/fradi.2022.1097533

Sun, K., Wang, X., Liu, S., and Liu, C. (2021). Apple, peach, and pear flower detection using semantic segmentation network and shape constraint level set. *Comput. Electron. Agric.* 185, 106150. doi: 10.1016/j.compag.2021.106150

Thorp, K., and Dierig, D. (2011). Color image segmentation approach to monitor flowering in lesquerella. *Ind. Crops Products* 34, 1150–1159. doi: 10.1016/j.indcrop.2011.04.002

Turgut, K., Dutagaci, H., and Rousseau, D. (2022). Rosesegnet: An attention-based deep learning architecture for organ segmentation of plants. *Biosyst. Eng.* 221, 138–153. doi: 10.1016/j.compag.2018.09.034

Van De Looverbosch, T., Raeymaekers, E., Verboven, P., Sijbers, J., and Nicolaï, B. (2021). Non-destructive internal disorder detection of conference pears by semantic segmentation of x-ray ct scans using deep learning. *Expert Syst. Appl.* 176, 114925. doi: 10.1016/j.eswa.2021.114925

Xiang, R. (2018). Image segmentation for whole tomato plant recognition at night. *Comput. Electron. Agric.* 154, 434–442. doi: 10.1016/j.compag.2018.09.034

Yu, L., Liu, L., Yang, W., Wu, D., Wang, J., He, Q., et al. (2022). A non-destructive coconut fruit and seed traits extraction method based on micro-ct and deeplabv3+ model. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.1069849

Zhang, Y., Liu, Q., Chen, J., Sun, C., Lin, S., Cao, H., et al. (2023). Developing non-invasive 3d quantificational imaging for intelligent coconut analysis system with x-ray. *Plant Methods* 19, 1–11. doi: 10.1186/s13007-023-01002-4

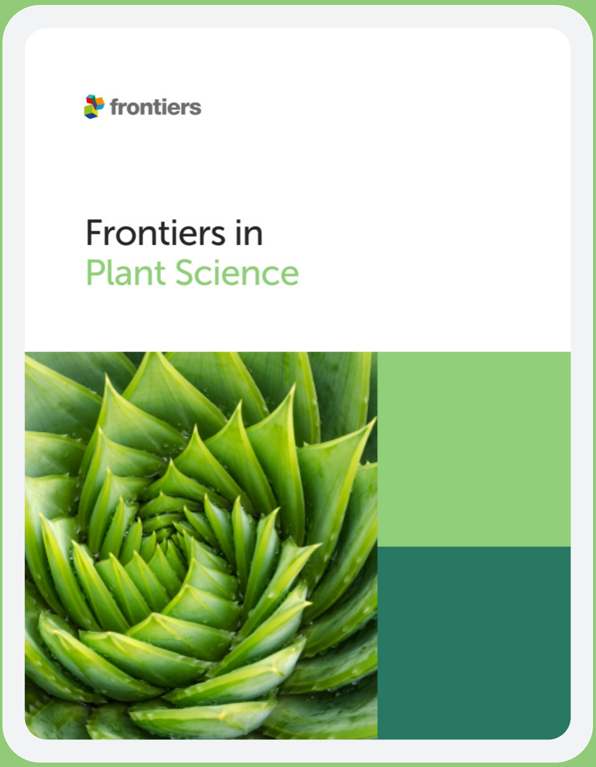
Frontiers in
Plant Science

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

Frontiers in
Plant Science

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact