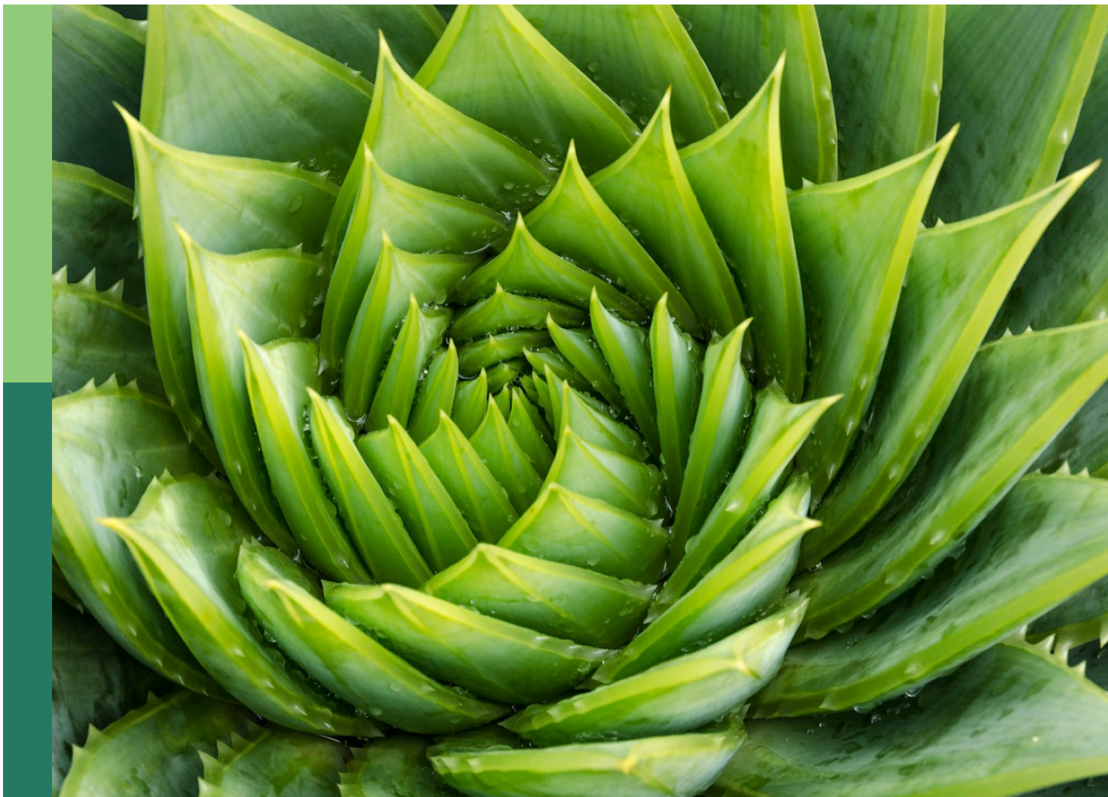
frontiers | Research Topics

Innovative approaches to agricultural plant disease identification: integrating deep learning into traditional methods

Edited by
Yongliang Qiao, Chunlei Xia and Meili Wang

Published in
Frontiers in Plant Science



ISSN 1664-8714
ISBN 978-2-8325-6055-6
DOI 10.3389/978-2-8325-6055-6

About Frontiers

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

Frontiers journal series

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

Dedication to quality

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

What are Frontiers Research Topics?

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

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

Innovative approaches to agricultural plant disease identification: integrating deep learning into traditional methods

Topic editors
Yongliang Qiao — University of Adelaide, Australia
Chunlei Xia — Chinese Academy of Sciences (CAS), China
Meili Wang — Northwest A&F University, China

Citation
Qiao, Y., Xia, C., Wang, M., eds. (2025). *Innovative approaches to agricultural plant disease identification: integrating deep learning into traditional methods*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-6055-6

Table of contents

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 09 October 2023
DOI 10.3389/fpls.2023.1255119

Tomato disease object detection method combining prior knowledge attention mechanism and multiscale features

Jun Liu* and Xuewei Wang*

Shandong Provincial University Laboratory for Protected Horticulture, Weifang University of Science and Technology, Weifang, China

OPEN ACCESS

EDITED BY
Lei Shu,
Nanjing Agricultural University, China

REVIEWED BY
Jun Yan,
China Agricultural University, China
Jun Su,
Fujian Agriculture and Forestry University, China

*CORRESPONDENCE
Jun Liu
✉ liu_jun860116@wfust.edu.cn
Xuewei Wang
✉ wangxuewei@wfust.edu.cn

RECEIVED 10 July 2023
ACCEPTED 21 September 2023
PUBLISHED 09 October 2023

CITATION
Liu J and Wang X (2023) Tomato disease object detection method combining prior knowledge attention mechanism and multiscale features. *Front. Plant Sci.* 14:1255119.
doi: 10.3389/fpls.2023.1255119

To address the challenges of insufficient accuracy in detecting tomato disease object detection caused by dense target distributions, large-scale variations, and poor feature information of small objects in complex backgrounds, this study proposes the tomato disease object detection method that integrates prior knowledge attention mechanism and multi-scale features (PKAMMF). Firstly, the visual features of tomato disease images are fused with prior knowledge through the prior knowledge attention mechanism to obtain enhanced visual features corresponding to tomato diseases. Secondly, a new feature fusion layer is constructed in the Neck section to reduce feature loss. Furthermore, a specialized prediction layer specifically designed to improve the model's ability to detect small targets is incorporated. Finally, a new loss function known as A-SIOU (Adaptive Structured IoU) is employed to optimize the performance of the model in terms of bounding box regression. The experimental results on the self-built tomato disease dataset demonstrate the effectiveness of the proposed approach, and it achieves a mean average precision (mAP) of 91.96%, which is a 3.86% improvement compared to baseline methods. The results show significant improvements in the detection performance of multi-scale tomato disease objects.

KEYWORDS

complex background, tomato diseases, prior knowledge, attention mechanism, multi-scale features, object detection

1 Introduction

Due to the ongoing expansion of tomato cultivation areas and limited arable land, a growing contradiction has emerged between the two. As a result, consecutive cropping of tomatoes has become prevalent, resulting in an increase in the variety and complexity of tomato diseases. According to relevant studies, there are currently more than thirty types of fungal diseases alone affecting tomatoes worldwide (Widjaja et al., 2022). In China, there are several prevalent and influential tomato diseases that significantly impact tomato cultivation. These include early blight, late blight, bacterial spot, gray leaf spot, gray mold, leaf mold, yellow leaf curl virus, mosaic virus, canker, and anthracnose (Liu and Wang, 2020).

Tomato diseases have become a prominent issue in China, leading to a reduction in yield of approximately 10%. In areas severely impacted by these diseases, complete crop failure has been observed (Thangaraj et al., 2022). Tomato diseases not only result in a reduction in tomato yield but also pose risks to storage and transportation due to the contamination of infected fruits. As a result, the efficient diagnosis and control of tomato diseases have emerged as critical concerns in tomato production.

During the early stages of tomato diseases, farmers often neglect to assess and manage these diseases due to their unclear symptoms. This oversight frequently results in missing the optimal period for disease prevention and control. As the tomato diseases progress and become severe, the application of a large amount of fungicides proves to be ineffective. Another group of farmers faces challenges in assessing whether their tomatoes are infected and lacks the ability to distinguish the severity of the diseases. Consequently, they resort to extensively using fungicides for disease prevention and control. Unfortunately, prolonged implementation of such practices leads to the excessive use of fungicides, posing risks to environmental safety and human health (Moussafir et al., 2022). Therefore, there is an increasing demand for timely and effective identification, detection, and precise application of treatments for tomato diseases, making it a prominent research topic in recent years.

Through the long-term collaborative efforts of agricultural and plant protection scholars, notable advancements have been made in the domain of tomato disease control and prevention in China. Commonly employed methods include empirical analysis based on observable symptoms and physicochemical analysis. However, when it comes to large-scale detection, the limited number of experts hinders their ability to provide real-time monitoring of tomato diseases across the entire production line. Additionally, expert judgments may be swayed by various influential elements, including weather conditions and theoretical knowledge, making it challenging to timely and accurately assess the occurrence of tomato diseases in actual production. Moreover, the physicochemical analysis of tomato diseases requires a significant number of specialized technicians, is time-consuming, and poses the risk of secondary transmission of diseases due to human activities. Consequently, there is an urgent need to explore and develop rapid, accurate, non-destructive, and environmentally-friendly methods for detecting tomato diseases, which has become a key research focus.

The development of modern computer technology has led to increasingly refined applications of new artificial intelligence information in agriculture. Over the course of more than 30 years of progress in artificial intelligence, intelligent diagnosis has been implemented in various aspects of crop cultivation management, plant protection, crop breeding, and agricultural planting decisions (Misra et al., 2020). These advancements have greatly enhanced the efficiency and accuracy of agricultural practices. Additionally, the integration of artificial intelligence and image recognition enables rapid, accurate, and non-destructive identification and diagnosis of diseases. Image detection primarily relies on cameras and other devices to capture information on crop diseases, thereby reducing the need for human observation (Mohammad-Razdari et al., 2022). By leveraging digital image processing, healthy and diseased crops can be identified and classified accurately.

However, there is still significant room for improvement in the actual tomato disease detection process, as the current detection accuracy and algorithm processing speed do not meet the requirements of real-world farming scenarios (David et al., 2021) (Karthik et al., 2020). Several challenges contribute to this limitation. Firstly, there is an imbalance in the number of samples available for different tomato diseases (Abbas et al., 2021). This scarcity of samples makes it difficult to obtain an adequate representation of various diseases, which in turn hampers model training and severely restricts the learning capacity of deep learning models. Secondly, tomato disease detection possesses unique characteristics. The natural background of tomato diseases is complex and diverse, and different types of diseases exhibit distinct characteristics (Gonzalez-Huitron et al., 2021). Even with a sufficient number of tomato disease samples, relying solely on visual features makes accurate identification challenging (Huang et al., 2023). In contrast, humans possess the ability to quickly learn and assimilate new knowledge based on their accumulated experiences, which is referred to as prior knowledge. This suggests that incorporating prior knowledge of tomato diseases into tomato disease detection is essential to enhance learning efficiency (Diligenti et al., 2017). Therefore, it is crucial to integrate deep learning models with prior knowledge in the field of tomato disease detection in order to overcome these challenges.

Applying existing deep learning models directly to tomato disease detection tasks makes it challenging to accurately differentiate the distinctive features of different diseases. This limitation often leads to a significant number of misclassifications or omissions. Consequently, the integration of deep learning models with prior knowledge and the improvement of tomato disease detection accuracy through a collaborative "data model knowledge" approach have become common challenges faced by both the agricultural and academic communities. To address the lack of explicit expression of objective prior knowledge in deep learning models and the imbalanced distribution of disease samples, this research aims to combine deep learning models with disease prior knowledge. The study focuses on tomato diseases occurring in complex backgrounds, considering the complexity of tomato disease data and utilizing prior knowledge. As a result, a tomato disease object detection method that integrates a prior knowledge attention mechanism and multi-scale features is proposed.

2 Related work

2.1 Object detection

Object detection technology encompasses multi-object classification and localization as its primary tasks. It is not only responsible for determining whether the detection area contains target objects but also for marking these targets with bounding boxes. Over the past few years, the remarkable and swift progress in the field of computer technology has been noteworthy, coupled with advancements in convolutional neural networks, has significantly propelled object detection technology forward. As a result, it has found extensive applications in diverse fields including traffic monitoring and tracking, video surveillance and security alert systems, drone scene analysis, and robotic vision (Zou et al., 2023).

Object detection technology can be categorized into traditional approaches and those based on deep learning. Traditional approaches are founded upon the dependence of manual feature extraction and conventional classifiers for object classification. However, they are plagued by problems like weak feature representation, suboptimal accuracy and inadequate real-time performance. In contrast, the rapid growth of big data and computing hardware has led to the widespread adoption and acceptance of deep learning in the field of object detection. It excels in feature extraction, which brings notable benefits such as enhanced detection accuracy and accelerated processing speed. As a result, it is gradually emerging as the dominant technology in the field of computer vision (Zhao et al., 2019). This technology has shown great potential in various applications, including tomato disease detection, where accurate and efficient identification of diseases is crucial for effective disease management in agriculture.

2.2 Plant disease object detection method in laboratory environment

Zhang et al. (2020) developed an enhanced iteration of the Faster-RCNN algorithm, specifically tailored for the identification of healthy tomato leaves and the detection of four different diseases. Instead of using VGG16, they utilized ResNet101 as the feature extractor. The experimental findings substantiated that the enhanced detection approach yielded a 2.71% increase in accuracy, while also providing faster detection speed. Wang et al. (2021) conducted experiments using the PlantVillage dataset and found that the DBA_SSD algorithm outperformed other object detection algorithms. However, It is important to highlight that the images employed in these studies was primarily captured in controlled laboratory environments. In such environments, the samples benefitted from sufficient lighting, simple and uniform backgrounds, and carefully controlled shooting angles. Moreover, agricultural experts screened and annotated the samples, resulting in more distinct disease features. In contrast, images collected in natural environments are significantly more complex. Various uncontrollable factors such as environmental location, weather conditions, and shooting angles pose challenges, including uneven lighting, shadow occlusion, overlapping leaves, and complex backgrounds (Liu and Wang, 2021). Consequently, object detection models trained solely under laboratory conditions are inadequate for real-world natural environments and fail to fully meet the production needs of farmers. The performance of these models can be affected by various factors such as lighting conditions, variations in plant appearance, and diverse backgrounds in the field. Therefore, it is crucial to train object detection models using datasets that encompass a wide range of real-world scenarios, including different weather conditions, growth stages, and farming practices.

2.3 Object detection method for plant diseases in real natural environments

In real natural environments, the complexity of the image sample backgrounds adds to the difficulty of the detection task. Training an effective deep learning model for disease object detection necessitates a substantial amount of data. Consequently, this task has garnered considerable attention and become a significant challenge in current research endeavors.

Fuentes et al. (2017) conducted fine-tuning of classical models using transfer learning on a self-built dataset of tomato diseases. After thorough analysis, they selected R-FCN with ResNet-50. This particular configuration achieved an impressive average precision (AP) of 85.98% and effectively recognized nine different diseases. In their study, Xu et al. (2022) presented a real-time technique for detecting diseases on cucumber leaves. In order to boost the model's performance, channel pruning was utilized to trim and fine-tune a sparsely trained model, and the pruned YOLO v5s+Shuffle model was then deployed on the Jetson Nano platform, achieving a remarkable mean average precision (mAP) of 96.7%. Zhang et al. (2021) developed a multi-feature fusion Faster R-CNN to accurately detect diseases on soybean leaves. Their approach yielded a best average precision of 83.34%, showcasing the effectiveness of their design. Chen et al. (2022) developed an improved plant disease identification model based on the original YOLOv5 network model with an average accuracy of 70%. Roy et al. (2022) put forward an exceptional-performance framework for real-time detection of fine-grained objects. Their framework achieved successful detection of diseases across diverse and challenging environmental conditions.

Taking inspiration from attention mechanisms (Vaswani et al., 2017), some research studies have enhanced feature extraction by incorporating attention mechanisms. For example, Qi et al. (2022) put forward an enhanced network model, SE-YOLOv5 for the identification of tomato virus diseases, which resulted in an average precision (mAP) of 94.10%. In another study, Guo et al. (2022) presented a CST model based on the Swin Transformer. This model employed a novel convolution design and achieved accuracies of 0.909 and 0.922. Furthermore, Thai et al. (2023) introduced FormerLeaf. Their contribution was the proposal of attention pruning. This algorithm achieved a reduction in model size by 28%, an evaluation speed acceleration by 15%, and an approximate 3% improvement in accuracy.

Furthermore, the contextual information captured during the recording of plant disease images contributes to more accurate category classification by the model. Wang et al. (2020) introduced a context-aware attention model which encodes various types of information, such as image context, geographical information, time information, and environmental information, into image annotations. They utilized a multi-task learning architecture with CNN models for each task to extract features related to pest coarse classification, geography, time, and environment. This algorithm surpasses traditional image feature extraction by incorporating external environmental factors like geography, time, and environment into the process. By extracting features relevant to pest habitat, it explores the possibility of integrating a wide range of environmental information into CNN for feature representation.

Zhao et al. (2020) developed the Multi-Context Fusion Network (MCFN), which leverages contextual features extracted from image sensors as prior knowledge. This introduction resulted in highly accurate predictions of crop diseases. Cheng et al. (2023) proposed a Position Attention Block that effectively extracts positional information from feature maps and constructs attention maps to bolster the feature extraction ability. These efforts aim to enhance the performance and applicability of disease object detection models in real-world agricultural settings.

2.4 Technical challenges of plant disease detection methods

Compared to earlier studies on plant disease detection algorithms, the methods mentioned above have significantly improved detection performance. However, they still encounter various obstacles that pose challenges to accurate detection. These challenges include four categories (Figure 1). One of them is intra-class variation, where different instances of the same disease may exhibit variations in appearance and symptoms. Inter-class resemblances refer to cases where different diseases or healthy plants may share similar visual characteristics, leading to misclassification. Complications arising from low resolution images can make it difficult to discern fine details and accurately identify diseases. Additionally, occlusion and overlap of plant parts or other objects in the image can further hinder detection accuracy. It is crucial for researchers to address these challenges through advanced techniques such as data augmentation, model optimization, and incorporating contextual information to improve the robustness and reliability of plant disease detection algorithms (Thakur et al., 2022). Figure 1 illustrates these challenges visually.

To tackle the aforementioned concerns, this study proposes a method for detecting objects related to diseases in tomato plants called PKAMMF. This method integrates a prior knowledge attention mechanism and incorporates features at different scales to tackle the obstacles of dense distribution of tomato disease objects in complex backgrounds, a broad spectrum of scale variations, and lack of feature information for small objects. By combining the prior knowledge attention mechanism and multi-scale features, PKAMMF aims to improve the performance of detecting tomato disease objects.

This study makes significant contributions in the following aspects:

(1) A backbone network was proposed, which integrates a prior knowledge attention mechanism to improve the capability of extracting features and improve model stability during training on large-scale datasets.

(2) The Rep Conv convolutional layer was reparameterized in a structured manner to construct the SPPCSPF module, reducing computational and memory costs during model training.

(3) A parallel multi-branch feature fusion network was established to minimize the loss of effective information in feature maps. Additionally, to enhance the capability of detecting small objects across multiple scales, an additional layer specifically designed for small object detection was incorporated.

(4) A novel A-SIOU loss function was employed to refine and improve bounding box regression, resulting in accelerated model convergence and improved training accuracy. Experiments were carried out to evaluate the effectiveness of the proposed approach using a self-built dataset of tomato disease. The findings indicate that the proposed method surpasses the performance of mainstream algorithms in tomato disease object detection tasks with complex backgrounds.

3 Methods

Because of the intricate background conditions of tomato disease images, where the background occupies a large portion of the image while the diseased area to be detected is often small, it is necessary to use a network structure with the ability to globally model the complex nature of the background. Therefore, this study

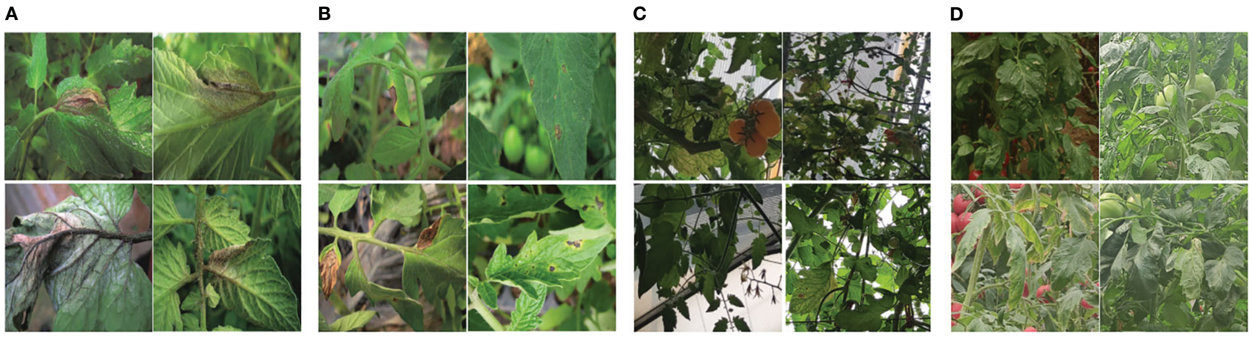


FIGURE 1
Various obstacles of plant disease detection task **(A)** intra-class discrepancies; **(B)** inter-class resemblances; **(C)** complications arising from low resolution; **(D)** occlusion overlap).

proposes a tomato disease object detection method that combines prior knowledge attention mechanism and multi-scale features. The specific improvements are described as follows:

3.1 Prior knowledge attention mechanism module

To improve disease detection and recognition, it is necessary to incorporate geographical location information, environmental parameters, and time information of tomato disease images. This is due to the inconsistent types of tomato diseases, occurrence time, surrounding environment, and geographical conditions. In this study, we propose a PKAM module that integrates the prior knowledge attention mechanism to enhance the capability of extracting target features in complex backgrounds. The framework of the PKAM module is illustrated in Figure 2.

Firstly, to encode the prior knowledge of tomato disease, we utilize the Bert model (Devlin et al., 2018). The Bert model, introduced by Google in 2018, is a language model built on a transformer encoder structure. It is specifically designed for encoding language information. In our case, the Bert model is employed to encode the prior knowledge of tomato diseases. By inputting the prior knowledge text information, the model generates an encoded prior knowledge vector K (C, T), where C denotes the maximum text length of the prior knowledge and T denotes the vector dimension. The default value for T is set at 100.

Furthermore, to handle the diverse perspectives of tomato disease images, we utilize convolutional kernels of various sizes (1×1, 3×3, 5×5) to extract features from the input visual features IVF (C, H, W). These convolutional kernels capture different spatial information at different scales. Afterwards, the features obtained at different scales are concatenated and merged to obtain the visual feature VF (CH, H, W). Additionally, we treat the obtained visual feature VF as the query Q, the input knowledge feature IKF as the key K and value V, and employ scaled dot-product attention to calculate the output feature map G (H, W, C). The calculation formula for this attention mechanism can be expressed as follows:

$$Attention(Q, K, V) = Softmax\left(\frac{Q, K^T}{\sqrt{d}}\right)V \tag{1}$$

In the above-mentioned formular, d denotes the dimension of the vectors Q and K.

Then, the fundamental concept behind the prior knowledge embedding strategy is to integrate visual information with prior knowledge through the attention mechanism. By combining the encoded knowledge features with the visual features corresponding to the input image, we achieve enhanced visual features that incorporate prior knowledge about tomato diseases. The output feature map G is multiplied by the visual features VF, resulting in the final enhanced visual features E-V (C, H, W) that are embedded with prior knowledge about tomato diseases.

In the end, the advantageous enhanced visual features EV resulting from the fusion process will be learned by the PKAM module in the subsequent encoding stage, thereby producing an output vector embedded with prior knowledge of tomato diseases. Here, C and C_1 represent the channel count in the feature maps, while H and W denote the height and width of the feature maps, respectively.

3.2 SPPCSPF module with structural reparameterization

This research has developed the SPPCSPF module to reduce memory access costs and improve model training efficiency. The module employs structurally reparameterized RepConv convolutional layers in place of regular convolutional layers within the residual

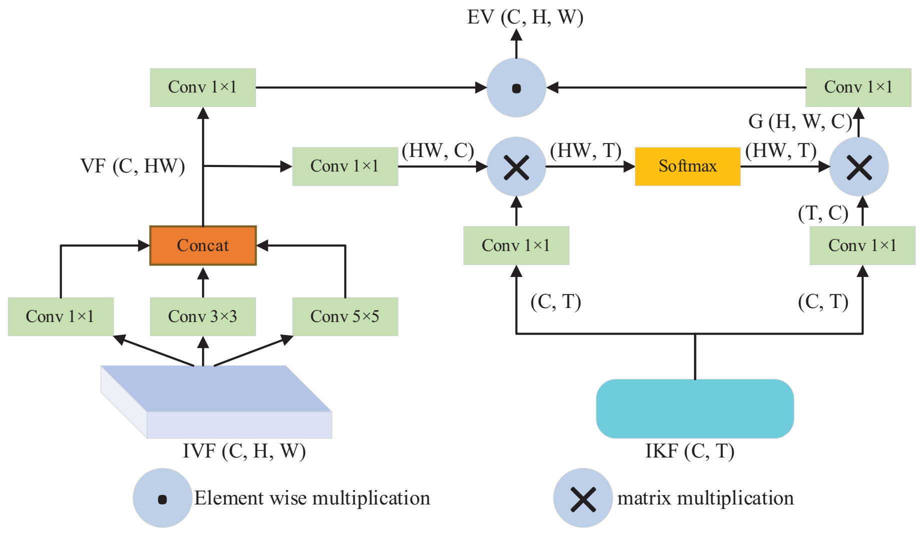


FIGURE 2
Framework of PKAM module.

structure. In training, a multi-branch residual structure is used for feature extraction. Additionally, during inference, the convolutional layers are merged with BN (Batch Normalization) layers, and the three branches are consolidated into a single-path model. As a result, all the trained parameters are equivalent to a 3×3 convolutional layer, enabling faster inference speed and reduced memory access costs.

Figure 3 illustrates the designed max pooling part of the SPPCSPF module, taking into consideration the increased training costs associated with using structurally reparameterized layers. The pooling kernel size is set to a fixed value of 5×5 in this design. Furthermore, the output from each pooling layer serves as input for the subsequent pooling layer. By setting a pooling stride of 1, two 5×5 max pooling layers can effectively perform the same function as a single 9×9 max pooling layer. Similarly, three 5×5 max pooling layers yield the same result as a single 13×13 max pooling layer. This approach significantly reduces the computational load during model training.

3.3 Multiscale detection module with small object detection layer

In this study, the target detection dataset images are of size 800×800. Since there are small objects with pixel sizes smaller than 8×8 in tomato disease detection images that have complex backgrounds, the study employs the principle of detecting large objects based on the small object detection feature map and vice versa. To maintain the same scale of the output feature map as the baseline model, a 160×160 detection layer is added in the prediction stage. This layer divides the input image into 160×160 grid cells, each measuring 5×5 in size. This approach enhances the regression and adjustment of prior boxes, resulting in accurate detection boxes for small objects and significantly enhancing precision in small object detection. To tackle the challenge of losing a substantial amount of feature information for smaller objects in deeper network layers, this study proposes the incorporation of innovative feature layers with alternative dimensions sourced from the backbone network. Furthermore, by enhancing the Neck component and constructing a parallel multi-branch feature fusion network, the loss of effective information in the feature maps is mitigated.

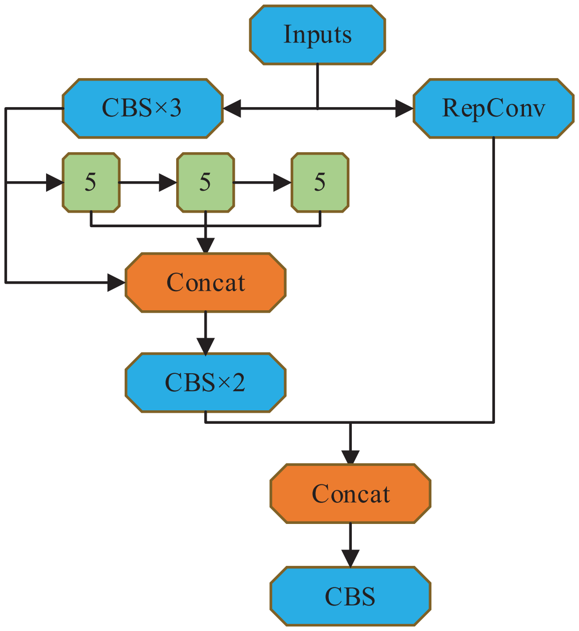


FIGURE 3
Network structure of SPPCSPF module.

3.4 A-SIOU loss function

Precise target localization is essential for successful target detection, relying heavily on the utilization of a superior quality bounding box loss function. The conventional CIOU loss function proficiently handles the task of orienting bounding boxes, even in situations where there is no intersection between the predicted and ground truth boxes. Accomplishing this involves incorporating the aspect ratio of the boxes into the loss calculation. However, the CIOU loss function has certain limitations, as it calculates all loss variables as a whole without adequately addressing the disparity between the actual target and the predicted box. As a result, this approach leads to slow convergence and instability issues.

In this study, we introduce the A-SIOU (Alpha SIOU) loss function as a replacement for the existing CIOU loss function in the tomato disease detection model. This novel bounding box loss function, based on enhancements to the SIOU loss function (Gevorgyan, 2205), offers significant improvements. It enhances the gradient convergence speed of the loss function through the parameter α (He et al., 2021). The A-SIOU loss function fully considers the influence of distance, angle, and area - these three key factors - on the boundary regression of the model. This ensures that the predicted box can converge towards the ground truth box more quickly, thereby controlling the convergence direction. The proposed loss function consists of four components: L_{angle}, L_{dis}, L_{shape}, and IOU.

The equation for computing the angle loss L_{angle} is given by:

$$\begin{cases} z_h = \left| b^t_{cy} - b^p_{cy} \right| \\ \sigma = \sqrt{(b^t_{cx} - b^p_{cx})^2 (b^t_{cy} - b^p_{cy})^2} \\ L_{angle} = 1 - 2\sin^2\left(arcsin\left(\frac{z_h}{\sigma}\right) - \frac{\pi}{4}\right) \end{cases} \tag{2}$$

In the given equation, z_h denotes the disparity in height between the center points of the predicted box and the ground truth box. σ represents the spatial displacement between the center coordinates of the ground truth box and the predicted box. Furthermore, b^t_{cx} and b^t_{cy} denote the center coordinates of the ground truth box, while b^p_{cx} and b^p_{cy} denote the center coordinates of the predicted box.

The formula for calculating the distance loss is provided below:

$$\begin{cases} \rho_x = \left(\frac{b^t_{cx} - b^p_{cx}}{c_w}\right)^2 \\ \rho_y = \left(\frac{b^t_{cy} - b^p_{cy}}{c_h}\right)^2 \\ L_{dis} = 2 - \sum_{i=x,y} exp\left((L_{angle} - 2)\rho_i\right) \end{cases} \tag{3}$$

Here, c_w and c_h represent the width and height of the minimum bounding rectangle of the ground truth box and the predicted box.

To compute the shape loss, use the following formula:

$$\begin{cases} \omega_\omega = \frac{|\omega^p - \omega^t|}{max(\omega^p, \omega^t)} \\ \omega_h = \frac{|h^p - h^t|}{max(h^p, h^t)} \\ L_{shape} = \sum_{i=w,h} (1 - exp(\omega_i))^\theta \end{cases} \tag{4}$$

In the equation above, ω^p and h^p represent the width and height of the predicted box, while ω^t and h^t represent the width and height of the ground truth box. θ is a constant used to control the emphasis on shape loss, with a value of 4 in this study.

The formula for calculating the A-SIOU loss function is as follows:

$$\begin{cases} IOU = \frac{A}{B} \\ L_{A-SIOU} = 1 - \left(IOU^\alpha - \frac{(L_{dis}^\alpha + L_{shape}^\alpha)}{2} \right) \end{cases} \tag{5}$$

In the equation above, A and B represent the intersection and union of the areas of the ground truth box and the predicted box. α is a variable that controls the convergence speed of the loss function. Through multiple experiments, it has been found that setting α to 2 helps the model focus more on targets with high intersection-over-union ratios, thereby improving the accuracy of object localization.

Compared to other functions, the A-SIOU boundary box loss function considers the influence of distance and angle on boundary regression, thereby avoiding the issue of gradient vanishing in cases where there is no overlap between the predicted box and the ground truth box. Additionally, the A-SIOU loss function include four components. In the angle loss component, the range of values for the angle loss L_{angle} is [0, 1] due to the characteristics of the sine function. In the distance loss component, considering the range of ρ_i to be [0, 1), the value range of the distance loss L_{dis} can be derived as (0, 2-2e-2). In the shape loss component, considering the range of ω_i to be (0, 1), the value range of the shape loss L_{shape} can be derived as (0,2(1-e)^4). In conclusion, the A-SIOU function has a value range of (-2 + 2e^{-4},1 + 2(1-e)^8), which has both upper and lower limits, effectively preventing gradient explosion.

3.5 Overall framework of PKAMMF

Based on the above improvement measures, the overall network framework of the PKAMMF method for tomato disease detection, which incorporates the fusion of prior knowledge attention mechanism and multi-scale features to enhance performance, is illustrated in Figure 4.

4 Dataset

4.1 Experimental data collection

The experimental research area was selected as the tomato planting base in Shouguang City, Shandong Province. This location is known for year-round cultivation of various tomato varieties, as shown in Figure 5. To collect data, agricultural IoT monitoring equipment equipped with a 4K high-definition camera (with a resolution of 4096x3112) was used. The equipment enabled the collection of typical tomato disease images from different plants, regions, and growth stages under natural conditions. Image collection took place during specific time intervals: 08:00 to 09:00, 11:00 to 12:00, and 15:00 to 16:00 every day, to capture images under varying lighting conditions. In total, 26,983 images depicting various types of tomato diseases were collected.

4.2 Dataset construction

To ensure data quality, the original image was cropped to a fixed size of 640 × 640. Cropping the image to this specific dimension ensures that the subject of the photograph is clear, the disease objects are easily discernible, and the real background is visible. In addition, we manually removed duplicate and low-quality images from the dataset. After the selection process, we obtained a total of 10,000 tomato disease images that represent various types of diseases. Next, we divided these images into a 9:1 ratio, creating a training set and a test set. The training set encompassed 9,000 images, while the test set contained 1,000 images. By including a diverse range of scene information, such as rainy and foggy weather, sunny days, cloudy conditions, and other scenarios, the dataset effectively captures real-world planting environment information.

In order to enhance the model's robustness to variations in tomato disease image sizes and lighting conditions, we employed a method to augment the training set. This involved changing the contrast and scaling the image sizes. The contrast coefficient and scaling factor were randomly generated within the ranges of [0.6, 1.5] and [0.6, 1.7], respectively. In order to enhance the model's capability in detecting occluded disease objects, we augmented the dataset by adding salt-and-pepper noise to simulate random pixels and artificially create occlusions. As a result, the augmented training set consisted of a total of 45,000 tomato disease images. Meanwhile, the test set remained in its original state, and data augmentation was solely applied to the training set. This decision was made to improve the dependability and precision of the test results.

4.3 Data annotations

The labeling of tomato disease samples is divided into two steps. Firstly, the prior knowledge information of tomato disease is labeled. Secondly, the tomato disease category information is labeled.

4.3.1 Labeling prior knowledge information of tomato diseases

The prior knowledge information of tomato diseases includes the identification of the disease infection location (leaves, stems, fruits) and the shooting angle from which the images are captured (main view, top view). To illustrate this, Table 1 presents a compilation of tomato disease images along with corresponding label examples.

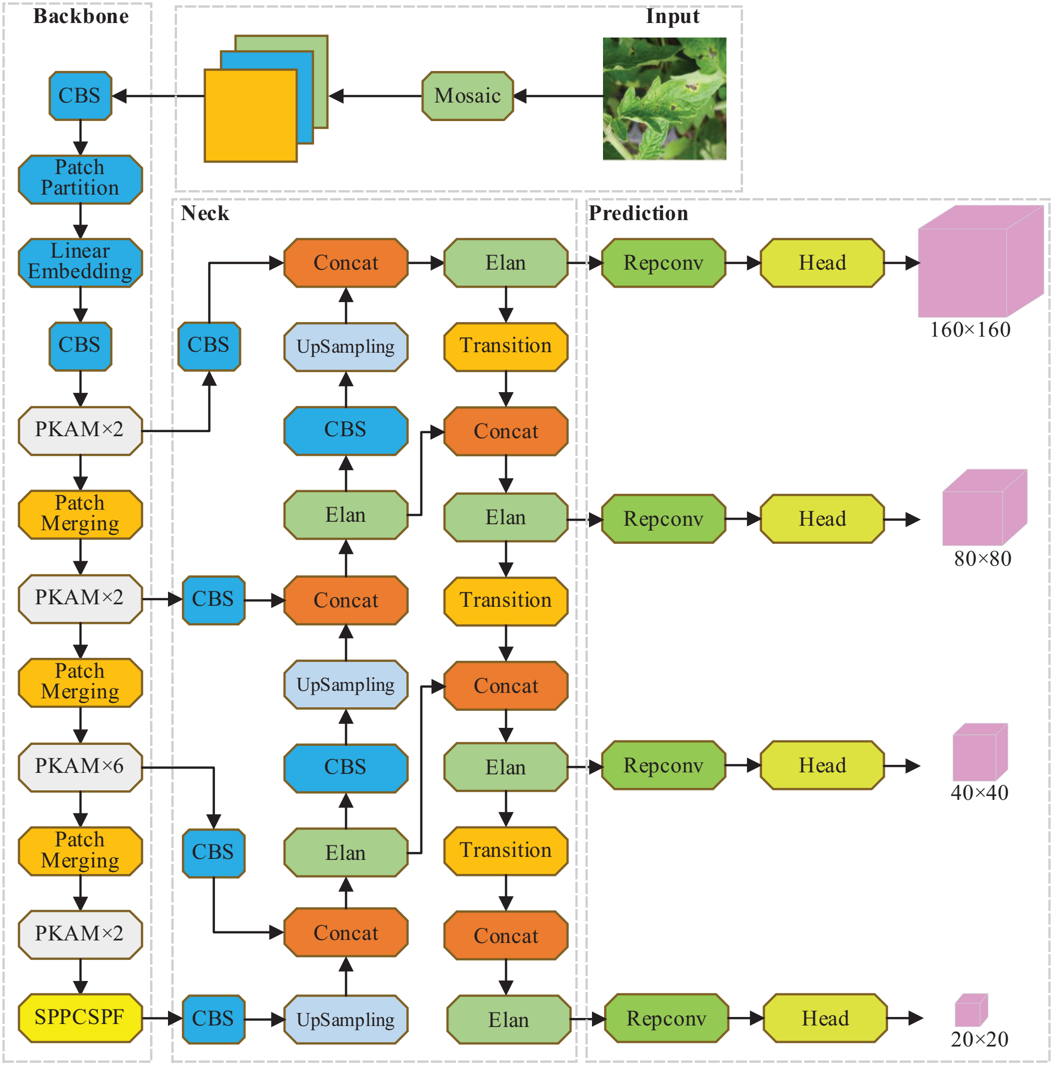


FIGURE 4
Overall framework of PKAMMF.

The labeled examples serve as valuable references for training and developing models or systems focused on tomato disease detection, making use of prior knowledge information.

4.3.2 Labeling tomato disease category information

The model used in this study was trained on a Pascal VOC-formatted dataset. To annotate the tomato disease images of different disease types, we utilized the LabelImg software. The annotation rules were as follows: 1) We annotated the diseased areas in the images without occlusion or with some occlusion that did not impact manual judgment of the disease type. 2) We did not annotate severely occluded areas where it was difficult for humans to determine the disease type accurately. Since tomato diseases rapidly spread, it is common for most images to contain multiple affected areas. In total, we annotated 127,356 diseased areas across 10 disease categories during the annotation process. The quantities of the different tomato disease types can be found in Table 2.

5 Results

5.1 Operating environment configuration

The experimental platform employs Ubuntu 22.04 as its operating system. It utilizes two NVIDIA RTX 3080 GPUs for deep learning, with a memory capacity of 12GB. Other software packages include Python 3.8, CUDA 11.0, Torch 1.7.0, and torchvision 0.8.1.

5.2 Model training

During the training phase, we performed pre-training using the weight file of the baseline model. Since the improved model shares most of its structure with the baseline model, many weights can be transferred from the baseline model to the improved network. This transfer of weights allows us to save a significant amount of training

FIGURE 5
Experimental site.

time. To carry out the training, a batch size of 16 was chosen, and the training process consisted of 300 epochs. We utilized the Adam algorithm for gradient descent. Additionally, the image size was adjusted to 416×416. The initial learning rate was set to 0.01, and the weight decay coefficient was determined to 0.000005. We employed the cosine annealing algorithm for learning rate adjustment.

Throughout the process of training the model, we recorded the loss function of the model and depicted it in Figure 6. According to the depicted graph, in the early stages of training, the loss function experiences rapid decrease with minor overall fluctuations. Notably, around the 20,000th iteration, the loss value reaches 0.022. After training for 30,000 iterations, the loss value converges and stabilizes at 0.016.

5.3 Metrics for evaluating performance

Before introducing the metrics, it is necessary to briefly explain the symbols used. In this research experiment, when IOU > 0.5, it is considered that the predicted box hits the annotated box; otherwise, it is considered that the predicted box does not hit the annotated box. TP indicates the count of correctly predicted boxes that match the annotated boxes for the given class, FP corresponds to the count of incorrectly predicted boxes that match the annotated boxes for the given class, TN is the count of predicted boxes that correctly match the annotated boxes for other classes, and FN represents the count of predicted boxes that fail to match any annotated boxes.

The commonly used metrics include Recall, Precision, Average Precision (AP), and mAP. Recall is used to evaluate whether the model predicts all target objects comprehensively. The model's prediction accuracy is assessed through Precision. To evaluate the model's classification performance for a particular class, AP calculates the area under the Precision-Recall curve, while mAP computes the average AP across multiple classes. Furthermore, the model's detection speed is measured in frames per second (FPS), representing the number of images detected per second. The formulas for calculating the above evaluation metrics are as follows:

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$AP = \int_{0}^{1} p(r)dr \tag{8}$$

$$mAP = \frac{\sum_{n=1}^{N} AP(n)}{N} \tag{9}$$

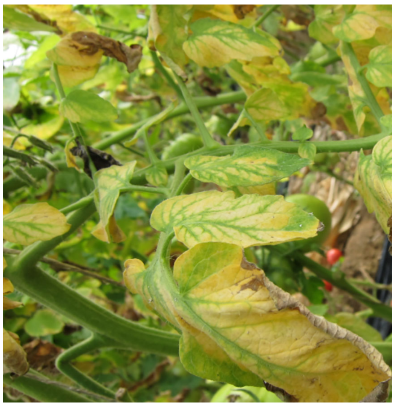
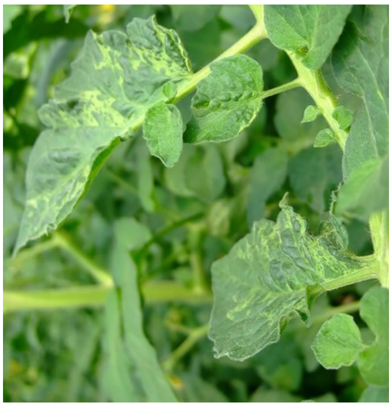
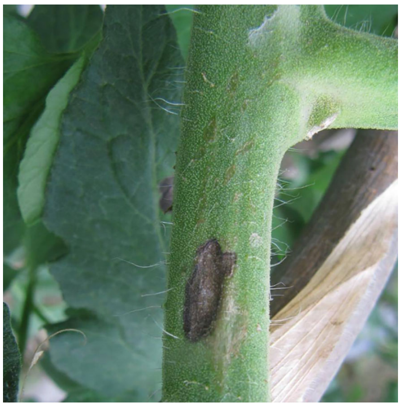
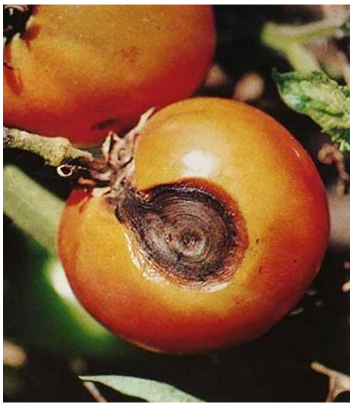
$$FPS = \frac{C_{img}}{Time_{detect}} \tag{10}$$

In the above-mentioned formulars, $p(\mathrm{r})$ represents the Precision-Recall curve. N denotes the overall count of categories within the tomato disease detection data, while n represents the current data category. C_{img} represents the count of pictures within the test dataset, and $Time_{detect}$ represents the time taken to detect C_{img} images.

5.4 Ablation experiment

In order to verify the performance enhancement of the various improvement measures proposed in this study for tomato disease image object detection, a series of experiments involving ablation was conducted. These experiments aimed to systematically assess the impact of each improvement measure by selectively removing or

TABLE 1 Tomato disease images and label examples.

Image No.	1	2	3	4
Tomato disease image				
Location	Leaves	Leaves	Stems	Fruits
Shooting Angle	Main View	Top View	Top View	Top View
Label	Pk-lm	Pk-lv	Pk-sv	Pk-fm

disabling them one by one. The experimental results are presented in Table 3, where PKAM, SPPCSPF, MSD, and A-SIOU represent the four improvement measures, namely, the PKAM module, the SPPCSPF module, the Multi-Scale Detection module, and the loss function, respectively.

The above-mentioned table reveals the following:

(1) Improvement 1 designs the PKAM module, which utilizes attention mechanisms to integrate visual information with prior knowledge. This enhances the network's ability to capture global observations. While the introduction of the PKAM module leads to a slight increase in the parameter count and a decrease in detection speed, it results in a significant improvement in mAP, with an increase of 2.23%. This indicates that the incorporation of the PKAM module helps to improve the overall performance of the detection by effectively leveraging both visual information and prior knowledge. Despite the trade-off in terms of computation and speed, the gained improvement in accuracy justifies the utilization of the PKAM module in the context of disease object detection in tomato plants.

(2) Building upon improvement 1, improvement 2 designs the SPPCSPF structure with structurally reparameterized RepConv convolutional layers. This design not only stabilizes the training process but also improves inference speed, resulting in an mAP improvement of 0.59% compared to improvement 1.

(3) Based on improvement 1, improvement 3 implements multi-scale detection by adding a new feature fusion layer in the Neck section and incorporating a small object prediction layer during inference. Compared to improvement 1, improvement 3 achieves a mAP improvement of 0.74%, indicating enhanced accuracy in detecting small objects.

(4) Improvement 4 integrates the first three improvements and achieves the highest mAP. Compared to the baseline model, improvement 4 shows an mAP enhancement of 3.86%.

(5) Continuing from improvement 4, this study further improves the proposed method by utilizing A-SIoU. Although the mAP improvement is only 0.59% compared to improvement 4, the convergence speed during actual training is faster.

Overall, through multiple improvement measures, the proposed method in this study achieves a 3.86% mAP increase compared to the baseline model. Despite a slight increase in parameter count, the number of parameters remains within the same order of magnitude as the baseline model, indicating that the additional computational requirements are manageable. The detection frame rate drops by 11.71 frames per second. However, it still meets the fundamental criteria for real-time performance. Considering these factors, the detection accuracy improvement obtained through the various improvement measures is highly cost-effective. The trade-offs in terms of parameter count and detection speed are reasonable, given the substantial enhancement in accuracy achieved by the proposed

TABLE 2 Sample quantities for each disease type.

No.	Disease class	Sample images in the training set	Annotated diseased areas in the training set	Sample images in the test set	Annotated diseased areas in the test set
1	Early blight	4500	10903	100	228
2	Late blight	4500	12317	100	247
3	Bacterial spot	4500	17302	100	469
4	Gray leaf spot	4500	13236	100	303
5	Gray mold	4500	12315	100	269
6	Leaf mold	4500	11871	100	235
7	Yellow leaf curl virus	4500	10036	100	213
8	Mosaic virus	4500	10277	100	208
9	Canker	4500	12398	100	249
10	Anthracnose	4500	13964	100	316
	Total	45000	124619	1000	2737

method. This implies that the proposed method may have practical value and can be considered as an effective solution for detecting objects related to diseases in tomato plants.

5.5 Comparative analysis of performance with alternative mainstream approaches

To further validate the advantages of the PKAMMF method in detecting tomato diseases in complex backgrounds, we conducted object detection experiments using our own tomato disease dataset under the same experimental environment and parameter settings. We compared and analyzed the object detection performance of mainstream methods such as Faster-RCNN, SSD, YOLO series, and PKAMMF in this study. The experimental outcomes from these various methodologies are presented in Table 4.

As shown in Table 4, the parameter size (Params) of single-stage object detection methods, including SSD, YOLO series, and the proposed method in this study, is relatively smaller than the two-stage object detection approach, Faster-RCNN. This reduction in parameter size leads to faster detection speed. In comparison to the baseline model, the proposed method in this study exhibits an increase of 14.87M parameters and a decrease of 11.71 frames per second (FPS) in the detection frame rate. However, it demonstrates an improvement of 3.26% in precision (P), 2.55% in recall (R), and achieves a mean average precision (mAP) of 91.96%. This performance enhancement surpasses YOLOv7 by 3.86% and outperforms other models in the table. It strongly indicates that

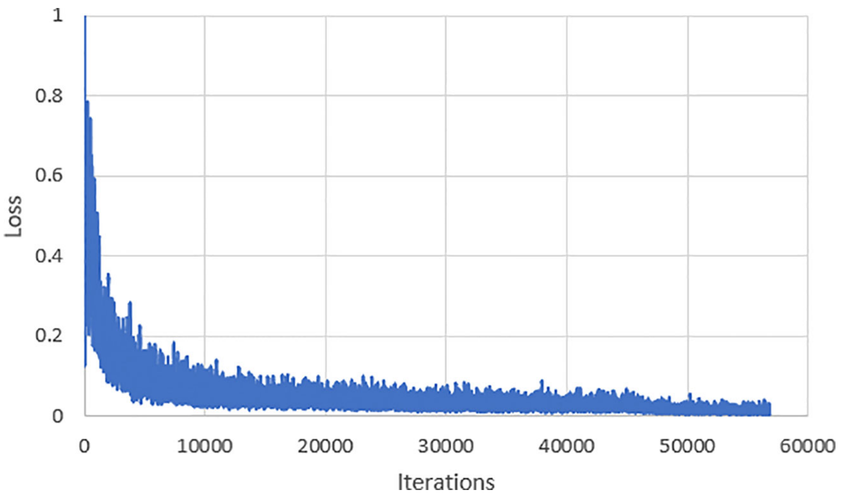


FIGURE 6
Variations of the loss function during the training process.

TABLE 3 Results of ablation experiments.

Methods	PKAM	SPPCSPF	MSD	A-SIOU	Params(M)	FPS(Frames per second)	mAP(%)
YOLOv7					36.853	66.39	88.10
Improvement 1	**Yes**				50.856	57.92	90.33
Improvement 2	**Yes**	**Yes**			51.267	57.98	90.92
Improvement 3	**Yes**		**Yes**		51.698	55.67	91.07
Improvement 4	**Yes**	**Yes**	**Yes**		51.279	55.32	91.37
PKAMMF	**Yes**	**Yes**	**Yes**	**Yes**	51.723	54.68	91.96

PKAMMF exhibits remarkable superiority in detecting tomato disease objects.

Given that the self-built tomato disease dataset comprises ten disease classes with significant variations in scale and features among different instances, achieving multi-scale object detection is typically challenging. Table 5 presents the average precision (AP) results of different methods for detecting various object categories.

According to Table 5, the experiment shows that the proposed PKAMMF outperforms the baseline model YOLOv7 among the 10 target categories. Therefore, in comparison to alternative prevalent object detection methods, PKAMMF also demonstrates significant advantages in detecting objects at multiple scales. The proposed method exhibits notable superiority in detecting tomato disease objects of varying scales, even within complex backgrounds. However, there is still room for improvement in detecting bacterial spot disease and gray leaf spot disease, as these targets have less distinct features. In scenarios involving multiple scales, there is a higher risk of false negatives occurring.

5.6 Performance comparison of different attention mechanisms in tomato disease detection

The comparative experiments were conducted under consistent conditions, and several classical attention mechanisms, namely SE (Hu et al., 2018) (Squeeze and Excitation), CBAM (Woo et al., 2018) (Convolutional Block Attention Module), GAM (Liu et al., 2021) (Global Attention Mechanism), and Biformer (Zhu et al., 2023), were added. Table 6 provides a performance comparison of the proposed prior knowledge attention mechanism and other attention mechanisms in tomato disease detection. The results clearly demonstrate that the proposed prior knowledge attention algorithm in this study achieves the highest mAP and exhibits a significant enhancement in detection performance.

This finding suggests that the prior knowledge attention mechanism effectively integrates prior knowledge of tomato diseases, enabling more focused feature extraction in the regions associated with tomato diseases. The prior knowledge attention mechanism helps to focus on relevant areas of the image that are more likely to contain disease objects. This can improve the efficiency and accuracy of detection by reducing false positives and identifying subtle disease symptoms. Consequently, the prior knowledge attention mechanism is deemed more suitable for feature extraction in tomato disease detection scenarios.

5.7 The influence of training samples of different sizes on detection results

The number of training samples can significantly influence the detection performance of a model, so training samples of different sizes should be evaluated. To investigate the impact of training sample size on the performance of the proposed PKAMMF model, it is necessary to employ a method that involves changing the number of training samples while keeping the test set samples unchanged. Building upon the experiments conducted in the previous sections, we randomly selected 5000, 10000, 15000,

TABLE 4 Performance metrics of various methods.

Methods	P(%)	R(%)	F_1 score(%)	Params(M)	FPS	mAP(%)
Faster-RCNN	85.98	61.79	71.85	129.8	10.59	70.73
SSD	90.87	55.87	68.92	25.43	47.65	72.54
YOLOv3	85.88	60.78	70.82	60.83	27.28	78.62
YOLOv4	87.95	68.54	76.21	62.73	34.07	80.37
YOLOv5	89.97	75.82	80.33	70.10	60.28	88.98
YOLOv7	88.64	83.52	85.97	36.853	66.39	88.10
PKAMMF	91.90	86.07	88.18	51.723	54.68	91.96

TABLE 5 Average precision of several methods for detecting different disease class.

Disease class	Faster-RCNN	SSD	YOLOv3	YOLOv4	YOLOv5	YOLOv7	PKAMMF
Early blight	89.3	90.2	90.6	90.1	95.4	96.7	98.3
Late blight	88.7	89.3	92.5	80.3	91.3	92.2	97.9
Bacterial spot	77.5	70.2	79.8	80.6	76.3	78.5	79.6
Gray leaf spot	77.3	80.1	72.7	75.8	76.2	76.4	79.1
Gray mold	69.4	66.8	92.1	88.9	90.0	90.2	93.4
Leaf mold	89.3	80.4	80.8	85.4	88.6	89.7	94.8
Yellow leaf curl virus	86.5	84.3	78.6	79.6	82.7	84.9	90.1
Mosaic virus	80.2	86.7	82.4	72.8	87.9	88.1	89.6
Canker	53.7	52.2	69.7	70.7	88.2	90.1	90.3
Anthracnose	42.6	53.9	80.4	76.9	89.3	90.5	91.1

20000, 25000, 30000, 35000, 40000, and 45000 samples, maintaining the same proportion of tomato disease categories from our self-built dataset. We employed the same training method to train datasets with varying sample sizes. The impact of training samples of different sizes on mean Average Precision (mAP) is illustrated in Figure 7.

From Figure 7, it can be seen that the learning ability of the model increases with the increase of sample size for different training sample Quantities. In the case of a small number of training samples (5000-200000), the mAP obtained in the experiment significantly can improve greatly. As the number of training samples continue to increase, the improvement of mAP slows down. When the number of training samples exceeds 30000, mAP gradually tends to stabilize.

5.8 Analysis of tomato disease object detection results

The proposed PKAMMF model was utilized to detect 10 types of tomato disease images under complex backgrounds in the test set, which comprised 1000 images. Figure 6 presents some of the disease detection results.

Based on Figure 8, it is evident that the proposed PKAMMF model exhibits accurate detection capabilities for tomato disease images. The results depicted in Figure 6 clearly showcase the robustness and adaptability of the PKAMMF model in handling challenging scenarios commonly encountered in tomato disease detection. The model effectively addresses the difficulties posed by objects at different scales, instances where objects are partially obscured, and varying lighting conditions, all of which are prevalent in real-world situations. Also, by effectively identifying objects in tomato disease images under challenging conditions, the model minimizes instances where diseases go undetected (missed detections) and also reduces the occurrence of incorrectly identifying healthy regions as diseased (false detections). By accurately detecting objects under such conditions, the PKAMMF model proves its capability to improve the performance of tomato disease detection. Its ability to handle complex backgrounds further strengthens its practical applicability in agricultural settings.

TABLE 6 Comparison of different attention mechanisms.

Algorithms with different attention mechanisms	mAP (%)
baseline	88.10
baseline+SE	88.26
baseline+ CBAM	89.39
baseline+ GAM	89.27
baseline+Biformer	89.56
baseline+PKAM	90.33

6 Discussion

In response to the limitations of existing deep learning models in learning prior knowledge of tomato disease objects and their reliance solely on visual features, this study proposes a method for tomato disease detection. The main objective is to leverage the prior knowledge available in tomato disease images and achieve accurate disease detection in complex backgrounds. To address this challenge, our proposed method, called tomato disease object detection method combining Prior Knowledge Attention Mechanism and Multiscale Features (PKAMMF), is introduced. By integrating the visual features extracted from detected images with the prior knowledge of tomato diseases, the overall performance of tomato disease detection in complex natural backgrounds is significantly enhanced. Through comprehensive experimental analysis and comparisons with existing methods, we have drawn the following discussions:

(1) In response to the challenge posed by complex backgrounds and unclear, overlapping target features in

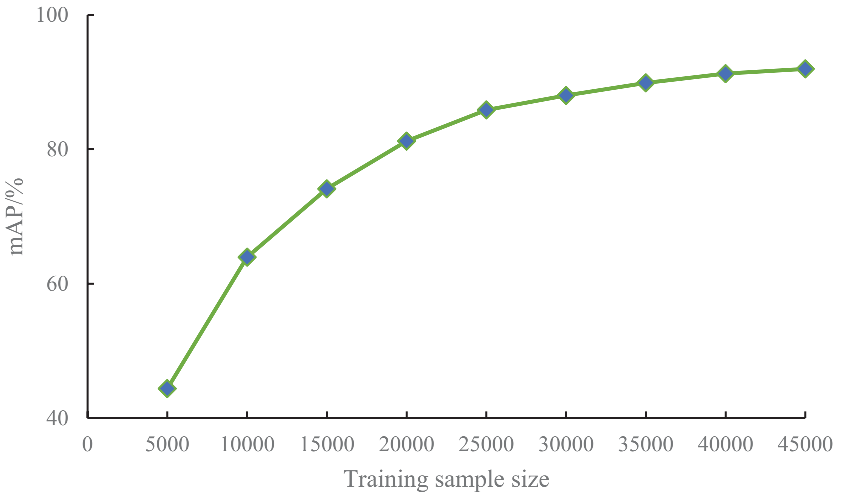


FIGURE 7
The impact of training sample size on detection results.

tomato disease images, this study investigates the utilization of prior knowledge on tomato diseases. We incorporate prior knowledge as auxiliary information into our model, enabling the detection network to effectively learn the distinctive features of various categories of tomato diseases and achieve accurate detection.

(2) Incorporate a feature fusion layer within the Neck section to facilitate effective information transmission across the backbone network. Additionally, augment the prediction section with a small object detection layer, enabling improved performance in detecting small objects at multiple scales. This enhancement reduces both the missed detection rate and false detection rate. Moreover, introduce the A-SIoU loss function to expedite bounding box regression, thereby accelerating the convergence speed.

(3) Validate the proposed algorithm using a self-built dataset specifically designed for tomato disease detection. The experimental results demonstrate that the proposed model adeptly utilizes the prior knowledge inherent in tomato disease images. It achieves accurate detection of small target diseases and effectively identifies densely occluded diseases against complex backgrounds. This approach significantly enhances the overall detection performance of tomato diseases and mitigates the occurrence of missed and false detections arising from complex backgrounds. Furthermore, the proposed model exhibits good real-time performance.

This study focuses on leveraging prior knowledge to enhance the detection effectiveness of tomato diseases. The experimental

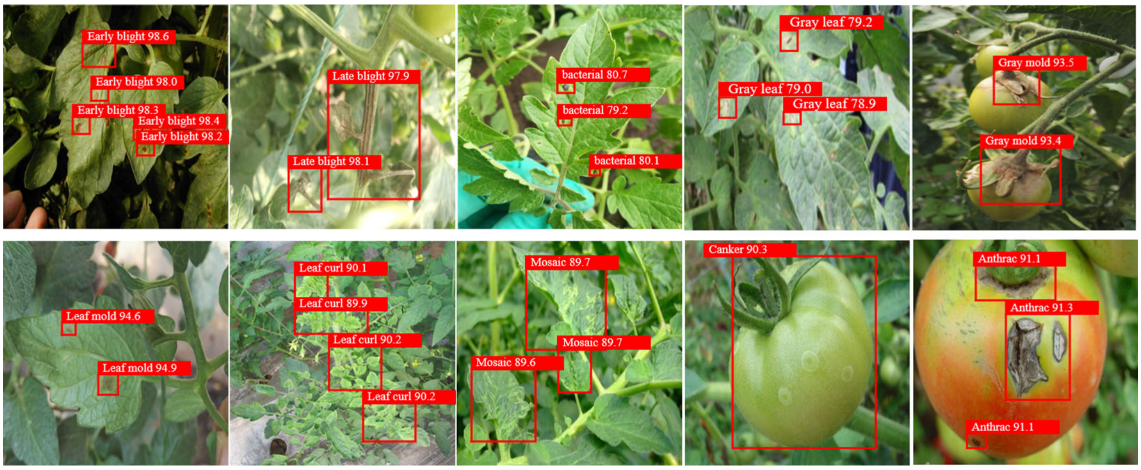


FIGURE 8
Object detection results of tomato disease.

results validate that, with the guidance of prior knowledge, the model performs significantly better in detecting tomato diseases amidst complex natural backgrounds. This research sets the groundwork for integrating prior knowledge of tomato diseases with deep learning models, offering new insights and ideas for intelligent disease detection technology in plants. However, it is important to note that the proposed model currently only incorporates explicit knowledge, such as the precise location and shooting angle of tomato diseases, at the coding level. It lacks the capability to autonomously acquire implicit knowledge, including expert experience and the utilization of existing "knowledge" for reasoning. Therefore, future work should explore ways to integrate implicit knowledge, such as expert experience, into the model by employing technologies like knowledge graphs. Additionally, there are plans to conduct in-depth research into the fusion of prior knowledge and the model, incorporating spatial location relationships among diseases or prior knowledge about disease occurrence time to achieve more accurate disease detection. Knowledge reasoning methods will be employed to express prior knowledge more effectively, and efforts will be made to further enhance the proposed method and apply it to a wider range of plant disease detection scenarios, aiming for more accurate multi-category plant disease detection.

Data availability statement

The dataset and code in this study can be accessed by contacting the corresponding author.

Author contributions

XW: Conceptualization, Investigation, Software, Writing – original draft, Writing – review & editing. JL: Funding acquisition, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing.

Funding

The funding for this study is provided by the Natural Science Foundation of Shandong Province (ZR2021QC173).

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

Abbas, A., Jain, S., Gour, M., and Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Comput. Electron. Agric.* 187, 106279. doi: 10.1016/j.compag.2021.106279

Chen, Z., Wu, R., Lin, Y., Li, C., Chen, S., Yuan, Z., et al. (2022). Plant disease recognition model based on improved YOLOv5. *Agronomy* 12 (2), 365. doi: 10.3390/agronomy12020365

Cheng, S., Cheng, H., Yang, R., Zhou, J., Li, Z., Shi, B., et al. (2023). A high performance wheat disease detection based on position information. *Plants* 12 (5), 1191. doi: 10.3390/plants12051191

David, H. E., Ramalakshmi, K., Gunasekaran, H., and Venkatesan, R. (2021). "Literature review of disease detection in tomato leaf using deep learning techniques," in *2021 7th International conference on advanced computing and communication systems (ICACCS)* IEEE, Vol. 1. 274–278.

Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* 18.

Diligenti, M., Roychowdhury, S., and Gori, M. (2017). "Integrating prior knowledge into deep learning," in *2017 16th IEEE international conference on machine learning and applications (ICMLA)*. IEEE 920–923.

Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17 (9), 2022. doi: 10.3390/s17092022

Gevorgyan, Z. (2205). SIoU loss: More powerful learning for bounding box regression. *arXiv* 12740, 2022.

Gonzalez-Huitron, V., León-Borges, J. A., Rodriguez-Mata, A. E., Amabilis-Sosa, L. E., Ramírez-Pereda, B., and Rodriguez, H. (2021). Disease detection in tomato leaves *via* CNN with lightweight architectures implemented in Raspberry Pi 4. *Comput. Electron. Agric.* 181, 105951. doi: 10.1016/j.compag.2020.105951

Guo, Y., Lan, Y., and Chen, X. (2022). CST: Convolutional Swin Transformer for detecting the degree and types of plant diseases. *Comput. Electron. Agric.* 202, 107407. doi: 10.1016/j.compag.2022.107407

He, J., Erfani, S., Ma, X., Bailey, J., Chi, Y., and Hua, X. S. (2021). $\alpha $-ioU: A family of power intersection over union losses for bounding box regression. *Adv. Neural Inf. Process. Syst.* 34, 20230–20242.

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7132–7141.

Huang, X., Chen, A., Zhou, G., Zhang, X., Wang, J., Peng, N., et al. (2023). Tomato leaf disease detection system based on FC-SNDPN. *Multimedia Tools Appl.* 82 (2), 2121–2144. doi: 10.1007/s11042-021-11790-3

Karthik, R., Hariharan, M., Anand, S., Mathikshara, P., Johnson, A., and Menaka, R. (2020). Attention embedded residual CNN for disease detection in tomato leaves. *Appl. Soft Computing* 86, 105933.

Liu, J., and Wang, X. (2020). Tomato diseases and pests detection based on improved Yolo V3 convolutional neural network. *Front. Plant Sci.* 11, 898. doi: 10.3389/fpls.2020.00898

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

Liu, Y., Shao, Z., and Hoffmann, N. (2021). Global attention mechanism: Retain information to enhance channel-spatial interactions. *arXiv*.

Misra, N. N., Dixit, Y., Al-Mallahi, A., Bhullar, M. S., Upadhyay, R., and Martynenko, A. (2020). IoT, big data, and artificial intelligence in agriculture and food industry. *IEEE Internet things J.* 9 (9), 6305–6324.

Mohammad-Razdari, A., Rousseau, D., Bakhshipour, A., Taylor, S., Poveda, J., and Kiani, H. (2022). Recent advances in E-monitoring of plant diseases. *Biosensors Bioelectronics* 201, 113953. doi: 10.1016/j.bios.2021.113953

Moussafir, M., Chaibi, H., Saadane, R., Chehri, A., Rharras, A. E., and Jeon, G. (2022). Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks. *Plant Soil* 479 (1-2), 251–266. doi: 10.1007/s11104-022-05513-2

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. Agric.* 194, 106780. doi: 10.1016/j.compag.2022.106780

Roy, A. M., Bose, R., and Bhaduri, J. (2022). A fast accurate fine-grain object detection model based on YOLOv4 deep neural network. *Neural Computing Appl.*, 1–27. doi: 10.1007/s00521-021-06651-x

Thai, H. T., Le, K. H., and Nguyen, N. L. T. (2023). FormerLeaf: An efficient vision transformer for Cassava Leaf Disease detection. *Comput. Electron. Agric.* 204, 107518. doi: 10.1016/j.compag.2022.107518

Thakur, P. S., Khanna, P., Sheorey, T., and Ojha, A. (2022). Trends in vision-based machine learning techniques for plant disease identification: A systematic review. *Expert Syst. Appl.*, 118117. doi: 10.1016/j.eswa.2022.118117

Thangaraj, R., Anandamurugan, S., Pandiyan, P., and Kaliappan, V. K. (2022). Artificial intelligence in tomato leaf disease detection: a comprehensive review and discussion. *J. Plant Dis. Prot.* 129 (3), 469–488. doi: 10.1007/s41348-021-00500-8

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst.* 30.

Wang, F., Wang, R., Xie, C., Yang, P., and Liu, L. (2020). Fusing multi-scale context-aware information representation for automatic in-field pest detection and recognition. *Comput. Electron. Agric.* 169, 105222. doi: 10.1016/j.compag.2020.105222

Wang, J., Yu, L., Yang, J., and Dong, H. (2021). DBA_SSD: A novel end-to-end object detection algorithm applied to plant disease detection. *Information* 12 (11), 474. doi: 10.3390/info12110474

Widjaja, G., Rudiansyah, M., Sultan, M. Q., Ansari, M. J., Izzat, S. E., Al Jaber, M. S., et al. (2022). Effect of tomato consumption on inflammatory markers in health and disease status: A systematic review and meta-analysis of clinical trials. *Clin. Nutr. ESPEN* 50, 93–100. doi: 10.1016/j.clnesp.2022.04.019

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)*. 3–19.

Xu, Y., Chen, Q., Kong, S., Xing, L., Wang, Q., Cong, X., et al. (2022). Real-time object detection method of melon leaf diseases under complex background in greenhouse. *J. Real-Time Image Process.* 19 (5), 985–995. doi: 10.1007/s11554-022-01239-7

Zhang, K., Wu, Q., and Chen, Y. (2021). Detecting soybean leaf disease from synthetic image using multi-feature fusion faster R-CNN. *Comput. Electron. Agric.* 183, 106064. doi: 10.1016/j.compag.2021.106064

Zhang, Y., Song, C., and Zhang, D. (2020). Deep learning-based object detection improvement for tomato disease. *IEEE Access* 8, 56607–56614. doi: 10.1109/ACCESS.2020.2982456

Zhao, Y., Liu, L., Xie, C., Wang, R., Wang, F., Bu, Y., et al. (2020). An effective automatic system deployed in agricultural Internet of Things using Multi-Context Fusion Network towards crop disease recognition in the wild. *Appl. Soft Computing* 89, 106128. doi: 10.1016/j.asoc.2020.106128

Zhao, Z. Q., Zheng, P., Xu, S. T., and Wu, X. (2019). Object detection with deep learning: A review. *IEEE Trans. Neural Networks Learn. Syst.* 30 (11), 3212–3232. doi: 10.1109/TNNLS.2018.2876865

Zhu, L., Wang, X., Ke, Z., Zhang, W., and Lau, R. W. (2023). "BiFormer: vision transformer with bi-level routing attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10323–10333.

Zou, Z., Chen, K., Shi, Z., Guo, Y., and Ye, J. (2023). Object detection in 20 years: A survey. *Proc. IEEE.* doi: 10.1109/JPROC.2023.3238524

TYPE Original Research
PUBLISHED 20 November 2023
DOI 10.3389/fpls.2023.1289464

OPEN ACCESS

EDITED BY
Chunlei Xia,
Chinese Academy of Sciences (CAS), China

REVIEWED BY
Lingxian Zhang,
China Agricultural University, China
Nitin Goyal,
Central University of Haryana, India

*CORRESPONDENCE
Jun Liu
liu_jun860116@wfust.edu.cn
Xuewei Wang
wangxuewei@wfust.edu.cn

RECEIVED 07 September 2023
ACCEPTED 30 October 2023
PUBLISHED 20 November 2023

CITATION
Liu J, Wang X, Zhu Q and Miao W (2023) Tomato brown rot disease detection using improved YOLOv5 with attention mechanism. *Front. Plant Sci.* 14:1289464. doi: 10.3389/fpls.2023.1289464

Tomato brown rot disease detection using improved YOLOv5 with attention mechanism

Jun Liu*, Xuewei Wang*, Qianyu Zhu and Wenqing Miao

Shandong Provincial University Laboratory for Protected Horticulture, Weifang University of Science and Technology, Weifang, China

Brown rot disease poses a severe threat to tomato plants, resulting in reduced yields. Therefore, the accurate and efficient detection of tomato brown rot disease through deep learning technology holds immense importance for enhancing productivity. However, intelligent disease detection in complex scenarios remains a formidable challenge. Current object detection methods often fall short in practical applications and struggle to capture features from small objects. To overcome these limitations, we present an enhanced algorithm in this study, building upon YOLOv5s with an integrated attention mechanism for tomato brown rot detection. We introduce a hybrid attention module into the feature prediction structure of YOLOv5s to improve the model's ability to discern tomato brown rot objects in complex contexts. Additionally, we employ the CIOU loss function for precise border regression. Our experiments are conducted using a custom tomato disease dataset, and the results demonstrate the superiority of our enhanced algorithm over other models. It achieves an impressive average accuracy rate of 94.6% while maintaining a rapid detection speed of 112 frames per second. This innovation marks a significant step toward robust and efficient disease detection in tomato plants.

KEYWORDS

object detection, tomato brown rot, YOLOv5, hybrid attention, loss function

1 Introduction

Plant protection is a crucial aspect of agriculture, and precise disease detection and early forecasting are pivotal for maximizing crop yields. Presently, the identification and prediction of crop diseases heavily depend on local plant protection agencies. Nevertheless, constrained resources and a scarcity of experts present obstacles to the widespread implementation of scientific prevention and control strategies across diverse regions. Additionally, rural agricultural workers frequently lack the necessary expertise, leading to suboptimal disease management and impeding large-scale control initiatives. Since crops are predominantly grown in dispersed locations by individual farmers, the outbreak of diseases poses a significant challenge. Indiscriminate use of chemical pesticides by farmers not only exacerbates regional drug resistance but also poses a severe threat to the ecological environment (Horvath, 2018). Hence, it is crucial to propose a timely and efficient detection method that accurately identifies and diagnoses crop diseases. This will enable the implementation of appropriate preventive measures aimed at minimizing losses.

Tomatoes are among the most widely cultivated vegetables globally. However, they are susceptible to various diseases, with brown rot being one of the most prevalent. Therefore, this study takes tomato brown rot as a representative example. Also known as tomato fruit drop, tomato brown rot is a prominent ailment affecting tomatoes (Review, 1991). This disease can occur in both open field and greenhouse production systems, and tomatoes may become infected at any stage of growth, leading to devastating consequences and significant losses for vegetable farmers. In recent years, the incidence of tomato brown rot has increased, resulting in substantial economic losses for farmers. Extensive field investigations and comprehensive analyses have revealed that tomato brown rot primarily manifests after heavy rainfall and high temperatures, causing plant wilting and fruit drop. Generally, crop yield losses range from 30% to 40% in affected areas, while severely affected regions experience yield reductions exceeding 50% (Liaquat et al., 2019). Immature fruits are particularly vulnerable to tomato brown rot, although the stems and leaves can also be infected. The symptoms of this disease are multifaceted and often coexist with other disease symptoms, significantly complicating diagnosis efforts. Accordingly, there is an urgent need to develop a rapid and reliable detection method for early identification of tomato brown rot.

The current diagnosis of tomato brown rot predominantly depends on visual assessment by trained experts. Nevertheless, this method demands substantial time for professional training, and human judgment is inherently subjective, complicating the establishment of standardized criteria. In contrast, Artificial Intelligence (AI) presents a range of advantages, encompassing objectivity, enhanced accuracy, and measurable judgment outcomes. Integrating AI into the investigation of tomato brown rot and related diseases allows for the resolution of qualitative issues with heightened precision and the analysis of quantitative concerns with greater accuracy.

Most previous research on disease identification methods has been conducted in laboratory settings or controlled conditions. The limited number of samples obtained in natural environments has hindered the generalization capability of models. When utilizing large public datasets, the simplicity of image backgrounds and insufficient data representation become significant issues. Consequently, when applied to practical scenarios, the lack of dataset representativeness diminishes the model's ability to extract disease characteristics from complex backgrounds. This inadequacy results in reduced accuracy and speed in crop disease detection. To enhance model accuracy, researchers typically employ deep learning network structures with more convolutional layers to extract object features. However, this increases computational requirements and hardware dependence, leading to excessively long recognition times. As a result, the effectiveness of practical implementation in natural environments is severely restricted.

In this study, we employ a deep learning object detection algorithm to develop an online, non-destructive method for identifying tomato brown rot. This approach seeks to overcome the limitations of traditional artificial recognition, addressing its challenges and constraints. To enhance the precision of tomato brown rot disease identification and localization, we present an algorithm built upon an improved YOLOv5.

Our primary contributions are as follows:

1. We introduce a hybrid attention module into YOLOv5's feature prediction structure, bolstering its capacity to learn features from disease objects in complex contexts.
2. We replace the GIoU Loss with CIoU Loss, resulting in an accelerated bounding box regression rate and improved positional accuracy. This, in turn, enhances the detection of diseased objects.
3. We conduct experiments using a tomato disease dataset, with results demonstrating that our algorithm achieves a mean average precision (mAP) of 94.6%, a noteworthy 4.8% improvement over YOLOv5. Moreover, our detection accuracy significantly outperforms other mainstream algorithms.

In conclusion, our algorithm successfully fulfills the demands for accurate identification and localization of tomato brown rot in complex greenhouse environments.

2 Related work

2.1 Plant disease image recognition based on simple background

Plant disease images with a simple background exhibit characteristics such as a single background, minimal interference factors, and distinctive disease features. Previous research in this area has focused on improving disease feature extraction and reducing recognition error rates. Chen et al. (2020) utilized VGGNet and Inception modules pre-trained on ImageNet for rice disease recognition, achieving an average precision rate at 92%. Mensah et al. (2020) introduced a Gabor Capsule network for tomato and citrus disease recognition in the PlantVillage dataset, attaining a test set accuracy of 98.13%. Atila et al. (2021) introduced EfficientNet, achieving an accuracy of 99.91%. Jain et al (Jain and Gour, 2021). proposed a method using conditional generation inverse network (C-GAN) to generate composite images, an accuracy rate of 99.51% was attained. Joshi et al (Joshi and Bhavsar, 2020). investigated feature extraction of crop images affected by bacillus through a multi-layer convolutional neural network and fusion of multi-feature images, yielding promising results in crop bacterial disease recognition. Agarwal et al. (2020) introduced a technology to categorie 10 types of tomato disease images, designing an 8-layer convolutional neural network structure. However, due to limited availability of samples for the 10 tomato disease categories, the author indicated room for improvement in classification accuracy. Zhang et al. (2020)

utilized an enhanced Faster RCNN to detect healthy tomato leaves and four distinct disease types. Instead of VGG16, they employed a depth residual network for image feature extraction and implemented the k-means clustering algorithm for bounding box clustering. Experimental results on open datasets showed an average recognition accuracy of 98.54% with a detection time of only 470 ms.

These studies have yielded favorable outcomes in identifying plant diseases against simple backgrounds. However, Notably, the experimental data acquired controlled laboratory settings significantly differs from the complex background scenarios encountered in actual agricultural production processes. Consequently, the aforementioned research might experience a notable decline in disease identification accuracy when confronted with images collected under realistic complex backgrounds.

2.2 Plant disease image recognition in natural scene

Plant disease images captured in natural scenes are characterized by complex backgrounds, which accurately represent real-world application scenarios. The primary focus of research regarding plant disease images in natural scenes lies in eliminating the impact of complex backgrounds and non-standard photography on disease recognition accuracy. Lee et al. (2020) collected and trained 1822 tea pathological images, conducting experiments using Faster RCNN, which achieved an accuracy rate of 89.4%. Bollis et al. (2020) implemented a approach that substantially reduced the annotation task and was applied to identify citrus crop diseases and pests using the self-established CPB dataset, achieving an accuracy of 91.8%. Demird et al (Demir and Vedat, 2021). combined a newly developed depth CNN model with the impulse neural network (SNN) model. Experimental results demonstrated that the hybrid model proposed achieved an accuracy rate of 97.78%. Wang et al. (2021) introduced a new method that achieved disease identification accuracy of 99.7% in controlled laboratory environments. However, when tested in realistic environments, the disease identification accuracy dropped to 75.58%. Yang et al. (2021) introduced an innovative rebalancing convolutional network designed specifically for rice diseases and pests based on field image data, achieving an accuracy of 97.58%. He et al. (2021) developed a watermelon disease detection algorithm. They improved the preselector setting formula of the SSD model, resulting in an average accuracy of 92.4%. Wu et al. (2021) utilized two state-of-the-art object detection algorithms, and experimental results showed a precision range of 0.602-0.64, wherein YOLOv3 demonstrated a smaller size and faster processing speed compared to Faster RCNN. Temniranrat et al. (2021) proposed a rice disease diagnosis system based on improvements to the YOLOv3 model. The detection accuracy of this model reached 78.86%, with the entire detection process taking approximately 2-3 seconds. Gautam et al. (2022) considered various architectures, namely InceptionV3, VGG16, ResNet, SqueezeNet, and VGG19, for the detection of diseases in rice leaves. They employed additional fully connected layers in deep neural networks (DNN) to identify biotic diseases in rice leaves caused by fungi and bacteria, achieving an accuracy of 96.4%. Aggarwal et al. (2023) introduced a lightweight federated learning architecture for rice leaf disease recognition, achieving outstanding training and evaluation accuracy of 99%. Their research revealed that a federated learning model with multiple clients outperformed traditional pre-trained models.

These studies demonstrate that deep learning exhibits outstanding performance within the domain of plant diseases detection under natural scenes and can serve as a powerful technical tool. However, there is currently a lack of research on tomato brown rot based on deep learning. Given the complexity of its symptoms, manual identification of diease symptoms remains the primary method for early diagnosis of this disease. Therefore, exploring the application of deep learning in recognizing tomato brown rot diease symptoms holds significant research potential.

2.3 Object detection algorithms

Computer vision encompasses a critical research field known as object detection, which serves as the fundamental linking component between object recognition and tracking. The main objective of algorithms for object detection is to accurately recognize and locate specific targets present in images by determining their location and classification. Two main categories exist for these algorithms: the first is candidate region-based (two-stage), and the second is regression-based (one-stage). The key distinction between these two categories lies in the approach utilized for generating candidate bounding boxes. The former utilizes sub-network assistance to generate candidate bounding boxes, while the latter directly produces them on the feature map.

The algorithm utilizing candidate regions is an adaptation of the RCNN proposed by Girshick et al. in 2014 (Girshick et al., 2014). RCNN was the pioneer in incorporating deep learning into the field of object detection, marking a significant breakthrough, achieving an mAP value of 66.0% on the Pascal VOC dataset. Building upon this foundation, subsequent algorithms such as Faster RCNN (Ren et al., 2017), and Mask RCNN (He et al., 2017) have emerged. On the other hand, the regression-based algorithm traces its origins to the YOLO (Redmon et al., 2016) introduced by Redmon et al. in 2016 and the SSD algorithm (Liu et al., 2016) proposed by Liu et al. This approach transforms detection into a regression problem and significantly enhances detection speed. Further advancements have led to the development of algorithms like RSSD (Jeong et al., 2017), YOLO v2 (Redmon and Farhadi, 2017), YOLO v3 (Redmon and Farhadi, 2018), YOLO v4 (Bochkovskiy et al., 2020), YOLO v5 (YOLOv5, 2021), YOLOX (Ge et al., 2021), YOLOV6 (Li et al., 2022), YOLOV7 (Wang et al., 2023) and YOLOV8 (Terven and Cordova-Esparza, 2023).

Algorithms based on candidate regions generally exhibit slower detection speeds and do not meet real-time detection requirements, but they achieve good detection accuracy. On the other hand, regression-based algorithms offer faster detection speeds and better real-time performance, although their detection accuracy is poorer compared to two-stage algorithms. Currently, extensive research has led to the proposal of various object detection algorithms. Moving forward, algorithm development should prioritize lightweight object detection algorithms that strike a balance between parallel detection speed and accuracy.

2.4 Attention mechanism

Since the data used in this study consists of tomato disease data obtained under greenhouse conditions, it is susceptible to environmental factors and may contain significant amounts of noise during the identification process. This noise information can also propagate through the network model. With the increase in the number of network layers during the learning process, there is a corresponding amplification of noise information within the feature map, ultimately impacting the model negatively (Zhu et al., 2021). To address this issue, the model incorporates an attention mechanism, which serves as a solution. It is a critical concept in neural networks that was initially applied in machine translation and has now gained wide usage in computer vision. The attention mechanism can be intuitively explained through the human visual mechanism. Its fundamental idea is to filter out irrelevant information and prioritize key information similar to human vision. By adjusting the weights of each channel, the attention mechanism assists the model in capturing more useful semantic information for the recognition task. As a result, it enhances valuable information, suppresses the weight of noise and other interfering elements, and mitigates their negative impact on recognition. Moreover, The incorporation of the attention mechanism enhances the overall performance and effectiveness by directing more attention to effective features, ultimately enhancing the model's recognition performance (Ju et al., 2021). Yohanandan et al. (2018) (Yohanandan et al., 2018) pointed out that the visual attention mechanism closely aligns with human visual cognition. Consequently, leveraging the visual attention mechanism in computer vision offers significant benefits for various tasks. In recent years, numerous researchers have effectively improved key feature extraction capabilities in object detection networks by incorporating visual attention mechanisms. It has been demonstrated that attention mechanisms are an excellent choice for enhancing model performance.

The object detection technology based on the YOLO algorithm has demonstrated significant advancements in image recognition in recent years. However, certain challenges still exist when it comes to object detection in plant disease images. Considering the issues prevalent in detecting tomato brown rot images within complex scenes, this study proposes the utilization of the YOLOv5 network with a Hybrid attention module for tomato brown rot detection. By integrating a Hybrid attention module into the feature prediction structure of YOLOv5, the capability to learn features of diseased objects amidst complex backgrounds is enhanced. Additionally, the loss function is improved considering the characteristics of the disease spots, thereby improving the detection performance of diseased objects within the image. Finally, a series of comparative tests are carried out to assess the efficacy of the algorithm.

3 Materials

3.1 Dataset collection

For this study, tomato disease images were captured at the greenhouse tomato experimental base of our laboratory to create a dataset of tomato disease images. The dataset comprises two categories of images: healthy tomato leaves and tomato brown rot images. To account for various weather conditions in natural settings, images were acquired under different scenarios, including sunny and cloudy days, morning and afternoon sessions, and both normal photography and backlight photography. The image acquisition process encompassed the early, middle, and late stages of the disease.

The collection device used in this study is a remote-operated patrol robot equipped with a high-definition camera (HS-CQAI-1080, 4 megapixels). This camera enables the capture of 360-view greenhouse plant images and offers various functionalities such as zoom, dimming, zoom-in and zoom-out capabilities, as well as preset position settings. The robot has a maximum horizontal moving distance of 27 meters and a vertical moving distance of 1.5 meters. It allows for the collection of high-definition images of various types of diseases affecting tomato. The main goal of the data collection process is to examine and capture clear images of tomato disease. Therefore, the collected images predominantly feature lesion region, which are positioned at the center of the images.

This study involves a significant volume of images depicting tomato disease, which were acquired over a considerable time span. A total of 8,956 tomato images were collected in 4 cycles. After excluding highly blurry samples, 7,029 samples were retained, forming a comprehensive tomato diease image database for training and testing purposes in the context of tomato brown rot. This dataset consisted of 3,968 healthy tomato leaves and 3,061 tomato brown rot images. Some samples are shown in Figure 1.

To augment the sample set specifically for tomato brown rot, an additional 871 images were acquired through the implementation of a web crawler technique, yielding a cumulative count of 3,932 tomato brown rot sample images. The dataset was divided into three sets, namely the training set, validation set, and test set, in a ratio of 6:2:2. Notably, the test set was not subjected to data augmentation using the web crawler approach, and therefore, the original images were selected for the test set.

3.2 Data annotation

Supervised training is required for the convolutional neural network. Since images themselves lack labels and semantics, they need to be annotated for training purposes. In this study, professional technicians performed a thorough comparison and confirmation process. The annotation tool, LabelImg, was utilized to label the tomato brown rot images, distinguishing between healthy leaves and those affected by brown rot, as seen in Figure 2. Following the annotation process, an XML file was generated for each tomato disease image.

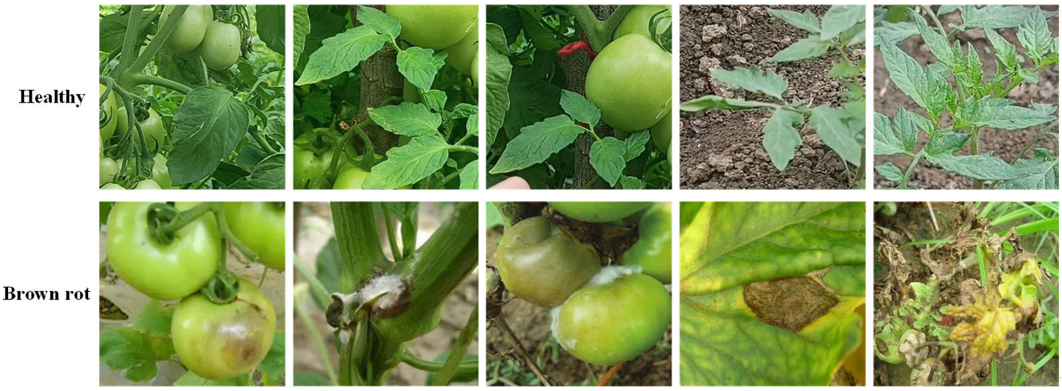


FIGURE 1
Partial Samples of the Self built Dataset.

3.3 Data enhancement

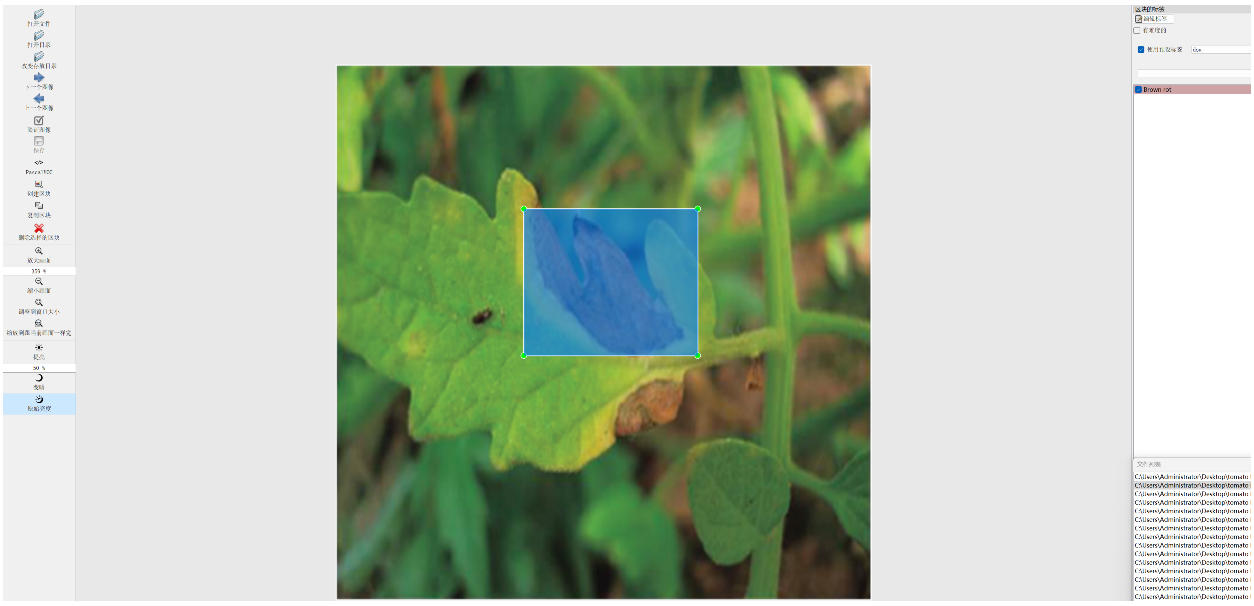
During the training of the tomato brown rot detection model, an issue arose due to the excessive number of model layers. The model tended to overlearn the details within the training data, resulting in subpar generalization capabilities and a propensity for overfitting. To tackle this issue, an image preprocessing method was employed to expand the training set and increase the sample size through random transformations. This approach aimed to enhance the model's generalization ability.

By applying these methods, the original training and verification datasets were expanded by a factor of 5, resulting in a total of 31,605 images. Importantly, the original annotations remained valid throughout the image augmentation process. Additionally, The dimensions of the images analyzed in this study were scaled to 224 × 224 pixels. Table 1 presents the distribution of the tomato disease image database.

4 Methods

4.1 YOLOv5

YOLOv5 (You Only Look Once) is one of the more advanced and mature target detection algorithms, with excellent performance in detection accuracy and speed, and more flexible network deployment.YOLOv5 has five versions, n, s, m, 1, x, with certain differences in accuracy, speed, network size, etc., as shown in Table 2, which shows that: under the condition of increasing a smaller number of parameters (Params) and computation (FLOPs),



□ 有难度的

☑ 使用预设标签 dog

☑ Brown rot

文件列表

```
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
C:\Users\Administrator\Desktop\tomato
```

FIGURE 2
Annotation interface.

TABLE 1 Detailed information of samples.

Class	Image count prior to data augmentation	Image count following data augmentation	Number of images for test
Tomato brown rot	3932	16516	786
Healthy	3968	16668	793
Total	7900	33184	1579

the accuracy of s is greatly improved compared with n and the speed remains unchanged, although the accuracy of m, 1, x is improved compared with s, but the parameters and computation volume (FLOPs) are increased. As shown in Table 1, it can be seen that: under the condition of increasing the number of parameters (Params) and the amount of computation (FLOPs), the accuracy of s is greatly improved compared to n and the speed remains unchanged, although the accuracy of m, 1, x is improved compared to s, the parameters and the amount of computation are greatly increased, therefore the YOLOv5s model is selected as the basis.

YOLOv5s combines various computer vision technologies into a small network model with fast computation speed. Refer to Figure 3 for an illustration of the network structure.

The main reasons behind the strong achievement of YOLOv5s are as follows:

1. Input: Mosaic data augmentation is employed, which involves combining images through random scaling, random cropping, and random arrangement. This technique enhances the diversity of the dataset. Additionally, adaptive anchor box calculation is employed to ascertain the ideal size of the bounding box by means of clustering. This approach contributes to improved detection speed.
2. Backbone. The YOLOv5s network model utilizes several components in its backbone architecture, including Focus (Lin et al., 2017b), CSP (Wang et al., 2020) and SPP (He et al., 2015). The Focus module performs a slicing operation that expands the feature dimension during the conversion from the input image to the feature map. CSP improves the network's learning ability while reducing memory usage. In the YOLOv5s network, ordinary images with dimensions of 3 * 608 * 608 are initially input into the network. The size of the feature map is determined after undergoing a single slicing operation using the Focus module and becomes 12 * 304 * 304. Subsequently, it is transformed into a feature map of dimensions 32 * 304 * 304 through a regular convolution operation involving 32 convolution cores.

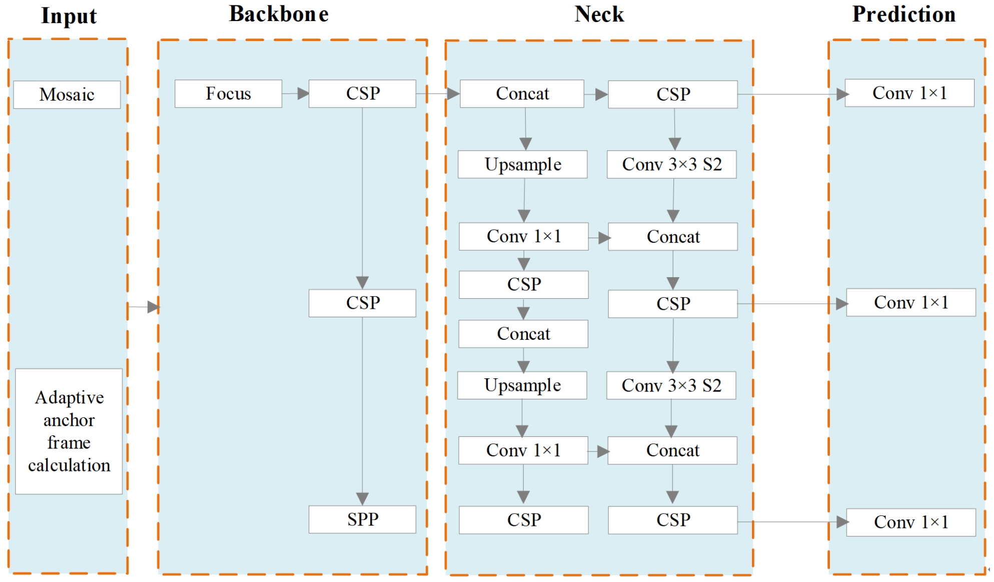


FIGURE 3
Network Structure of YOLOv5s.

TABLE 2 Different versions of YOLOv5.

Versions	mAP0.5/%	Speed/ms	Parameters size/10^6	Computational size/10^9
YOLOv5n	45.7	6.3	1.9	4.5
YOLOv5s	56.8	6.4	7.5	16.5
YOLOv5m	64.1	8.2	21.2	49
YOLOv5l	67.3	10.1	46.5	109.1

The SPP structure enables the network to handle images of varying scales, expanding its processing capabilities.

3. Neck. The neck component of the YOLOv5s network model consists of PANET (Liu et al., 2018). PANET extends the feature learning capabilities of the network by introducing additional information transmission paths based on FPN (Lin et al., 2017a). This enrichment allows for a broader range of feature learning. In contrast to the CSP structure, the neck employs a different variant. In the YOLOv5s network model, the CSP1_1 Structure and CSP1_3 Structure are employed as the backbone network. Additionally, the neck section integrates CSP2_1 to strengthen feature fusion.
4. Prediction. The prediction phase of object detection encompasses several tasks, including prediction of bounding boxes, computation of the loss function, and application of non-maximum suppression. In terms of bounding box prediction, the loss function has been enhanced from CIOU (Complete IoU) loss to generalized IoU (GIoU) loss. This modification improves the accuracy of localization. During the post-processing stage of object detection, when there is a high density of objects in certain areas of the image, the weighted NMS (Non-Maximum Suppression) method is employed to mitigate the impact of redundant bounding boxes on network parameter updates.

Although YOLOv5s offers rapid recognition, adaptive anchor boxes, and commendable precision and accuracy, it exhibits limitations in its object feature extraction capability. The existing feature fusion network primarily concentrates on high-level semantic information, resulting in a bottleneck when detecting small objects with inconspicuous features, such as tomato diseases. In order to tackle this problem, our team has developed an improved approach that fully extracts and leverages object features. This approach aims to augment the model's detection capability for small and complex objects like tomato brown rot, ultimately yielding improved detection results.

4.2 Hybrid attention module

The original algorithm treats all image regions with equal attention, which renders the network insensitive to feature discrepancies and hinders the extraction of features from small objects in the presence of complex backgrounds. To tackle this concern, the present study introduces a novel methodology, the introduction of a Hybrid attention mechanism. In scenes with intricate backgrounds and numerous small objects, the significance of various channels and spaces is simultaneously emphasized with the aim of improving the extraction capacity of features related to smaller objects. The configuration of the Hybrid attention feature augmentation module is depicted in Figure 4. The ordering of the attention modules aligns with the conclusions drawn from Hu et al (Jie et al., 2017) and Woo et al. (2018).

The Hybrid attention mechanism is an effective module designed to operate in two dimensions: channel and space. It achieves feature adaptive learning by multiplying the feature map with the attention map, the two are combined together. The Hybrid attention mechanism serves as a lightweight and versatile module capable of enhancing network representation without significantly increasing network parameters.

The main focus of the channel attention module lies in highlighting the channel-related details present in the feature map. By utilizing maximum pooling and average pooling, The spatial dimension of the feature map is compressed into a condensed representation consisting of two descriptive values. Subsequently, a shared network comprising hidden layers of multilayer perceptrons computes a channel attention map.

In contrast, the positional information of object features is the primary focus of the spatial attention module. Using maximum pooling and average pooling, two feature descriptions are obtained. These descriptions are then combined through a joining operation, Afterwards, a conventional convolution operation is employed to produce a spatial attention map.

Let $F(i,j,z) \in R^{H\times W\times C}$ represent the feature map input to the Hybrid attention module, where H denotes the length of the feature map, W represents the width of the feature map, and C indicates the channel count in the input feature map. The indices i, j, and z lie within the ranges $i \in [1,H]$, $j \in [1,W]$, $z \in [1,C]$, respectively.

Within the channel attention module, the input features undergo spatial dimension compression using both mean value pooling and maximum pooling layers. These operations aim to emphasize crucial information within the channel domain. Subsequently, the compressed feature map is fed into the perception layer. Finally, the outputs of the two feature maps are

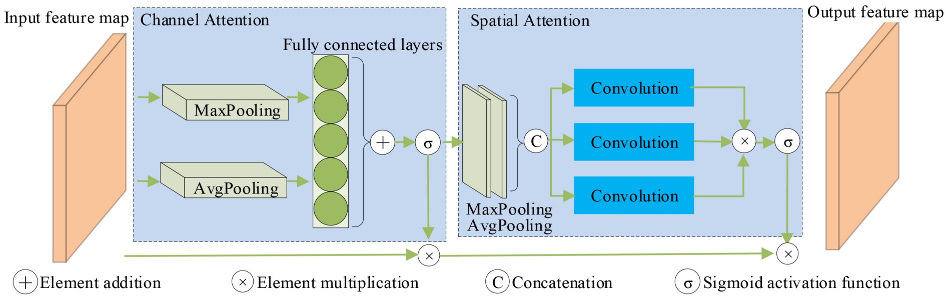


FIGURE 4
Hybrid attention module.

superimposed and passed through an activation function, yielding the channel attention weight $W_1 \in R^{1\times1\times C}$, as illustrated in formula (1).

$$W_1 = \sigma\left[\left(f_{MLP}\left(AT_{avg}\left(F(i,j)\right)\right)\right) \oplus f_{MLP}\left(AT_{max}\left(F(i,j)\right)\right)\right] \quad (1)$$

In the aforementioned formula, σ represents the sigmoid activation function, and $\oplus$ denotes the addition of corresponding elements. The terms AT_{avg} and AT_{max} correspond to the mean value pooling layer and maximum pooling layer, respectively, as depicted in formulas (2) and (3).

$$AT_{avg}(F(i,j)) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W}F(i,j) \quad (2)$$

$$AT_{max}(F(i,j)) = argmax\left(\sum_{i=1}^{H}\sum_{j=1}^{W}F(i,j)\right) \quad (3)$$

The term f_{MLP} refers to a multi-layer perceptron that consists of an adaptive convolution layer $f^{1\times m}$ and ReLU activation function, as illustrated in formula (4).

$$f_{MLP} = ReLU\left(f^{1\times1\times m}(A)\right) \quad (4)$$

In the aforementioned formula, A represents the feature matrix that is input into f_{MLP}. The term $f^{1\times1\times m}$denotes a one-dimensional convolution composed of m parameters. The relationship between m and the number of feature channels C is depicted in formula (5).

$$m = \left|\frac{\log_2(C)}{k} + \frac{b}{k}\right|_{Odd} \quad (5)$$

Since the channel dimension C is usually a multiple of 2, f_{MLP} is utilized to map the non-linear relationship between the convolution kernel size and the number of feature channels C. The value of m can be adjusted flexibly by modifying parameters b and k. If m is a non-integer, an odd number closest to m is chosen. This ensures that the anchor point of the convolution core is positioned in the middle, facilitating subsequent sliding convolution and avoiding location offset. In comparison to the fully connected layer, f_{MLP} significantly reduces the model parameters while preserving the ability to capture interaction information between channels, thus minimizing the speed impact on the original module.

The output features of the channel attention module are denoted as $F_C \in R^{H\times W\times C}$, as shown in formula (6).

$$F_C = W_1 \times F(i,j,z) \quad (6)$$

In the spatial attention module, the input feature map undergoes compression in the channel domain through mean value pooling and maximum pooling layers, respectively. This compression enhances the distinction between background and objects on the spatial domain. Subsequently, the compressed feature map is reassembled in the channel domain. Finally, the convolution layer, $f_{con}^{7\times7}$, adjusts the channel depth and feeds it into the activation function to obtain the spatial attention weight, $W_2 \in R^{H\times W\times1}$, as shown in formula (7).

$$W_2 = \sigma\left[\left(f_{con}^{7\times7}\left(AT_{avg}\left(F_C(z)\right)\right)\right) \oplus f_{con}^{7\times7}\left(AT_{max}\left(F_C(z)\right)\right)\right] \quad (7)$$

In the aforementioned formula, $f_{con}^{7\times7}$ represents the convolution kernel with a size of 7×7.

Both attention modules utilize mean value pooling and maximum pooling along the channel axis. Mean value pooling emphasizes background information on the feature map, whereas maximum pooling provides feedback to the pixel point that exhibits the highest response in the feature map, thereby highlighting object information in the image. Consequently, with the incorporation of these two pooling layers, the network becomes more sensitive to distinguishing objects from the background in the image.

The output features of the spatial attention module are denoted as $F_S \in R^{H\times W\times C}$, as shown in formula (8).

$$F_S = W_2 \times F_C \quad (8)$$

Finally, in YOLOv5s, F_S is utilized to predict the location of tomato disease objects and strengthens the network's effectiveness to learn about disease objects by selecting and weighting the transmitted features.

In Figure 5, we present the improved network architecture, featuring the integration of the Hybrid attention module just before the prediction component of YOLOv5s. This alteration empowers the network to make object predictions using the global attention map created by the attention module. Given that the original YOLOv5s network incorporates numerous residual links in its feature extraction section, it's essential to replace all of these residual links with the attention module.

4.3 The improved loss function

To enhance the accuracy of model positioning, the bounding box loss function in YOLOv5s incorporates GIOU_LOSS. This ensures that even when the predicted box and the real box do not intersect, GIOU_LOSS can predict their distance, overcoming the limitations of IOU_LOSS. However, the GIOU_LOSS algorithm encounters an issue where the position of the prediction box cannot be determined if it is entirely contained within the true box (i.e., A∩B=B).

Therefore, this study examines the influence of the center point distance and aspect ratio of detection and labeling bounding boxes on the basis of the overlapping area. For object detection tasks, the regression loss function used is CIOU_LOSS. CIOU_LOSS considers the intersection area and distance between the central points of the predicted box and the object box. In the event that the object box encloses the prediction box, the separation between the two boxes is directly measured. It also considers the center point distance of the bounding box and the scale information of the width-height ratio of the bounding box. Furthermore, the ratio between the length and width of the prediction box and the object box is taken into consideration to improve the quality of the bounding regression result. Figure 6 illustrates the schematic diagram of CIOU.

Let's assume that the diagonal distance of the minimum bounding rectangle C is represented by $Distance_1$, and the distance between the center point of the object's true box and the prediction box is represented by $Distance2$. The loss function used

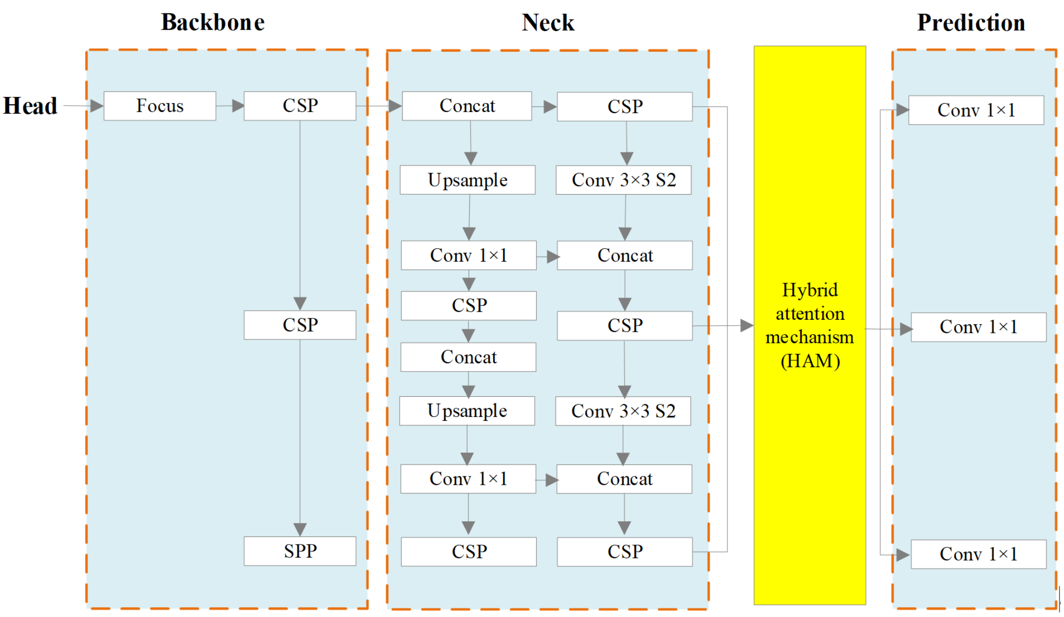


FIGURE 5
The improved network structure.

in this study is CIOU_LOSS, as shown in formula (9).

$$CIOU_{Loss} = 1 - CIOU = 1 - \left(IOU - \frac{Distance_2^2}{Distance_1^2} - \frac{\gamma^2}{(1 - IOU) + \gamma} \right) \quad (9)$$

In the above-mentioned formula, $\gamma$is a parameter that measures the consistency of the aspect ratio of the object prediction box. It is calculated as shown in formula (10).

$$\gamma = \frac{4}{\pi^2} \left(arctan \frac{w^{gt}}{h^{gt}} - arctan \frac{w^p}{h^p} \right)^2 \quad (10)$$

In the above-mentioned formula, w^{gt} and h^{gt} represent the width and height of the object bounding box, while w^p and h^p represent the width and height of the prediction bounding box.

5 Experimental design

5.1 Experimental operation environment

The experimental setup for this study utilized the following components: PaddlePaddle 2.4.0 as the deep learning framework, an Intel Core i7 8700 K CPU, 32 GB of memory, and an NVIDIA GeForce GTX 1070 GPU. The programming language employed was Python.

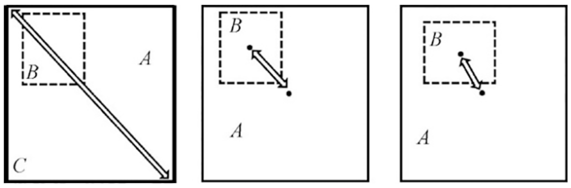


FIGURE 6
CIOU schematic diagram.

5.2 Evaluating indicator

The performance of the enhanced YOLOv5 algorithm is assessed using several evaluation indicators including average accuracy (AP), mean Average Precision (mAP), F1 score, and detection rate. AP represents the average accuracy across different recall rates, while mAP is the average sum of AP values. The F1 score is a measure of the harmonic mean between accuracy and recall. Additionally, the detection rate is calculated as the number of frames per second (FPS) that the model processes, reflecting both the time complexity and the size of the model's parameters.

AP, mAP, and F1 scores are expressed as shown in formular (11), (12) and (13), respectively.

$$AP = \int_0^1 P(R)d(R) \quad (11)$$

$$mAP = \frac{\sum_{k=0}^{C} AP_k}{C} \quad (12)$$

$$F1 = \frac{2PR}{P + R} \quad (13)$$

In the above-mentioned formula, $\sum_{k=0}^{C} AP_k$ represents the average accuracy of each category, where C is the total number of categories. P (Precision) represents the accuracy, and R (Recall) represents the recall rate. The formulas for P and R are shown as follows:

$$P = \frac{TP}{TP + FP} \quad (14)$$

$$R = \frac{TP}{TP + FN} \tag{15}$$

In the above-mentioned formula, in the case of detecting Tomato Brown Rot disease, TP (True Positive) indicates the count of accurately identified instances, FP (False Positive) represents the count of incorrectly identified instances, and FN (False Negative) corresponds to the count of undetected instances.

5.3 Model training

During the model training stage, we utilized an attenuation coefficient of 0.0005, conducted 10,000 iterations, and initialized the learning rate at 0.001. At the 2,000th and 3,000th iterations, we adjusted the learning rate to 0.0001 and 0.00001, respectively. Convergence was achieved after approximately 3,000 iterations, as illustrated in Figure 7, depicting the loss function and accuracy.

Based on the performance evaluation results depicted in Figure 5, it can be concluded that the enhanced YOLOv5 model exhibits favorable outcomes during the training phase.

6 Analysis of experimental results

6.1 Qualitative analysis

To ensure a comprehensive assessment of the algorithm's generalization capabilities and avoid biased conclusions from a single training-validation-test split, we employ multifold cross-validation. Figure 8 illustrates the test results, revealing the model's effectiveness in real-world scenarios. The image sequences in Figure 8 display the detection results at early, middle, and late stages of tomato brown rot. More detailed experimental results can be found in Table 3.

Based on the aforementioned results, the enhanced object detection algorithm utilizing YOLOv5 achieves a detection accuracy of 93.2% for tomato brown rot disease and 96.5% for healthy instances, respectively. Additionally, the processing speed reaches 112 FPS. These results demonstrate that the proposed model accurately detects tomato brown rot with excellent efficacy.

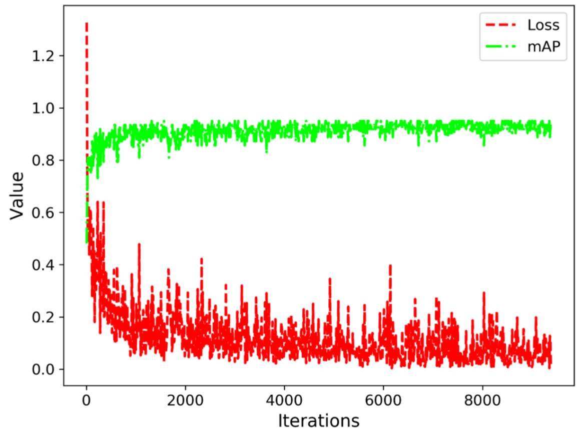


FIGURE 7
Performance Evaluation of Model Training Process.

6.2 Quantitative analysis

To gauge the effectiveness of the enhanced YOLOv5s algorithm, we conducted a comparative analysis against widely used object detection algorithms, including Faster RCNN, FCOS, YOLOX, EfficientDet, YOLOv4-tiny, and YOLOv5s. We ensured uniformity in the training platform, configuration details, and dataset throughout all experiments. Each algorithm was trained and applied to detect the same set of images, enabling a comprehensive performance evaluation. The outcomes of this comparative analysis are presented in Table 4.

Table 4 clearly illustrates that Faster RCNN falls short in all parameters, especially with extended inference time and a lower frame rate, making it unsuitable for deployment on edge devices. On the other hand, YOLOv5s stands out as one of the most favored target detection algorithms, striking a commendable balance between accuracy and speed.

In our study, we've harnessed the proposed method, which outperforms other algorithms. It enhances both average accuracy and detection speed, surpassing the original YOLOv5s algorithm by 4.8% and the Faster R-CNN algorithm by 4.3%. While YOLOX boasts an impressive 119 frames per second (FPS), which slightly exceeds our method's 112 FPS, it sacrifices detection accuracy due to its limited capability to detect small targets. This improvement in detection accuracy primarily results from the introduction of the hybrid attention module, enhancing feature learning in the disease target region, and utilizing CIoU as the loss function for edge regression, which elevates edge regression accuracy.

As a result, our enhanced algorithm excels in the complex detection of tomato brown rot. The comprehensive results highlight that, compared to other advanced algorithms, our method strikes a superior balance between detection accuracy and speed in the task of tomato brown rot detection, meeting the real-time detection requirements of edge-end devices. Clearly, our method exhibits distinct advantages over other detection models, firmly aligning with the needs of online tomato brown rot detection tasks.

6.3 Ablation experimental analysis

In order to comprehensively assess the effectiveness of the proposed methodology, an extensive experimental analysis was conducted. These experiments aimed to assess how different improvement modules impact the detection performance. The model's performance was evaluated based on two key aspects: the precision of object detection and the rate of detection. The results of these comparisons are provided in Table 5.

Our results highlight the substantial enhancement in model detection accuracy achieved by incorporating the Hybrid attention module, marking a notable increase of 3.4%. This improvement

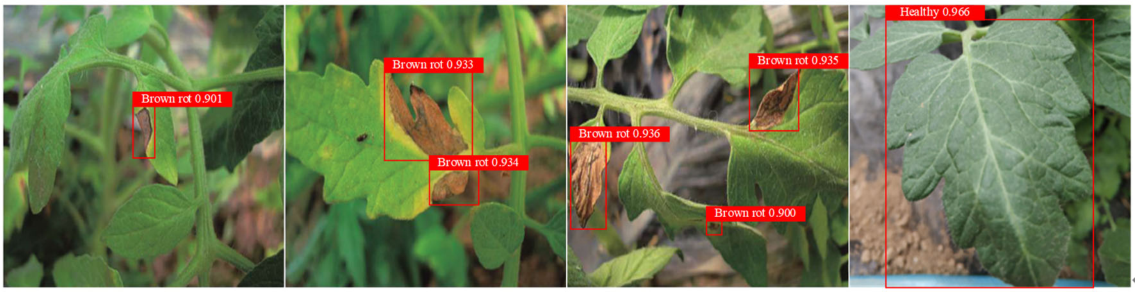


FIGURE 8
The visual representation of the detection approach in this study.

TABLE 3 Detection results using the proposed algorithm.

Class	AP/%	Precision/%	Recall/%	FPS
Brown rot	93.2	93.5	92.7	112
Healthy	96.5	96.6	95.8	112

TABLE 4 Detection outcomes obtained from diverse algorithms.

Algorithms	Backbone	mAP(%)	FPS(Frame/second)
Faster R-CNN	ResNet50	90.3	8
FCOS	ResNet101	85.7	16
YOLOX	CSPDarkNet53	86.9	119
EfficientDet	EfficientNet-B2	87.8	44
YOLOv4-tiny	DarkNet53	88.7	98
YOLOv5s	CSPDarkNet53	89.8	118
The proposed algorithm	CSPDarkNet53	94.6	112

comes with minimal impact on detection speed, with only a marginal decrease of 6 frames per second. These findings underscore the effectiveness of the proposed Hybrid attention module in judiciously assigning network learning weight to object-rich areas.

Furthermore, through the optimization of the loss function, we observe a further 3.7% increase in detection accuracy while maintaining consistent detection speed. This underlines the success of the proposed loss function in prioritizing high-score prediction bounding boxes and diminishing the influence of redundant bounding boxes during subsequent screening.

Considering the assessment of model accuracy and speed, it is evident that the proposed model strikes a commendable balance between detection precision and efficiency. This makes it well-suited for deployment on resource-constrained embedded systems.

7 Application prospect

The model developed in this study demonstrates remarkable accuracy and holds profound significance in several key areas. It contributes to the formulation of effective disease prevention

TABLE 5 Results of ablation experiments on tomato brown rot disease object detection.

Strategies	Hybrid attention mechanism	The improved loss function	mAP(%)	FPS(Frame/second)
1	×	×	89.8	118
2	√	×	93.2	112
3	×	√	93.5	119
4	√	√	94.6	112

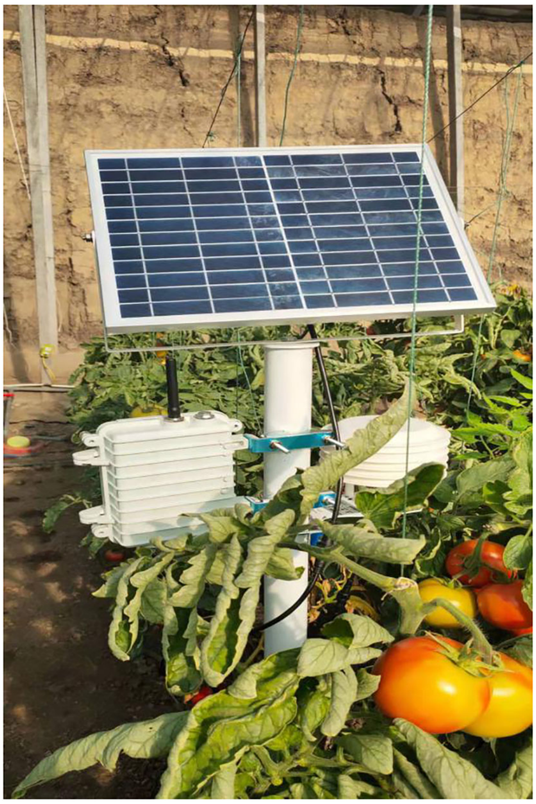


FIGURE 9
Tomato Greenhouse Internet of Things Equipment.

strategies, improves tomato yield and quality, reduces the cost of on-site diagnosis of tomato diseases, and offers a scientific basis for the creation of intelligent pesticide spray robots.

The results of this study readily translate into real-time disease identification, facilitating precise prevention and control measures while minimizing economic losses caused by diseases. We have already established the infrastructure for the Tomato Greenhouse Internet of Things equipment, as shown in Figure 9. This infrastructure provides a strong foundation for the future implementation of an integrated system for the detection and prevention of tomato diseases through intelligent control. Furthermore, it sets the stage for ongoing disease inspection and monitoring within greenhouses, employing continuous video surveillance.

8 Conclusions and future directions

8.1 Conclusions

In this study, we harnessed a neural network model for the precise detection and localization of tomato brown rot disease. We introduced a novel hybrid attention module into the feature prediction structure of YOLOv5, while refining the loss function. Our experimental findings unequivocally confirm the efficacy of our proposed approach. Notably, our method outperforms other cutting-edge object detection algorithms when it comes to identifying tomato brown rot in a greenhouse environment.

While it’s true that our algorithm’s detection speed is marginally slower than the original YOLOv5, this trade-off is well-justified by its superior average accuracy, surpassing both the original YOLOv5 and faster RCNN algorithms. The algorithm we’ve developed here is eminently practical and can be seamlessly integrated into tomato disease detection systems. It empowers precise, real-time disease identification, particularly beneficial for vegetable growers and individuals who lack comprehensive disease knowledge. This, in turn, facilitates the timely implementation of effective preventive and control measures, thereby minimizing economic losses.

8.2 Future directions

This study introduced a tomato brown rot detection algorithm, bolstering the accuracy of disease identification. While progress has been made, several areas warrant further exploration and resolution. Future research can focus on:

1. Enhancing CNN Structures: Investigate and optimize convolutional neural network (CNN) structures, continually innovating and incorporating high-capacity models for improved disease detection accuracy.
2. Early Disease Recognition: Develop recognition algorithms and models for early disease identification, especially in complex backgrounds. This research aims to boost the efficiency and precision of tomato brown rot disease detection, enabling timely disease prevention and control.

3. Federated Learning Integration: Explore the potential of integrating federated learning into our research to further enhance disease detection results, ultimately providing more effective tools for disease identification in agriculture.

By pursuing these research directions, we can advance the field of tomato disease detection, contributing to comprehensive and effective disease management.

Data availability statement

The dataset and code in this study can be accessed by contacting the corresponding author.

Author contributions

JL: Funding acquisition, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. XW: Funding acquisition, Conceptualization, Investigation, Software, Writing – original draft, Writing – review & editing. QZ: Writing – review & editing. WM: Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. The present study receives support from the Shandong Province Natural Science Foundation (Grant No. ZR2021QC173 and ZR2023MF048), Key R&D Plan of Shandong Province (Soft Science Project) with project number 2023RKY02013.

Conflict of interest

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

Publisher's note

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

References

Agarwal, M., Gupta, S. K., and Biswas, K. K. (2020). Development of efficient cnn model for tomato crop disease identification. Sustainable Computing: Informatics and Systems 28.

Aggarwal, M., Khullar, V., Goyal, N., Alammari, A., Albahar, M. A., and Singh, A. (2023). Lightweight federated learning for rice leaf disease classification using non independent and identically distributed images. *Sustainability* 15 (16), 12149. doi: 10.3390/su151612149

Atila, M., Uar, M., Akyol, K., and Uar, E. (2021). Plant leaf disease classification using efficientnet deep learning model. *Ecol. Inf.* 61, 101182. doi: 10.1016/j.ecoinf.2020.101182

Bochkovskiy, A., Wang, C. Y., and Liao, H. (2020). *Yolov4: optimal speed and accuracy of object detection.*

Bollis, E., Pedrini, H., and Avila, S. (2020). Weakly supervised learning guided by activation mapping applied to a novel citrus pest benchmark. *IEEE.* doi: 10.1109/CVPRW50498.2020.00043

Chen, J., Chen, J., Zhang, D., Sun, Y., and Nanehkaran, Y. A. (2020).). Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* 173, 105393.

Demir, K., and Vedat, T. (2021). Drone-assisted automated plant diseases identification using spiking deep conventional neural learning. *Ai. Commun.* 1.

Gautam, V., Trivedi, N. K., Singh, A., Mohamed, H. G., Noya, I. D., Kaur, P., et al. (2022). A transfer learning-based artificial intelligence model for leaf disease assessment. *Sustainability* 14 (20), 13610.

Ge, Z., Liu, S., Wang, F., Li, Z., and Sun, J. (2021). Yolox: Exceeding yolo series in 2021. *arXiv. preprint. arXiv:2107.08430.*

Girshick, R., Donahue, J., Darrelt,, et al. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 580–587.

He, X., Fang, K., Qiao, B., Zhu, X., and Chen, Y. (2021). Watermelon disease detection based on deep learning. *Int. J. Pattern Recognition. Artif. Intell.* 35 (05), 96–107.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask RCNN. IEEE transactions on pattern analysis & Machine intelligence. *IEEE.*

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *Pattern Anal. Mach. Intell. IEEE Trans.* 37 (9), 1904–1916. doi: 10.1109/TPAMI.2015.2389824

Horvath, D. M. (2018). Putting science into action to address threats to food security caused by crop diseases. *Outlooks. Pest Manage.* doi: 10.1564/v29_jun_07

Jain, S., and Gour, M. (2021). Tomato plant disease detection using transfer learning with c-gan synthetic images. *Comput. Electron. Agric.* 187 (2021).

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

Jie, H., Li, S., Gang, S., and Albanie, S. (2017). Squeeze-and-excitation networks. *IEEE Trans. Pattern Anal. Mach. Intell.* PP (99).

Joshi, B. M., and Bhavsar, H. (2020). Plant leaf disease detection and control: a survey. *J. Inf. Optimization. Sci.* 41 (2), 475–487. doi: 10.1080/02522667.2020.1734295

Ju, M., Luo, J., Wang, Z., and Luo, H. (2021). Adaptive feature fusion with attention mechanism for multi-scale target detection. *Neural Computing. Appl.* 33 (7). doi: 10.1007/s00521-020-05150-9

Lee, S., Lin, S., and Chen, S. (2020). Identification of tea foliar diseases and pest damage under practical field conditions using a convolutional neural network. *Plant Pathol.* 69. doi: 10.1111/ppa.13251

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., et al. (2022). YOLOv6: A single-stage object detection framework for industrial applications. *arXiv. preprint. arXiv:2209.02976.*

Liaquat, F., Liu, Q., Arif, S., Shah, I. H., and Munis, M. (2019). First report of brown rot caused by rhizopus arrhizus on tomato in Pakistan. *J. Plant Pathol.* 101 (4). doi: 10.1007/s42161-019-00320-8

Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). Feature pyramid networks for object detection. 2017 IEEE conference on computer vision and pattern recognition (CVPR). *IEEE Comput. Soc.* doi: 10.1109/CVPR.2017.106

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). Focal loss for dense object detection. IEEE transactions on pattern analysis & Machine intelligence (Vol.PP, pp.2999-3007). *IEEE.* doi: 10.1109/ICCV.2017.324

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., et al. (2016). "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision* (Cham: Springer).

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). Path aggregation network for instance segmentation. *2018. IEEE/CVF. Conf. Comput. Vision Pattern Recognition. (CVPR)*. doi: 10.1109/CVPR.2018.00913

Mensah, P. K., Weyori, B. A., and Ayidzoe, M. A. (2020). Gabor capsule network for plant disease detection. *Int. J. Adv. Comput. Sci. Appl.* 11 (10), 388–395.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: unified, real-time object detection. Computer vision & Pattern recognition. *IEEE*. doi: 10.1109/CVPR.2016.91

Redmon, J., and Farhadi, (2018). Yolov3: an incremental improvement. *arXiv. e-prints*.

Redmon, J., and Farhadi, A. (2017). YOLO9000: better, faster, stronger. IEEE conference on computer vision & Pattern recognition (pp.6517-6525). *IEEE*. doi: 10.1109/CVPR.2017.690

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster RCNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (6), 1137–1149. doi: 10.1109/TPAMI.2016.2577031

Review, B. (1991). Compendium of tomato diseases. *Mycologia* 84 (1), 133.

Temniranrat, P., Kiratiratanapruk, K., Kitvimonrat, A., Sinthupinyo, W., and Patarapuwadol, S. (2021). A system for automatic rice disease detection from rice paddy images serviced via a chatbot. . *Comput. Electron. Agric.* doi: 10.1016/j.compag.2021.106156

Terven, J., and Cordova-Esparza, D. (2023). A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond. *arXiv. preprint. arXiv:2304.00501*.

Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *In Proc. IEEE/CVF. Conf. Comput. Vision Pattern Recognition*. pp, 7464–7475). doi: 10.1109/CVPR52729.2023.00721

Wang, C. Y., Liao, H., Wu, Y. H., Chen, P. Y., and Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. 2020 IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW). *IEEE*. doi: 10.1109/CVPRW50498.2020.00203

Wang, D., Wang, J., Li, W., and Guan, P. (2021). T-CNN: Trilinear convolutional neural networks model for visual detection of plant diseases. *November. 2021.Computers. Electron. Agric.* 190 (1), 106468. doi: 10.1016/j.compag.2021.106468

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). "CBAM: convolutional block attention module," in *European conference on computer vision* (Cham: Springer).

Wu, B., Liang, A., Zhang, H., Zhu, T., and Su, J. (2021). Application of conventional uav-based high-throughput object detection to the early diagnosis of pine wilt disease by deep learning. *For. Ecol. Manage.* 486 (2), 118986. doi: 10.1016/j.foreco.2021.118986

Yang, G., Chen, G., Li, C., Fu, J., and Liang, H. (2021). Convolutional rebalancing network for the classification of large imbalanced rice pest and disease datasets in the field. *Front. Plant Sci.* 12, 671134. doi: 10.3389/fpls.2021.671134

Yohanandan, S., Song, A., Dyer, A. G., and Tao, D. (2018). Saliency preservation in low-resolution grayscale images. doi: 10.1007/978-3-030-01231-1_15

YOLOv5 (2021). Available at: https://github.com/ultralytics/yolov5.

Zhang, Y., Song, C., and Zhang, D. (2020). Deep learning-based object detection improvement for tomato disease. *IEEE Access* PP (99), 1–1. doi: 10.1109/ACCESS.2020.2982456

Zhu, H., Xie, C., Fei, Y., and Tao, H. (2021). Attention mechanisms in cnn-based single image super-resolution: a brief review and a new perspective. *Electronics* 10 (10), 1187. doi: 10.3390/electronics10101187

TYPE Original Research
PUBLISHED 30 January 2024
DOI 10.3389/fpls.2023.1330527

OPEN ACCESS

EDITED BY
Dun Wang,
Northwest A&F University, China

REVIEWED BY
Olarik Surinta,
Mahasarakham University, Thailand
Guoxiong Zhou,
Central South University Forestry and Technology, China

*CORRESPONDENCE
Wei Yao
YaoWei-hebau@hotmail.com

RECEIVED 31 October 2023
ACCEPTED 28 December 2023
PUBLISHED 30 January 2024

CITATION
Liu B, Wei S, Zhang F, Guo N, Fan H and Yao W (2024) Tomato leaf disease recognition based on multi-task distillation learning.
Front. Plant Sci. 14:1330527.
doi: 10.3389/fpls.2023.1330527

Tomato leaf disease recognition based on multi-task distillation learning

Bo Liu[1,2], Shusen Wei[1,2], Fan Zhang[1,2], Nawei Guo[1,2], Hongyu Fan[1,2] and Wei Yao[1,2*]

[1]College of Information Science and Technology, Hebei Agricultural University, Baoding, China,
[2]Hebei Key Laboratory of Agricultural Big Data, Baoding, China

Introduction: Tomato leaf diseases can cause major yield and quality losses. Computer vision techniques for automated disease recognition show promise but face challenges like symptom variations, limited labeled data, and model complexity.

Methods: Prior works explored hand-crafted and deep learning features for tomato disease classification and multi-task severity prediction, but did not sufficiently exploit the shared and unique knowledge between these tasks. We present a novel multi-task distillation learning (MTDL) framework for comprehensive diagnosis of tomato leaf diseases. It employs knowledge disentanglement, mutual learning, and knowledge integration through a multi-stage strategy to leverage the complementary nature of classification and severity prediction.

Results: Experiments show our framework improves performance while reducing model complexity. The MTDL-optimized EfficientNet outperforms single-task ResNet101 in classification accuracy by 0.68% and severity estimation by 1.52%, using only 9.46% of its parameters.

Discussion: The findings demonstrate the practical potential of our framework for intelligent agriculture applications.

KEYWORDS

multi-task learning, knowledge distillation, tomato leaf diseases, disease classification, severity prediction

1 Introduction

Tomato is one of the most widely cultivated vegetables in the world, with its versatility extending to various applications such as a culinary ingredient (Kumar et al., 2022), an industrial raw material (Botineştean et al., 2015), a component in cosmetics (Septiyanti and Meliana, 2020), and medicinal uses (Kumar et al., 2012). However, tomato diseases can rapidly spread through a field if not identified and managed in a timely manner, leading to substantial losses in both yield and quality of the crop (Zhang et al., 2022). As symptoms of many tomato diseases can appear on the leaves, leveraging computer vision techniques for automated recognition of leaf diseases has attracted widespread attention from researchers (Boulent et al., 2019; Habib et al., 2020; Nanehkaran et al., 2020; Roy and Bhaduri, 2021; Albahli and Nawaz, 2022; Harakannanavar et al., 2022). Although these techniques effectively improve the accuracy and speed of disease diagnosis, they also present challenges. These include variations in disease symptoms and lighting conditions (Zhang et al., 2018a), difficulty in collecting enough disease samples (Zhang et al., 2021), varying levels of disease severity (Wang et al., 2021), and limitations in computing power (Bi et al., 2022). Such factors potentially influence the applicability of the learning models.

Most of the computer vision-based leaf disease recognition methods are mainly divided into two categories: hand-crafted feature-based methods and deep learning-based methods. Traditionally, hand-crafted features refer to the manual extraction of specific features such as textures, colors, shapes, and sizes from leaf images. These features are then used as input for training a classifier to identify the presence of plant diseases. The utilization of classical classifiers, such as support vector machines (SVM) (Cortes and Vapnik, 1995) and random forests (RF) (Breiman, 2001), has been instrumental in leaf disease identification, owing to their robust nature in handling high-dimensional, noisy, and missing data (Patil et al., 2017). Consequently, the research community has significantly focused on developing improved methods for feature extraction to enhance recognition performance. Mokhtar et al. (2015) employed geometric features and histogram features for classifying two tomato leaf viruses, achieving the highest accuracy of 91.5% using the Quadratic kernel function. Meenakshi et al. (2019) improved plant leaf disease identification using exact Legendre moments shape descriptors, with a high accuracy of 99.1% on three tomato diseases (early and late blight and mosaic). In Rahman et al. (2022), texture features from tomato leaf images were analyzed using a gray level co-occurrence matrix (GLCM). In addition to single-type features, hybrid features have been well-studied. Sharif et al. (2018) proposed a hybrid method for automatic detection and classification of six types of diseases in citrus plants, which used color, texture, and geometric features combined in a codebook and selected by PCA score, entropy, and skewness-based covariance vector before being fed to a multi-class SVM. Similarly, Basavaiah and Arlene Anthony (2020) recognized four main diseases in tomato plants through the fusion of multiple features, including color histograms, Hu Moments, Haralick, and local binary pattern, resulting in 94% accuracy achieved by a RF classifier. In summary, hand-crafted feature-based methods are highly valued for their simplicity and interpretability, as well as they have demonstrated satisfactory performance on small to medium-sized datasets. However, they struggle to scale up large and diverse datasets, and fall short in coping with biases and noises in the data distribution, leading to decreased accuracy and robustness in real-world applications.

Recently, deep learning has revolutionized the field of computer vision, resulting in significant improvements in detecting leaf diseases (Sujatha et al., 2021; Shoaib et al., 2022). For instance, a novel tomato leaf disease recognition framework was proposed, which used binary Wavelet transform for image preprocessing to remove noise, and both-channel residual attention network (B-ARNet) for identification (Sujatha et al., 2021). Other types of attention mechanisms are also incorporated to enhance the model's recognition capability. In Zhao et al. (2021), to adaptively recalibrate channel-wise feature responses, a squeeze-and-excitation (SE) module (Hu et al., 2018) is integrated into a ResNet50 network (He et al., 2016), with an average identification accuracy of 96.81% on the publicly available PlantVillage dataset (Hughes et al., 2015).

Additionally, Bhujel et al. (2022) compared the performance and computational complexity of different attention modules and found that the convolutional block attention module (CBAM) (Woo et al., 2018) was the most effective in enhancing classification performance, resulting in an average accuracy of 99.69%. Despite the successes of these deep learning-based methods, they face limitations such as the need for large amounts of labeled data and substantial computational resources. To address these challenges, researchers have proposed a series of strategies for constructing lightweight networks, such as depthwise separable convolutions (MobileNet (Howard et al., 2017)), channel shuffling (ShuffleNet (Zhang et al., 2018a)), and a combination of network scaling and architecture search (EfficientNet (Tan and Le, 2019)). For example, Zeng et al. (2022) developed a lightweight CNN model named LDSNet, which uses an improved dense dilated convolution (IDDC) block and coordinated attention scale fusion (CASF) mechanism to identify corn leaf diseases in complex backgrounds. Similarly, Janarthan et al. (2022) utilized a simplified MobileNetV2 architecture and an empirical method for creating class prototypes, requiring low processing power and storage space. Li et al. (2023) explored a hybrid transformer-based architecture by integrating shuffle-convolution and a lightweight transformer encoder. While compact models achieve computational efficiency gains by reducing the parameters, these gains may come at the cost of decreased accuracy (Atila et al., 2021; Thai et al., 2023).

In addition to identifying the presence of a plant disease, it is also crucial to estimate the severity of the disease, providing a quantitative assessment for disease diagnosis (Ilyas et al., 2022; Ji and Wu, 2022). The precise localization, size, and distribution of infected regions in plant leaves can significantly enhance the accuracy of disease classification, especially in field images with complex backgrounds (Barbedo, 2019). Moreover, these factors are vital for severity grading, disease progression monitoring, and assessment of treatment efficacy. The process of estimating the level of leaf diseases often involves two main steps: segmentation and grading. Segmentation refers to the operation of separating infected regions from healthy areas of the leaf or plant. This can be achieved through various methods such as morphological operations (Gupta, 2022), k-means clustering and thresholding (Karlekar and Seal, 2020; Singh et al., 2021), and deep learning-based semantic segmentation (Wang et al., 2021; Liu et al., 2022; Deng et al., 2023). Grading then assigns a numerical score or rating to the severity of the disease, based on proportional area measurement (Wu et al., 2022) or ordinal categories (Ozguven and Adem, 2019; Pal and Kumar, 2023). Considering the complementary nature of disease classification and severity estimation, there is an emerging trend toward multi-task learning. This approach aims to jointly optimize both tasks by leveraging shared representations and correlations between them. For example, Ji et al. (2020) presented a set of binary relevance-CNNs that can simultaneously recognize 7 crop species, classify 10 crop diseases (including healthy), and estimate 3 disease severity levels, achieving the best test accuracy of 86.70% for recognition and 92.93% for severity estimation. Other techniques, such as alternating training (Jiang et al., 2021) and weighting adjustment (Wang et al., 2022), have been explored to enhance the accuracy of the combined task. Although multi-task learning can lead to better performance than individual tasks, it may also introduce increased computational effort and suboptimal solutions due to the difficulty in balancing tasks.

To address these challenges, we propose a novel multi-task distillation learning framework for tomato leaf disease diagnosis (MTDL). Unlike traditional distillation learning (Hinton et al., 2015) that relies on one-to-one and one-way knowledge transfer from a teacher model to a student model. Instead, our framework considers tomato disease category identification and severity prediction as a multi-task model that can be optimized simultaneously, as well as two single-task models that can be mutually informative. Accordingly, we develop a learning process for knowledge decoupling and reorganization, facilitating the efficient transfer of knowledge between the two tasks. Furthermore, this process is designed to be integrated with deep networks of varying complexity and architecture, making it adaptable to different disease identification scenarios with diverse computational power configurations and performance requirements.

Specifically, MTDL uses a multi-task model that contains disease classification and severity estimation as the baseline. It adopts a multi-stage learning strategy, including knowledge disentanglement, single-task mutual learning, and knowledge integration, In this process, the goal of knowledge disentanglement is to transfer the shared knowledge from the original multi-task model to the corresponding single-task models. This enables the specialization of task-specific models and avoids negative transfer of knowledge between tasks. For mutual learning between tasks, the goal is to fully exploit the complementarity between different learning objectives. Finally, through knowledge integration, the disentangled and mutually learned knowledge components are re-combined and unified to produce the refined high-quality multi-task model.

Furthermore, considering that multi-stage distillation learning will lead to a dependency of the current student model on the teacher model from the previous stage, we propose a decoupled teacher-free knowledge distillation (DTF-KD) strategy to simplify the training process. DTF-KD introduces a virtual teacher, replacing the traditional teacher model in the distillation process. This approach allows for increased adaptability by applying different learning intensities to target and non-target knowledge. In the context of the classification problem addressed in this paper, the target knowledge corresponds to the correct classification assignment of the ground truth.

The main contributions of this paper are summarized as follows:

1. We propose a novel multi-task distillation learning (MTDL) framework for leaf disease identification. This framework progressively decomposes and integrates the inherent knowledge from two tasks: tomato disease classification and severity prediction, through a distillation process, thereby generating a robust multi-task model for comprehensive disease diagnosis.
2. We propose a decoupled teacher-free knowledge distillation (DTF-KD) method to simplify MTDL by reducing the reliance on teacher models during the learning process. A virtual teacher is introduced to guide the learning process by providing separate instructions for the correct class and non-correct classes.
3. The experimental results demonstrate that the proposed framework effectively leverages the complementary characteristics of tomato disease category identification and severity prediction, reducing the model size while improving the performance.

2 Materials and methods

2.1 Dataset

The dataset employed in this study is aggregated from three distinct sources.The first source is drawn from the AI Challenger 2018 Crop Leaf Disease Challenge (Dataset AI Challenger, 2018), encompassing 11 types of plants and 27 types of diseases. Some of these diseases are further categorized into general and severe degrees, resulting in a total of 61 categories. Specifically, the dataset includes instances of leaf diseases for the following plants: apple (2,765), grape (3,144), peach (2,146), potato (3,246), citrus (4,577), pepper (1,929), strawberry (1,263), cherry (939), maize (3,514), pumpkin (1,465), and tomato (11,610). For the purposes of our study, we focus on the tomato subset. However, as the dataset contains only three samples of Canker disease, we decide to exclude this category from our analysis. The second source, the PlantDoc dataset (Singh et al., 2020), consists of 2,598 data samples that involve 13 types of plants and 27 categories (17 diseases, 11 healthy). These samples were mainly obtained from the internet and manually annotated, with the tomato subset containing 8 categories. The third source is the Taiwan Tomato Disease dataset (Huang and Chang, 2020), which is originally comprising 622 samples, was first employed in the study detailed in Thuseethan et al. (2022). In addition, it encompasses six distinct categories, namely Bacterial Spotted (110), Leaf Mold (67), Gray Spot (84), Health (106), Late Blight (98), and Powdery Mildew (157). We choose this dataset for its diverse disease conditions and combine it with larger datasets like AI Challenger 2018 and PlantDoc to further enrich the diversity of our data. Figure 1 shows examples of different tomato leaf diseases.

2.2 Data preprocessing

For the AI Challenger dataset, given the scarcity of data for the canker disease category (only 3 instances), we excluded this data. The dataset provided severity labels for most of the data, categorized into three levels: healthy, moderate, and severe. In addition, we supplemented the dataset with severity labels for the tomato spotted wilt virus. For the PlantDoc dataset, due to the complexity of the leaf background, we manually cropped the tomato leaf subset to meet the needs of the disease identification task. Each image was cropped to retain the main area of a single leaf while preserving some background information from the plant. For the Taiwan Tomato dataset, we used all the original data. For all three datasets, we applied consistent severity labeling. Specifically, we hired five agricultural experts to manually annotate the severity of the disease. The final severity level was determined by a majority vote. Table 1 summarizes the information about the three datasets used in this study.

We divide the dataset into training, validation, and test sets in an 8:1:1 ratio, ensuring a balanced and representative distribution for each set. The division is performed randomly to maintain fairness and diversity. Furthermore, we rigorously validate both the results reported in the paper and the determination of hyperparameters through 10-fold cross-validation.

2.3 Multi-task distillation framework

The proposed MTDL for tomato leaf disease diagnosis is comprised of three components: two single-task models, one for disease recognition and the other for severity prediction, and a hybrid model that integrates these two tasks. As illustrated in Figure 2, the MTDL pipeline enables mutual knowledge transfer between the two individual tasks, facilitating knowledge disentanglement and integration to enhance performance. In traditional distillation learning processes (Hinton et al., 2015), a powerful teacher model transfer knowledge to a lightweight student model. However, our MTDL framework emphasizes bidirectional knowledge transfer between teacher and student models, allowing for greater flexibility in their selection.

2.3.1 Problem formulation

Given a leaf disease dataset $D = \{(\mathrm{x}_i, y_i^c, y_i^s)\}_{i=1}^N$ containing N images, where $\mathrm{x_i} \in \mathbb{R}^{C\times H\times W}$ is the i-th leaf image with C, H, and W denoting the number of channels, height, and width of the image, respectively. Each image is labeled with two types of annotations: $y_i^c \in \{1, 2, \cdots, K^c$ is the disease category label, with K^c being the number of disease categories, and $y_i^s \in \{1, 2, \cdots, K^s\}$ is the disease degree label, with K^s being the number of severity levels.

In MTDL, there are three basic tasks denoted as $\mathcal{T}_c$ for disease category recognition, $\mathcal{T}_s$ for severity estimation, and $\mathcal{T}_h$ for the hybrid task that jointly performs $\mathcal{T}_c$ and $\mathcal{T}_s$. As shown in Figure 2, each task uses a standard ResNet50 (He et al., 2016) as the backbone for feature extraction. In particular, the two single tasks $\mathcal{T}_c$ and $\mathcal{T}_s$, each uses a multi-layer perceptron (MLP) to output the logits of its corresponding task, denoted as $\mathbf{z}_i^c \in \mathbb{R}^{K_c}$ and $\mathbf{z}_i^s \in \mathbb{R}^{K_s}$, respectively. For $\mathcal{T}_h$, two separate MLPs are used to perform two tasks simultaneously on a shared backbone, and the output is denoted as $\mathbf{z}_i^h = [\mathbf{z}_i^{hc} : \mathbf{z}_i^{hs}] \in \mathbb{R}^{K_c+K_s}$, where $\mathbf{z}_i^{hc}$ and $\mathbf{z}_i^{hs}$ corresponding to the logits for the disease category and severity, respectively. Usually, a softmax function is applied to the output of each task to produce the predicted probabilities, $\mathbf{p}_i^c \in \mathbb{R}^{K^c}$, $\mathbf{p}_i^s \in \mathbb{R}^{K^s}$ and $\mathbf{p}_i^h = [\mathbf{p}_i^{hc} : p_i^{hs}] \in \mathbb{R}^{K_c+K_s}$, respectively. Guided by these three basic tasks, MTDL employs a designed knowledge routing mechanism to build a tomato disease diagnosis model. The process begins with the distillation of multi-task knowledge from $\mathcal{T}_h$ back to the corresponding task models $\mathcal{T}_c$ and $\mathcal{T}_s$ (as shown in Figure 2A). These two models then engage in mutual learning (as shown in Figure 2B). Finally, the knowledge from these two models is integrated to output an enhanced multi-task model, namely $\mathcal{T}_h'$ (as shown in Figure 2C). The detailed learning process is described in the following sections, including, knowledge decomposition (Section 2.3.2), mutual knowledge tranfer (Section 2.3.3), and knowledge integration (Section 2.3.4).

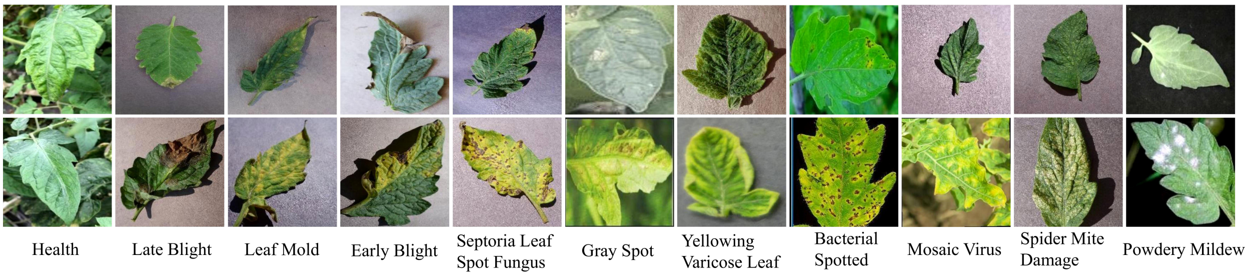


FIGURE 1
Examples of tomato diseases from the datasets.

TABLE 1 Summary of main datasets used in the study.

Dataset	AIChallenger2018			PlantDoc			Taiwan			Total
Class	Healthy	Moderate	Severe	Healthy	Moderate	Severe	Healthy	Moderate	Severe	
Health	1381			120			106			1607
Late Blight		302	1267		10	29		16	82	1706
Leaf Mold		371	384		40	67		22	45	929
Early Blight		287	505		22	86				900
Septoria Leaf Spot Fungus		481	922		23	141				1567
Gray Spot								25	59	84
Yellowing Varicose Leaf		1616	2790		35	88				4529
Bacterial Spotted		47	27		15	56		29	81	255
Mosaic Virus		104	194		26	43				367
Spider Mite Damage		619	310							929
Powdery Mildew								47	110	157
Total	1381	3827	6399	120	171	510	106	139	377	13030

2.3.2 Knowledge disentanglement

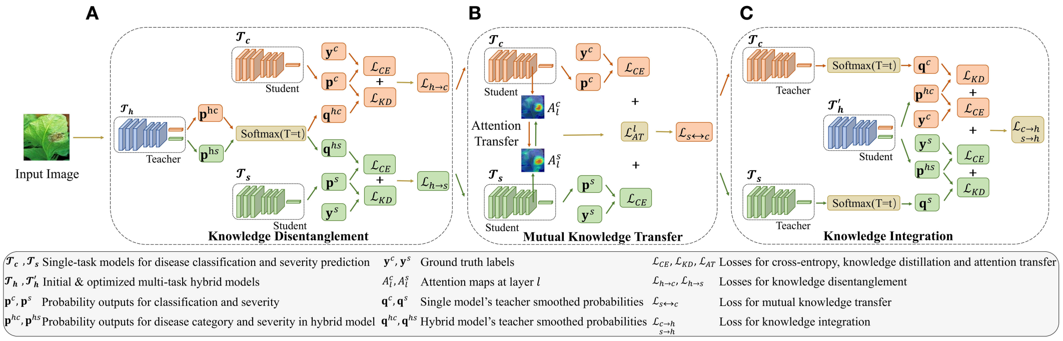
Multi-task learning has demonstrated its advantages in leveraging shared information among related tasks to improve performance on individual tasks. However, directly training a multi-tasking model can be suboptimal, as the tasks may have different levels of difficulty. For instance, the task of severity estimation is more challenging than the leaf disease classification task because it typically necessitates a finer analysis of the leaf and disease spot attributes (Wang et al., 2017). Therefore, given a multi-task model $\mathcal{T}_h$ pre-trained on dataset D, as shown in Figure 2A, it is reasonable to disentangle the shared knowledge and transfer it back to the single-task models, i.e., $\mathcal{T}_c$ and $\mathcal{T}_s$, using knowledge distillation (Hinton et al., 2015). Specifically, when distilling knowledge from $\mathcal{T}_h$ to $\mathcal{T}_c$, we first soften the probability $\mathbf{p}_i^{hc}$ by:

$$q_{i,j}^{hc} = \frac{\exp\left(p_{i,j}^{hc}/T\right)}{\sum_j \exp\left(p_{i,j}^{hc}/T\right)} \quad (1)$$

where T is the temperature hyperparameter that controls the sharpness of $\mathbf{q}_i^{hc}$, $p_{i,j}^{hc}$ is the j-th element of $\mathbf{p}_i^{hc}$, and $q_{i,j}^{hc}$ denotes the softened probability distribution of the j-th class for the i-th input data. The formulation of the knowledge distillation process from $\mathcal{T}_h$ to $\mathcal{T}_c$ involves minimizing the loss function $\mathcal{L}_{h\rightarrow c}$, which is defined as follows:

$$\mathcal{L}_{h\rightarrow c} = \frac{1}{N}\sum_{i=1}^{N}\left[\mathcal{L}_{CE}(\mathbf{p}_i^c, \mathbf{y}_i^c) + \mathcal{L}_{KD}(\mathbf{p}_i^c, \mathbf{q}_i^{hc})\right] \quad (2)$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, which measures the dissimilarity between the predicted probability distribution $\mathbf{p}_i^c$ and



$\mathcal{T}_c$,$\mathcal{T}_s$	Single-task models for disease classification and severity prediction	$\mathbf{y}^c, \mathbf{y}^s$	Ground truth labels	$\mathcal{L}_{CE}, \mathcal{L}_{KD}, \mathcal{L}_{AT}$	Losses for cross-entropy, knowledge distillation and attention transfer
$\mathcal{T}_h$,$\mathcal{T}'_h$	Initial & optimized multi-task hybrid models	A_l^c, A_l^s	Attention maps at layer l	$\mathcal{L}_{h\rightarrow c}, \mathcal{L}_{h\rightarrow s}$	Losses for knowledge disentanglement
$\mathbf{p}^c, \mathbf{p}^s$	Probability outputs for classification and severity	$\mathbf{q}^c, \mathbf{q}^s$	Single model's teacher smoothed probabilities	$\mathcal{L}_{s\leftrightarrow c}$	Loss for mutual knowledge transfer
$\mathbf{p}^{hc}, \mathbf{p}^{hs}$	Probability outputs for disease category and severity in hybrid model	$\mathbf{q}^{hc}, \mathbf{q}^{hs}$	Hybrid model's teacher smoothed probabilities	$\mathcal{L}_{\substack{c\rightarrow h \\ s\rightarrow h}}$	Loss for knowledge integration

FIGURE 2
Architecture of the multi-task distillation learning (MTDL). The MTDL framework uses a three-stage distillation process involving single-task models $\mathcal{T}_c$ and $\mathcal{T}_s$, and a multi-task model $\mathcal{T}_h$. Initially, knowledge from $\mathcal{T}_h$ is transferred to the single-task models. Then, $\mathcal{T}_c$ and $\mathcal{T}_s$share knowledge. Finally, their knowledge is integrated back into $\mathcal{T}_h$, creating an improved multi-task model $\mathcal{T}_h^{'}$. For simplicity, sample indices are omitted from the symbols in the figure. Additionally, the temperature parameter T in KD is fixed at t during the learning process. **(A)** Knowledge Disentanglement, **(B)** Mutual Knowledge Transfer, **(C)** Knowledge Integration.

the one-hot ground-truth label vector $\mathbf{y}_i^c$ for the single-task model $\mathcal{T}_c$. It can be written as shown in Equation 3:

$$\mathcal{L}_{CE}(\mathbf{p}_i^c, \mathbf{y}_i^c) = -\sum_{j=1}^{K_c} y_{i,j}^c \log p_{i,j}^c \tag{3}$$

And $\mathcal{L}_{KD}$, the knowledge distillation loss, which quantifies the divergence between q_i^{hc} and p_i^c, is defined as shown in Equation 4:

$$\mathcal{L}_{KD}(\mathbf{p}_i^c, \mathbf{q}_i^{hc}) = \sum_{j=1}^{K_c} q_{i,j}^{hc} \log \frac{q_{i,j}^{hc}}{p_{i,j}^c} \tag{4}$$

Similar to Equation 2, we can define a loss function from $\mathcal{T}_h$ to $\mathcal{T}_s$, denoted as $\mathcal{L}_{h \to s}$, which is given by:

$$\mathcal{L}_{h \to s} = \frac{1}{N} \sum_{i=1}^{N} \left[\mathcal{L}_{CE}(\mathbf{p}_i^s, \mathbf{y}_i^s) + \mathcal{L}_{KD}\left(\mathbf{p}_i^s, \mathbf{q}_i^{hs}\right) \right] \tag{5}$$

where $\mathbf{q}_i^{hs}$ is the probability distribution obtained by softening the severity prediction output $\mathbf{p}_i^{hs}$ from $\mathcal{T}_h$ (referred to in Equation 1), and $\mathbf{p}_i^s$ is the output from $\mathcal{T}_s$.

2.3.3 Mutual knowledge transfer

Upon completing the knowledge disentanglement process, the shared knowledge from the hybrid tasks $\mathcal{T}_h$ is individually transferred back to the corresponding subtasks, i.e., $\mathcal{T}_c$ for disease species classification and $\mathcal{T}_s$ for disease severity identification. We then employ mutual distillation to further investigate the complementarity of the two subtasks. Here, we assume that $\mathcal{T}_c$ and $\mathcal{T}_s$ use the same backbones, such as ResNet50. Motivated by Komodakis and Zagoruyko (2016), as shown in Figure 2B, the commonality of knowledge between subtasks is reflected in the consistency of attention maps from the middle layer. Specifically, given two feature mappings, F_l^c and F_l^s, which are the outputs of layer l of the models $\mathcal{T}_c$ and $\mathcal{T}_s$, respectively, we can calculate the attention maps, A_l^c and A_l^s, as shown in Equation 6:

$$A_l^c(x,y) = \frac{1}{C_i} \sum_{c=1}^{C_i} F_l^c(k,x,y), \quad A_l^s(x,y) = \frac{1}{C_i} \sum_{k=1}^{C_i} F_l^s(k,x,y) \tag{6}$$

where C_i is the number of channels in the feature mappings of F_l^c and F_l^s, and (k,x,y) specifies the location and channel of an activation value within the feature mapping. The attention maps A_l^c and A_l^s are computed by averaging the activation values across the channels of the respective feature mappings, F_l^c and F_l^s. For stability of optimization, we first reshape the A_l^c and A_l^s into a vector form as $\mathbf{a}_l^c = \mathbf{vec}(A_l^c)$ and $\mathbf{a}_l^s = \mathbf{vec}(A_l^s)$, where vec(.) is an operation that transforms a matrix into a vector by concatenating its columns. Then, we normalize the vectors using l_2 norm as shown in Equation 7:

$$\hat{\mathbf{a}}_l^c = \frac{\mathbf{a}_l^c}{\|\mathbf{a}_l^c\|_2}, \quad \hat{\mathbf{a}}_l^s = \frac{\mathbf{a}_l^s}{\|\mathbf{a}_l^s\|_2} \tag{7}$$

The attention transfer loss for layer l is written as shown in Equation 8:

$$\mathcal{L}_{AT}(\hat{\mathbf{a}}_l^c, \hat{\mathbf{a}}_l^s) = \|\hat{\mathbf{a}}_l^c - \hat{\mathbf{a}}_l^s\|_2^2 \tag{8}$$

And the total loss for mutual learning between subtasks is defined as follows:

$$\mathcal{L}_{s \leftrightarrow c} = \frac{1}{N} \sum_{i=1}^{N} \left[\mathcal{L}_{CE}(\mathbf{p}_i^c, \mathbf{y}_i^c) + \mathcal{L}_{CE}(\mathbf{p}_i^s, \mathbf{y}_i^s)] + \frac{1}{L} \sum_{l=1}^{L} \mathcal{L}_{AT}(\hat{\mathbf{a}}_l^c, \hat{\mathbf{a}}_l^s) \right] \tag{9}$$

where L denotes the number of layers considered for attention transfer loss.

2.3.4 Knowledge integration

The primary objective of the proposed MTDL is to enhance multi-task learning capabilities. In the final step of this learning framework, we consider the two sub-tasks after mutual learning, $\mathcal{T}_c$ and $\mathcal{T}_s$, and reintegrate them into the original multi-tasking model, denoted as … As shown in Figure 2C, this reintegration process results in an enhanced multi-task model $\mathcal{T}_h'$.The knowledge integration loss is formulated as follows:

$$\mathcal{L}_{\substack{c \to h \\ s \to h}} = \frac{1}{N} \sum_{i=1}^{N} \left[\mathcal{L}_{CE}(\mathbf{p}_i^{hc}, \mathbf{y}_i^c) + \mathcal{L}_{KD}(\mathbf{p}_i^{hc}, \mathbf{q}_i^c) + \mathcal{L}_{CE}(\mathbf{p}_i^{hs}, \mathbf{y}_i^s) + \mathcal{L}_{KD}(\mathbf{p}_i^{hs}, \mathbf{q}_i^s) \right] \tag{10}$$

where $\mathbf{q}_i^c$ and $\mathbf{q}_i^s$ represent the output of softened probability distributions of $\mathcal{T}_c$ and $\mathcal{T}_s$, respectively, which are obtained by applying the process described in Equation 1. The whole process of MTDL is summarized in Algorithm 1.

```
Require: Inputs: Single-task models $\mathcal{T}_c$, $\mathcal{T}_s$ and multi-task model $\mathcal{T}_h$.
Ensure: Outputs: Enhanced multi-task model $\mathcal{T}_h'$.
  1: Decompose $\mathcal{T}_h$into two sub-tasks $\mathcal{T}_c$ and $\mathcal{T}_s$using Equations 2 and Equation 5.
  2: Perform mutual learning between $\mathcal{T}_c$ and $\mathcal{T}_s$ using Equation 9.
  3: Reintegrate $\mathcal{T}_c$ and $\mathcal{T}_s$ into the original multi-task model $\mathcal{T}_h$ to produce the enhanced model us $\mathcal{T}_h'$ using Equation 10.
```

Algorithm 1. MTDL process.

2.4 Teacher-free based MTDL

In the staged learning process of MTDL, the current stage can be considered the teacher model for subsequent stages. While this approach fully utilizes the process of knowledge transfer, it also leads to a dependency on the teacher model, thereby reducing the flexibility of the framework. To overcome this limitation, inspired by the work of Yuan et al. (2020) and Zhao et al. (2022), we propose a decoupled teacher-free KD (DTF-KD) method. In the following sections, we first present the general form of the DTF-KD, and then demonstrate how it can be applied to MTDL.

In the absence of a teacher model, we introduce a virtual teacher. We define the output of this virtual teacher as a categorical distribution, $v_{i,j}$, given by:

$$v_{i,j} = \begin{cases} \alpha & \text{if} \quad j = t \\ (1-\alpha)/(K-1) & \text{if} \quad j \in \setminus t \end{cases} \tag{11}$$

where α is a predefined constant, typically ≥ 0.95, t is the correct class or target class for the i-th sample, K is the total number of classes, j represents the class index, and $\setminus t$ denotes all classes except the correct class t. This definition ensures that the virtual teacher assigns the highest probability to the correct class, while distributing the remaining probability equally among the incorrect classes.

In our proposed DTF-KD method, we divide the information distillation process into two parts: teacherfree based correct class KD (CC-KD) and teacher-free based non-correct class KD (NCC-KD). CC-KD focuses on the learning of target knowledge. It aims to transfer knowledge that is particularly important or challenging for the student model. In CC-KD, according to Equation 11, the binary probability outputs the virtual teacher for the correct class t and the K−1 non-correct classes are denoted as $\mathbf{q}_i^v = [q_{i,t}^v, q_{i,\setminus t}^v] \in \mathbb{R}^2$. These outputs are calculated using:

$$q_{i,t}^v = \frac{\exp\ (\alpha)}{\exp\ (\alpha) + \sum_{k=1,k\neq t}^{K} \exp\ (v_{i,k})}, \tag{12}$$

$$q_{i,\setminus t}^v = \frac{\sum_{k=1,k\neq t}^{K} \exp\ (v_{i,k})}{\exp\ (\alpha) + \sum_{k=1,k\neq t}^{K} \exp\ (v_{i,k})}$$

Correspondingly, for the student model, we can obtain $\mathbf{b}_i = [b_{i,t}, b_{i,}\setminus t] \in \mathbb{R}^2$, defined as:

$$b_{i,t} = \frac{\exp\ (z_{i,t})}{\sum_{j=1}^{K} \exp\ (z_{i,j})}, \qquad b_{i,\setminus t} = \frac{\sum_{k=1,k\neq t}^{K} \exp\ (z_{i,k})}{\sum_{j=1}^{K} \exp\ (z_{i,j})} \tag{13}$$

where $z_{i,j}$ represents the logit for the j-th class of i-th instance of the student model. Therefore, combining Equations 12 and 13, the loss function of CC-KD can be written as:

$$\mathcal{L}_{CC-KD}(\mathbf{b}_i, \mathbf{q}_i^v) = q_{i,t}^v \log \frac{q_{i,t}^v}{b_{i,t}} + q_{i,\setminus t}^v \log \frac{q_{i,\setminus t}^v}{b_{i,\setminus t}} \tag{14}$$

In NCC-KD, we consider the probability outputs for the K−1 non-correct classes, denoted as $\tilde{\mathbf{q}}_i^v \in \mathbb{R}^K{-}1$ for the virtual teacher and $\tilde{\mathbf{p}}_i \in \mathbb{R}^K{-}1$ for the student model. For each $m \in \{1, 2, \ldots, K\}\setminus\{t\}$, we calculate these outputs as follows:

$$\tilde{q}_{i,m}^v = \frac{\exp\ (v_{i,m})}{\sum_{k=1,k\neq t}^{K} \exp\ (v_{i,k})}, \qquad \tilde{p}_{i,m} = \frac{\exp\ (z_{i,m})}{\sum_{k=1,k\neq t}^{K} \exp\ (z_{i,k})} \tag{15}$$

where $v_{i,m}$ is defined in Equation 11, and $z_{i,m}$ represents the logit for the m-th class of the i-th instance from the student model. According to Equation 15, the NCC-KD loss function is then defined as:

$$\mathcal{L}_{NCC-KD}(\tilde{\mathbf{p}}_i, \tilde{\mathbf{q}}_i^v) = \sum_{j=1,j\neq t}^{K} \tilde{q}_{i,j}^v \log \frac{\tilde{q}_{i,j}^v}{\tilde{p}_{i,j}} \tag{16}$$

Combining Equations 14 and 16, the total loss of DTF-KD is

$$\mathcal{L}_{DFK-KD}(\mathbf{b}_i, \mathbf{q}_i^v, \tilde{\mathbf{p}}_i, \tilde{\mathbf{q}}_i^v) = \mathcal{L}_{CC-KD}(\mathbf{b}_i, \mathbf{q}_i^v) + \mathcal{L}_{NCC-KD}(\tilde{\mathbf{p}}_i, \tilde{\mathbf{q}}_i^v) \tag{17}$$

According to DTF-KD, we propose two variants of the MTDL framework. The first variant, as shown in Figure 3A which we call partially teacher-free MTDL (MTDL-PTF), eliminates the knowledge disentanglement stage from the MTDL process, thereby removing the dependency on the initial multi-task teacher model, known as $\mathcal{T}_h$. To compensate for the absence of $\mathcal{T}$ $_h$, we introduce two virtual teacher models corresponding to the two learning tasks of disease category recognition and severity estimation, denoted as $\mathcal{T}_c^v$ and $\mathcal{T}_s^v$, respectively. For $\mathcal{T}_c^v$, as described in Equations 12, 13 and 15, we obtain $\mathbf{q}_i^{vc} \in \mathbb{R}^2$ and $\mathbf{b}_i^c \in \mathbb{R}^2$ for the distillation outputs for the correct class, as well as and $\widetilde{\mathbf{q}}_i^{vc} \in \mathbb{R}^{K^c-1}$and $\widetilde{\mathbf{p}}_i^c \in \mathbb{R}^{K^c-1}$ the non-correct classes. Similarly, for $\mathcal{T}_s^v$, we can obtain $\mathbf{q}_{\mathrm{i}}^{\mathrm{vs}} \in \mathbb{R}^2$ and $\mathbf{b}_i^s \in \mathbb{R}^2$ for the correct severity level. For the non-correct severity levels, we can also obtain $\widetilde{\mathbf{q}}_i^{vs} \in \mathbb{R}^{K^s-1}$ and $\widetilde{\mathbf{p}}_i^{vs} \in \mathbb{R}^{K^s-1}$. Therefore, the mutual knowledge transfer process in MTDL-PTF is given as shown in Equation 18:

$$\mathcal{L}_{s\leftrightarrow c}^v = \mathcal{L}_{s\leftrightarrow c} + \frac{1}{N}\left[\sum_{i=1}^{N} \mathcal{L}_{DFK-KD}(\mathbf{b}_i^c, \mathbf{q}_i^{vc}, \tilde{\mathbf{p}}_i^c, \tilde{\mathbf{q}}_i^{vc}) + \sum_{i=1}^{N} \mathcal{L}_{DFK-KD}(\mathbf{b}_i^s, \mathbf{q}_i^{vs}, \tilde{\mathbf{p}}_i^s, \tilde{\mathbf{q}}_i^{vs})\right] \tag{18}$$

where $\mathcal{L}_{s\leftrightarrow c}$ and $\mathcal{L}_{DFK-KD}$ $\mathrm{L}_{DFK\text{-}KD}$ are defined in Equations 9 and 17, respectively.

In the second variant of MTDL, named teacher-free MTDL (MTDL-TF), we completely abandon the teacher model. The process of MTDL-TF is illustrated in Figure 3B. Instead, we directly introduce the distillation information from the virtual teacher models $\mathcal{T}_c^v$ and $\mathcal{T}_s^v$ into $\mathcal{T}_h$, which is defined as shown in Equation 19:

$$\begin{aligned} \mathcal{L}^v_{\substack{c\to h \\ s\to h}} = \frac{1}{N}\sum_{i=1}^{N} \Big[& \mathcal{L}_{CE}(\mathbf{p}_i^{hc}, \mathbf{y}_i^c) + \mathcal{L}_{DFK-KD}(\mathbf{b}_i^{hc}, \mathbf{q}_i^{vc}, \tilde{\mathbf{p}}_i^{hc}, \tilde{\mathbf{q}}_i^{vc}) \\ & + \mathcal{L}_{CE}(\mathbf{p}_i^{hs}, \mathbf{y}_i^s) + \mathcal{L}_{DFK-KD}(\mathbf{b}_i^{hs}, \mathbf{q}_i^{vs}, \tilde{\mathbf{p}}_i^{hs}, \tilde{\mathbf{q}}_i^{vs}) \Big] \end{aligned} \tag{19}$$

where b_i^{hc} and $\mathrm{b}_i^{hs}$are two binary probability outputs corresponding to the correct class and non-correct classes for the disease category recognition and severity estimation tasks, respectively, in the hybrid model $\mathcal{T}_h$. They can be obtained via $\mathbf{z}_i^{hc}$ and $\mathbf{z}_i^{hs}$ using Equation 13. Accordingly, the output for the non-correct classes in $\mathcal{T}_h$, $\widetilde{\mathbf{p}}_i^{hc}$ and $\widetilde{\mathbf{p}}_i^{hs}$, can be calculated by Equation 15.

3 Experimental results and discussion

3.1 Experimental setup

3.1.1 Model training

The MTDL framework consists of three main components: knowledge disentanglement, subtask mutual learning, and knowledge integration. To ensure simplicity and generality of the framework, we employ a consistent training strategy for different learning components. Specifically, the framework is trained using the SGD optimizer with a batch size of 32 and a momentum of 0.9. The initial learning rate is set to 0.001, and it is reduced by a factor

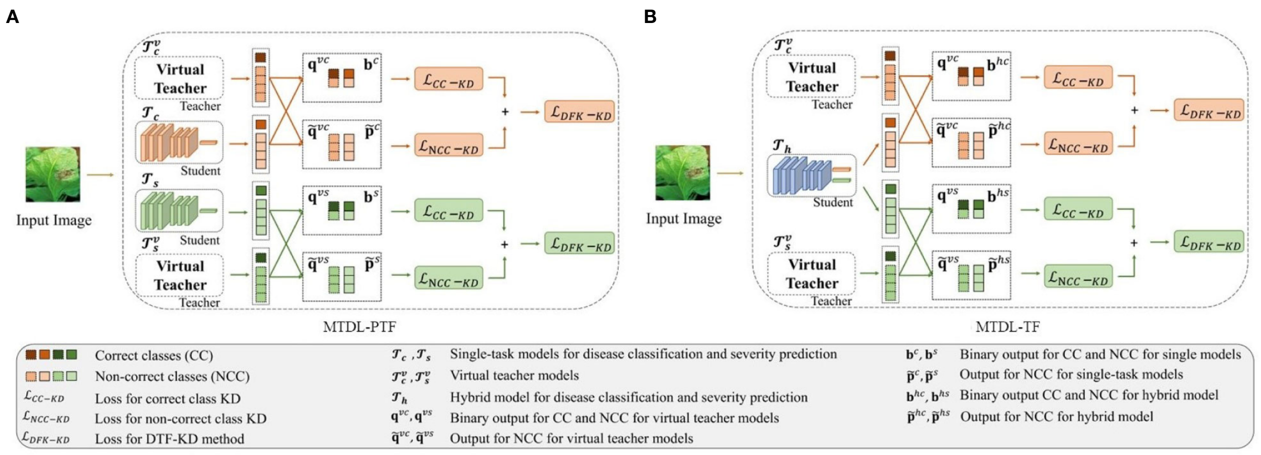

A
B
$\mathcal{T}_c^v$ Virtual Teacher Teacher
$\mathcal{T}_c$ Teacher
$\mathcal{T}_s$ Student
$\mathcal{T}_s^v$ Student
Virtual Teacher Teacher
Input Image
$\mathbf{q}^{vc}$ $\mathbf{b}^c$ $\tilde{\mathbf{q}}^{vc}$ $\tilde{\mathbf{p}}^c$
$\mathbf{q}^{vs}$ $\mathbf{b}^s$ $\tilde{\mathbf{q}}^{vs}$ $\tilde{\mathbf{p}}^s$
$\mathcal{L}_{CC-KD}$
$\mathcal{L}_{NCC-KD}$
$\mathcal{L}_{DFK-KD}$
+
MTDL-PTF
$\mathcal{T}_h$ Student
$\mathbf{b}^{hc}$ $\tilde{\mathbf{p}}^{hc}$ $\mathbf{b}^{hs}$ $\tilde{\mathbf{p}}^{hs}$
MTDL-TF
Correct classes (CC)
Non-correct classes (NCC)
$\mathcal{L}_{CC-KD}$ Loss for correct class KD
$\mathcal{L}_{NCC-KD}$ Loss for non-correct class KD
$\mathcal{L}_{DFK-KD}$ Loss for DTF-KD method
$\mathcal{T}_c$, $\mathcal{T}_s$ Single-task models for disease classification and severity prediction
$\mathcal{T}_c^v$, $\mathcal{T}_s^v$ Virtual teacher models
$\mathcal{T}_h$ Hybrid model for disease classification and severity prediction
$\mathbf{q}^{vc}, \mathbf{q}^{vs}$ Binary output for CC and NCC for virtual teacher models
$\tilde{\mathbf{q}}^{vc}, \tilde{\mathbf{q}}^{vs}$ Output for NCC for virtual teacher models
$\mathbf{b}^c, \mathbf{b}^s$ Binary output for CC and NCC for single models
$\tilde{\mathbf{p}}^c, \tilde{\mathbf{p}}^s$ Output for NCC for single-task models
$\mathbf{b}^{hc}, \mathbf{b}^{hs}$ Binary output CC and NCC for hybrid model
$\tilde{\mathbf{p}}^{hc}, \tilde{\mathbf{p}}^{hs}$ Output for NCC for hybrid model

FIGURE 3
Overview of the decoupled teacher-free (DTF) based MTDL. **(A)** Partially teacher-free MTDL (MTDL-PTF): Eliminating dependency on the multi-task teacher model in the knowledge disentanglement stage. **(B)** Teacher-Free MTDL (MTDL-TF): Simplifying MTDL to only retain the final knowledge integration stage with virtual teachers.

of 0.1 every 20 epochs. The weight decay is set to 1e-4. The maximum number of training epochs is set to 100, and an early stopping strategy is used based on the validation performance. If the validation loss does not improve for 5 consecutive epochs, the training process is stopped.

3.1.2 Hyperparameter settings

The MTDL framework involves three main stages of knowledge distillation, which correspond to the objective functions in Equations 2, 9, and 10. During the process, we use a temperature parameter T to smooth the output of the teacher model. This hyperparameter is determined through cross-validation using the validation set. A comprehensive analysis of hyperparameter selection can be found in Section 3.3.4.

3.1.3 Evaluation metrics

To evaluate the performance of the proposed MTDL method, we employ four commonly used evaluation metrics, namely Accuracy, Precision, Recall, and F1-score. Given true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), the specific definitions of these metrics are as shown in Equations 20 and 21:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}, \tag{20}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1} - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{21}$$

3.1.4 Baseline methods

The MTDL framework is a flexible knowledge distillation approach designed for tomato disease diagnosis. It aims to improve the performance of recognition models while reducing their parameter size and can be combined with various existing neural network architectures. To ensure the versatility of the MTDL framework, we incorporate four conventional network models, including ResNet101 (He et al., 2016), ResNet50 (He et al., 2016), DenseNet121 (Huang et al., 2017), and VGG16 (Simonyan and Zisserman, 2014), as well as four lightweight network models such as EfficientNet (Tan and Le, 2019), ShuffleNetV2 (Zhang et al., 2018b), MobileNetV3 (Howard et al., 2019), and SqueezeNet (Iandola et al., 2016). Detailed information about these models can be found in Table 2. These backbone models serve as the learning components in different stages of the MTDL framework. We use the original classification results of these models as a baseline and compare the results before and after the multi-task distillation process to validate the effectiveness of the proposed framework.

3.2 Results

3.2.1 Performance comparison

In this section, we report the results from two experimental settings. The first setting, referred to as unified MTDL, employs the same network architecture for teacher and student modules. This setting aims to verify the effectiveness of the multi-stage distillation architecture proposed in this paper. The second setting, termed heterogeneous MTDL, involves using lightweight network architectures for all student models within the MTDL framework. This setting is designed to demonstrate the advantages of the proposed architecture in achieving a balance between performance and efficiency. As a reference, Table 2 lists the baseline results of the initial two single tasks $\mathcal{T}_c$ and $\mathcal{T}_s$, as well as the multi-task model $\mathcal{T}_h$, where $\mathcal{T}_{hc}$ and $\mathcal{T}_{hs}$ correspond to the results of $\mathcal{T}_h$ for disease classification and severity estimation tasks, respectively. The results in Table 2 demonstrate that the multi-task learning approach effectively enhances performance across various network architectures.

The results for MTDL with a unified architecture are presented in Table 3. We can observe that all models show improvement when using MTDL for knowledge learning. This indicates that the MTDL

TABLE 2 Baseline results of single and multi-task models.

Methods	Single Task (Accuracy)		Multi Task (Accuracy)		Single Task (F1-score)		Multi Task (F1-score)		Parameter	FLOPs
	$\mathcal{T}_c$	$\mathcal{T}_s$	$\mathcal{T}_{hc}$	$\mathcal{T}_{hs}$	$\mathcal{T}_c$	$\mathcal{T}_s$	$\mathcal{T}_{hc}$	$\mathcal{T}_{hs}$	(M)	(G)
VGG16	96.68	93.34	96.76 (↑0.08)	93.43 (↑0.09)	96.57	94.34	96.82 (↑0.25)	94.53 (↑0.19)	253.864	15.699
ResNet101	98.11	93.61	98.56 (↑0.45)	94.33 (↑0.72)	97.72	94.51	98.14 (↑0.42)	95.13 (↑0.62)	42.529	7.832
ResNet50	97.21	93.43	97.75 (↑0.54)	93.70 (↑0.27)	97.20	94.43	97.41 (↑0.21)	94.69 (↑0.26)	23.537	4.109
DenseNet121	95.68	91.63	96.58 (↑0.90)	91.99 (↑0.36)	95.68	92.63	96.58 (↑0.90)	93.02 (↑0.39)	6.968	2.865
MobileNetV3Large	97.66	93.43	98.20 (↑0.54)	93.52 (↑0.09)	96.46	94.43	97.18 (↑0.72)	94.52 (↑0.09)	5.450	0.225
EfficientNet	97.75	93.88	98.11 (↑0.36)	93.97 (↑0.09)	96.65	94.78	97.11 (↑0.46)	94.97 (↑0.19)	4.025	0.398
MobileNetV3Small	97.03	91.72	97.21 (↑0.18)	92.35 (↑0.63)	96.01	92.62	96.21 (↑0.20)	93.34 (↑0.72)	2.123	0.059
ShuffleNetV2	96.58	91.63	96.76 (↑0.18)	91.99 (↑0.36)	95.37	92.62	95.76 (↑0.39)	92.79 (↑0.17)	1.268	0.148
SqueezeNet	94.15	90.37	94.33 (↑0.18)	90.45 (↑0.08)	94.35	91.37	94.53 (↑0.18)	91.75 (↑0.38)	0.743	0.738

$\mathcal{T}_c$ and $\mathcal{T}_s$ represent the disease category recognition and severity estimation tasks in single-task models, respectively. $\mathcal{T}_{hc}$ and $\mathcal{T}_{hs}$ represent the corresponding tasks in multi-task models. The symbol ↑ symbol indicates Accuracy or F1-score improvement from the single-task baseline.

framework effectively leverages the staged learning of knowledge and the complementarity between different tasks. In terms of specific models, ResNet101 achieves the highest performance in both tasks under the MTDL setting, with Accuracy scores of 98.92% for $\mathcal{T}_c$ and 95.32% for $\mathcal{T}_s$, respectively. The corresponding F1-scores are 98.78% and 96.32%, respectively. These results can be attributed to both the ResNet101's powerful feature extraction capabilities and MTDL's effective multi-task learning strategy. On the other hand, SqueezeNet shows significant improvement with an increase of 1.08% and 2.53% in Accuracy of $\mathcal{T}_c$ and $\mathcal{T}_s$ respectively, and an increase of 0.68% and 2.26% in F1-scoref or each task. This suggests that the MTDL allows the lightweight model to learn more robust and comprehensive features. Furthermore, Table 3 also provides a comparison between the MTDL, MTDL-PTF, and MTDL-TF methods across various architectures. The results indicate that while the overall performance of MTDL-PTF and MTDL-TF decreases when the dependence on the teacher model is reduced, the introduction of a virtual teacher model significantly improves the accuracy of both methods compared to the original multitask learning. This indeed validates the effectiveness of the decoupled teacher-free knowledge distillation approach that we proposed. We also display the confusion matrices for results using ResNet50 as the backbone. As shown in Figure 4, it is evident that our proposed MTDL method either maintains or improves performance across all individual classes for both disease classification and severity estimation tasks. This demonstrates MTDL's ability to achieve a balanced enhancement in both overall performance and category-specific outcomes.

Furthermore, to investigate the impact of using teacher and student models with different architectures on the performance of the MTDL framework, we employ complex models like DenseNet121 for the teacher and lightweight models such as EfficientNet for the student. The results presented in Table 4 substantiate the effectiveness of this heterogeneous MTDL approach. For instance, when using ResNet101 as the teacher model, the SqueezeNet student model shows an improvement of 1.95% and 3.07% in $\mathcal{T}_c$ and $\mathcal{T}_s$ respectively, which are higher than the result obtained under the unified architecture MTDL setting. These results suggest that a more powerful teacher model enriches the student model's learning.

Finally, to ensure the effectiveness of our proposed method, we conduct a comprehensive comparison with four well-established approaches in the field to validate its performance:

(a) Dual-stream hierarchical bilinear pooling (DHBP) (Wang et al., 2022): As a multi-task method initially developed for crops and diseases classification, we adapt DHBP for both disease classification and severity prediction tasks. This comparison allows us to evaluate the performance of our MTDL approach against a specialized multi-task learning method within the same domain.

(b) Traditional knowledge distillation (KD) (Ghofrani and Toroghi, 2022) and decouple knowledge distillation (DKD) (Zhao et al., 2022): These two methods represent the knowledge distillation category. We apply KD and its enhanced version, DKD, to our disease recognition and severity estimation tasks, providing a direct comparison with standard and advanced distillation techniques.

(c) Attention transfer (AT) (Komodakis and Zagoruyko, 2016): Differing from KD and DKD that focus on distilling knowledge through predicted outcomes, AT utilizes attention maps to transfer knowledge between the teacher and student models. Including AT in our comparison allows us to assess the efficacy of a distinct transfer learning approach.

To ensure fair comparisons among KD, DKD, AT, and MTDL, we consistently used ResNet-101 as the teacher and MobileNetV3Small as the student model. This approach enables a reliable assessment of knowledge distillation efficacy. Additionally, we present MTDL results using ResNet-101 as both teacher and student, aligning with DHBP's backbone, to effectively demonstrate its multi-tasking capabilities.

TABLE 3 Performance of MTDL and its variants in a unified architecture.

Methods (Accuracy)	MTDL		MTDL-PTF		MTDL-TF	
	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$	$\mathcal{T}^v_{hc}$	$\mathcal{T}^v_{hs}$	$\mathcal{T}^v_{hc}$	$\mathcal{T}^v_{hs}$
VGG16	97.75 (↑0.99)	94.15 (↑0.72)	97.48 (↑0.72)	94.24 (↑0.81)	97.12 (↑0.36)	93.70 (↑0.27)
ResNet101	98.92 (↑0.36)	95.32 (↑0.99)	98.65 (↑0.09)	94.87 (↑0.54)	98.65 (↑0.09)	94.78 (↑0.45)
ResNet50	98.20 (↑0.45)	94.87 (↑1.17)	98.11 (↑0.36)	94.60 (↑0.90)	97.93 (↑0.18)	94.34 (↑0.64)
DenseNet121	97.30 (↑0.72)	93.79 (↑1.80)	97.30 (↑0.72)	93.79 (↑1.80)	97.30 (↑0.72)	92.35 (↑0.36)
Average Improvement	↑0.63	↑1.17	↑0.47	↑1.01	↑0.34	↑0.43
MobileNetV3Large	98.74 (↑0.54)	94.60 (↑1.08)	98.65 (↑0.45)	94.24 (↑0.72)	98.56 (↑0.36)	93.97 (↑0.45)
EfficientNet	98.74 (↑0.63)	94.78 (↑0.81)	98.47 (↑0.36)	94.33 (↑0.36)	98.56 (↑0.45)	94.24 (↑0.27)
MobileNetV3Small	97.48 (↑0.27)	93.16 (↑0.81)	97.84 (↑0.63)	93.16 (↑0.81)	97.30 (↑0.09)	92.53 (↑0.18)
ShuffleNetV2	97.21 (↑0.45)	93.52 (↑1.53)	97.21 (↑0.45)	93.70 (↑1.71)	96.94 (↑0.18)	93.07 (↑1.08)
SqueezeNet	95.41 (↑1.08)	92.98 (↑2.53)	96.40 (↑2.07)	93.07 (↑2.62)	95.14 (↑0.81)	91.63 (↑1.18)
Average Improvement	↑0.59	↑1.35	↑0.79	↑1.24	↑0.38	↑0.63
Methods (F1-Score)	MTDL		MTDL-PTF		MTDL-TF	
	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$	$\mathcal{T}^v_{hc}$	$\mathcal{T}^v_{hs}$	$\mathcal{T}^v_{hc}$	$\mathcal{T}^v_{hs}$
VGG16	97.85 (↑1.03)	95.15 (↑0.62)	97.47 (↑0.65)	95.24 (↑0.41)	96.96 (↑0.14)	94.77 (↑0.24)
ResNet101	98.78 (↑0.64)	96.32 (↑1.19)	98.46 (↑0.32)	95.86 (↑0.56)	98.49 (↑0.35)	95.68 (↑0.38)
ResNet50	97.52 (↑0.32)	95.87 (↑1.44)	98.11 (↑0.70)	95.58 (↑0.89)	97.59 (↑0.18)	95.24 (↑0.55)
DenseNet121	97.11 (↑0.53)	94.80 (↑1.78)	97.11 (↑0.53)	94.60 (↑1.58)	97.03 (↑0.45)	93.34 (↑0.32)
Average Improvement	↑0.63	↑1.26	↑0.55	↑0.86	↑0.28	↑0.37
MobileNetV3Large	97.65 (↑0.47)	95.60 (↑1.08)	97.41 (↑0.23)	95.24 (↑0.72)	97.25 (↑0.07)	94.56 (↑0.04)
EfficientNet	97.95 (↑0.84)	95.78 (↑0.81)	97.52 (↑0.41)	95.33 (↑0.36)	97.36 (↑0.25)	95.24 (↑0.27)
MobileNetV3Small	97.41 (↑1.20)	94.16 (↑0.82)	97.28 (↑1.07)	94.16 (↑0.82)	97.14 (↑0.93)	93.36(↑0.02)
ShuffleNetV2	97.01 (↑1.25)	94.52 (↑1.73)	97.01 (↑1.25)	94.60 (↑1.81)	96.74 (↑0.98)	94.27 (↑1.45)
SqueezeNet	95.21 (↑0.68)	94.01 (↑2.26)	96.52 (↑1.99)	94.27 (↑2.52)	94.97 (↑0.81)	92.63 (↑0.88)
Average Improvement	↑0.89	↑1.34	↑0.99	↑0.76	↑0.61	↑0.53

$\mathcal{T}'_{hc}$ and $\mathcal{T}'_{hs}$ represent MTDL's performance, while $\mathcal{T}^v_{hc}$ and $\mathcal{T}^v_{hs}$ are for MTDL-PTF and MTDL-TF with a virtual teacher. The ↑ symbol indicates Accuracy and F1-score improvement, referencing the multi-task baseline from Table 2.

The results are shown in Table 5. In our experiments, MTDL with ResNet-101 as both teacher and student models achieve the best results, outperforming DHBP in disease classification by 0.53% in Accuracy and 0.29% in F1-score, and in severity prediction by 0.86% in Accuracy and 1.08% in F1-score. These improvements validate MTDL's phased multi-task learning approach. Moreover, when compared under the same teacher-student model setup with other distillation methods (KD, DKD, AT), MTDL excelled, particularly surpassing DKD by 0.37% in Accuracy and 0.16% in F1-score for disease classification, and by 0.62% in Accuracy and 0.38% in F1-score for severity prediction. This indicates the effectiveness of MTDL's proposed mutual distillation learning between teachers and students.

3.2.2 Significance analysis

In this subsection, we conduct a Wilcoxon Signed-Rank Test (Corder and Foreman, 2014) to evaluate the significance of the performance improvements across all CNN architectures. We provide the detailed significance analysis corresponding to the results originally presented in Tables 3 and 4 in the following Table 6 and 7. In Table 6, we present a comparison of the performance of our MTDL model and its variants against several baseline CNN architectures. This table focuses on scenarios within our MTDL framework where both the teacher and student models utilize identical architecture. The results from this table demonstrate statistically significant improvements across all comparisons in both disease classification and severity prediction tasks. The p-values obtained are consistently well below the 0.05 threshold, indicating robust enhancements attributed to our MTDL approach. Similarly, Table 7 showcases the results in a heterogeneous setting, where the MTDL model employs a more complex architecture as the teacher model and a lightweight network as the student model. In these comparisons, the results again confirm significant improvements across all evaluated aspects.

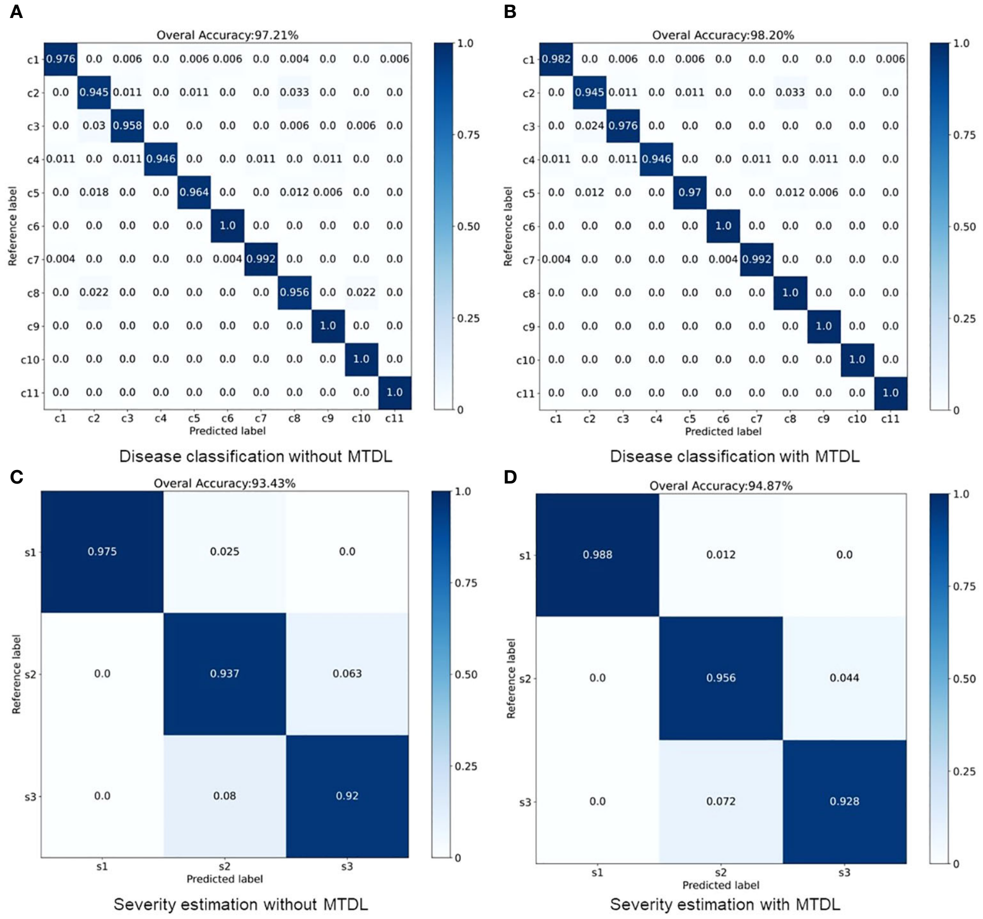


FIGURE 4
Performance improvement through multi-stage distillation in MTDL. **(A)** Disease classification without MTDL, **(B)** Disease classification with MTDL, **(C)** Severity estimation without MTDL, **(D)** Severity estimation with MTDL.

In addition, we also perform the significance of the results in comparison with other multi-task and distillation learning methods. with the results recorded in Table 8. It can be seen that in most cases, the MTDL framework shows statistically significant differences when compared with methods like DHBP, KD, DKD, and AT, with p-values well beneath the 0.05 significance threshold. However, there is one exception to note: in the case of MTDL (ResNet101-MobileNetV3Small) vs DHBP for severity prediction, the p-value is slightly above the conventional threshold for significance. This exception likely stems from MTDL employing lightweight MobileNetV3Small as the distillation target, whereas DHBP uses the more substantial ResNet101 as its base model.

3.3 Discussion

3.3.1 The effectiveness of multi-stage distillation learning

We assess the effectiveness of the three stages in our MTDL framework: knowledge disentanglement, mutual knowledge transfer, and knowledge integration. To do so, we employ single-task and multi-task models as our baselines and incorporate the results obtained after each stage of learning. As illustrated in Figure 5, the results in terms of Accuracy and F1-score align with our expectations. The results clearly demonstrate that each stage of learning contributes to the final performance improvement, thereby validating the effectiveness of staged distillation in the MTDL framework.

3.3.2 Trade-off between performance and efficiency

We investigate the balance between performance and efficiency within the context of our MTDL framework. Performance is measured by Accuracy, while efficiency is represented by the number of parameters and floating-point operations (FLOPs). We use the single-task ResNet101 model and the multi-task ResNet101 model as baselines due to their superior performance across all single-task and multi-task models, as shown in Table 3. The results are presented in Figure 6, and the size of each model's marker in the figure represents the number of parameters used by the model.

It can be observed that there is a similar trend in both task of disease classification (Figure 6A) and disease severity estimation (Figure 6B). Our MTDL-enhanced ResNet101 notably surpasses the single-task baseline with an Accuracy improvement of 0.81% for disease classification and 1.71% for severity estimation, and it

TABLE 4 Performance evaluation of MTDL under a heterogeneous setting.

Methods (Accuracy)		MTDL		Methods (Accuracy)		MTDL	
Teacher	Student	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$	Teacher	Student	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$
VGG16	MobileNetV3Large	98.74 (↑0.54)	94.51 (↑0.99)	ResNet50	MobileNetV3Large	98.92 (↑0.72)	94.42 (↑0.90)
	EfficientNet	98.47 (↑0.36)	94.54 (↑0.57)		EfficientNet	98.74 (↑0.63)	94.51 (↑0.54)
	MobileNetV3Small	97.48 (↑0.27)	93.52 (↑1.17)		MobileNetV3Small	97.66 (↑0.45)	94.15 (↑1.80)
	ShuffleNetV2	97.57 (↑0.81)	93.07 (↑1.08)		ShuffleNetV2	97.66 (↑0.90)	93.07 (↑1.08)
	SqueezeNet	95.95 (↑1.62)	92.62 (↑2.17)		SqueezeNet	96.04 (↑1.71)	92.98 (↑2.53)
Average Improvement		↑0.72	↑1.20	Average Improvement		↑0.88	↑1.37
ResNet101	MobileNetV3Large	98.92 (↑0.72)	95.05 (↑1.53)	DenseNet121	MobileNetV3Large	98.38 (↑0.18)	94.51 (↑0.99)
	EfficientNet	98.79 (↑0.68)	95.13 (↑1.16)		EfficientNet	98.47 (↑0.36)	94.87 (↑0.90)
	MobileNetV3Small	97.93 (↑0.72)	94.24 (↑1.89)		MobileNetV3Small	97.87 (↑0.66)	93.34 (↑0.99)
	ShuffleNetV2	98.02 (↑1.26)	93.97 (↑1.98)		ShuffleNetV2	97.48 (↑0.72)	93.79 (↑1.80)
	SqueezeNet	96.28 (↑1.95)	93.52 (↑3.07)		SqueezeNet	96.17 (↑1.84)	92.80 (↑2.35)
Average Improvement		↑1.07	↑1.93	Average Improvement		↑0.75	↑1.41
Methods (F1-Score)		MTDL		Methods (F1-score)		MTDL	
Teacher	Student	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$	Teacher	Student	$\mathcal{T}'_{hc}$	$\mathcal{T}'_{hs}$
VGG16	MobileNetV3Large	98.54 (↑1.36)	95.24 (↑0.72)	ResNet50	MobileNetV3Large	98.72 (↑1.54)	95.62 (↑1.10)
	EfficientNet	97.98 (↑0.80)	95.36 (↑0.39)		EfficientNet	98.46 (↑1.35)	95.51 (↑0.54)
	MobileNetV3Small	97.46 (↑1.25)	94.52 (↑1.18)		MobileNetV3Small	97.66 (↑1.45)	94.10 (↑0.76)
	ShuffleNetV2	97.27 (↑1.51)	94.29 (↑1.50)		ShuffleNetV2	97.66 (↑1.45)	93.98 (↑1.19)
	SqueezeNet	95.76 (↑1.23)	93.42 (↑1.67)		SqueezeNet	96.04 (↑1.51)	93.67 (↑1.92)
Average Improvement		↑0.83	↑1.09	Average Improvement		↑1.46	↑1.10
ResNet101	MobileNetV3Large	98.62 (↑1.44)	95.85 (↑1.33)	DenseNet121	MobileNetV3Large	98.38 (↑1.20)	94.97 (↑0.45)
	EfficientNet	98.54 (↑1.43)	96.03 (↑1.06)		EfficientNet	98.27 (↑1.16)	95.62 (↑0.65)
	MobileNetV3Small	97.72 (↑1.51)	94.94 (↑1.60)		MobileNetV3Small	97.87 (↑1.66)	94.34 (↑1.00)
	ShuffleNetV2	98.22 (↑2.46)	94.87 (↑2.08)		ShuffleNetV2	97.28 (↑1.52)	94.09 (↑1.30)
	SqueezeNet	96.28 (↑1.75)	93.52 (↑1.77)		SqueezeNet	96.17 (↑1.64)	93.70 (↑1.95)
Average Improvement		↑1.72	↑1.57	Average Improvement		↑1.44	↑1.07

The ↑ symbol indicates an improvement in Accuracy and F1-score, as compared to the results listed in Table 2, where both teacher and student models use a unified lightweight network for multi-task learning.

outperforms the multi-task baseline with 0.36% and 0.99% improvements respectively. When using MobileNetV3Large as the MTDL-optimized model, we achieved significant performance gains with reduced parameter count and FLOPs, while still enhancing Accuracy over both baselines. For example, the MobileNetV3Large model, enhanced by our MTDL framework, outperforms the ResNet101 baseline by 0.63% and 1.44% in the two tasks, respectively. Remarkably, this is achieved with only 12.81% of the parameters (5.450M vs. 42.529M) and 2.87% of the FLOPs (0.225G vs. 7.832G). These findings highlight the MTDL framework's capability to improve performance significantly while maintaining computational efficiency, thereby reinforcing its advantage over conventional models.

Therefore, we need to select the appropriate distillation model for each specific scenario. The choice depends on balancing computational resources and performance. Typically, complex teachers like ResNet101 outperform compact students such as MobileNet, owing to deeper architectures. MTDL promotes mutual learning between teachers and students, simultaneously enhancing both models. With abundant resources, an MTDL-optimized teacher offers substantial performance gains. In contrast, for limited-resource scenarios like mobile inference, MTDL can distill a lightweight yet performant student model. Additionally, the teacher-free MTDL-TF variant reduces dependency on complex teachers, offering an alternative when resources are constrained.

TABLE 5 Comparative performance analysis of MTDL with other distillation-based and multi-task learning methods for disease classification and severity prediction.

Methods	Teacher	Student	Disease	Classification	Severity	Prediction
			Accuracy	F1-score	Accuracy	F1-score
DHBP (Wang et al., 2022)	ResNet101		98.39	98.49	94.46	95.24
KD (Ghofrani and Toroghi, 2022)	ResNet101	MobileNetV3Small	97.30	97.28	93.16	93.96
DKD Zhao et al. (2022)	ResNet101	MobileNetV3Small	97.56	97.56	93.62	94.56
AT Komodakis and Zagoruyko (2016)	ResNet101	MobileNetV3Small	97.39	97.46	93.28	94.09
MTDL	ResNet101 ResNet101	MobileNetV3Small ResNet101	97.93 98.92	97.72 98.78	94.24 95.32	94.94 96.32

3.3.3 Visual analysis for multi-task learning

In this section, we use Grad-CAM (Selvaraju et al., 2017) for visual analysis to gain deeper insights into the learning process of our MTDL framework. We examine three severity levels of Early Blight: healthy, general, and severe. Visualizations for single-task and multi-task models, as well as for each stage of MTDL learning, are provided. Figure 7 shows that the model's attention shifts toward task-relevant areas as it learns. For healthy leaves, the MTDL-enhanced model more precisely identifies the leaf as a whole, aligning with human visual systems. For leaves at a general severity level, the model focuses on localized disease spots for classification but expands its attention to surrounding regions for severity estimation. In cases of severe disease levels, the disease spots typically exhibit a widespread distribution across the leaf area. The knowledge integration model, in its pursuit to accurately recognize both the disease type and severity, tends to produce a Grad-CAM sensitivity map covering the entire leaf area. This comprehensive coverage contrasts with the single-task model, which primarily focuses on localized diseased regions, and the multi-task model, which, although it expands the area of interest, does not distribute sensitivity intensity as effectively. Moreover, the distribution of sensitivity intensity in the knowledge integration model offers a more realistic representation of the disease's extensive impact, thereby enhancing the model's explanatory power for Severe Early Blight. This analysis highlights the MTDL framework's adaptability in shifting its focus based on the task and severity, thereby improving performance and interpretability.

3.3.4 Parameter sensitivity analysis

The temperature parameter T adjusts the softmax output in the neural network, smoothing the probability distribution and revealing more nuanced information about the model's predictions. This is crucial for knowledge distillation, where it aids in transferring detailed information from a teacher to a student model. This concept is introduced and utilized in Equation 1. To assess the sensitivity of our model to T, we vary T within the interval [0.1,50] and record the Accuracy of the disease classification and severity estimation tasks for each value. The results of nine common network architectures are shown in Figure 8. Despite the differences in architecture, a similar trend is observed: as T increases, the model's performance improves, but rapidly declines when T exceeds 10. Notably, the model's

TABLE 6 Wilcoxon Signed-Rank Test results for MTDL variants' Accuracy in a unified architecture.

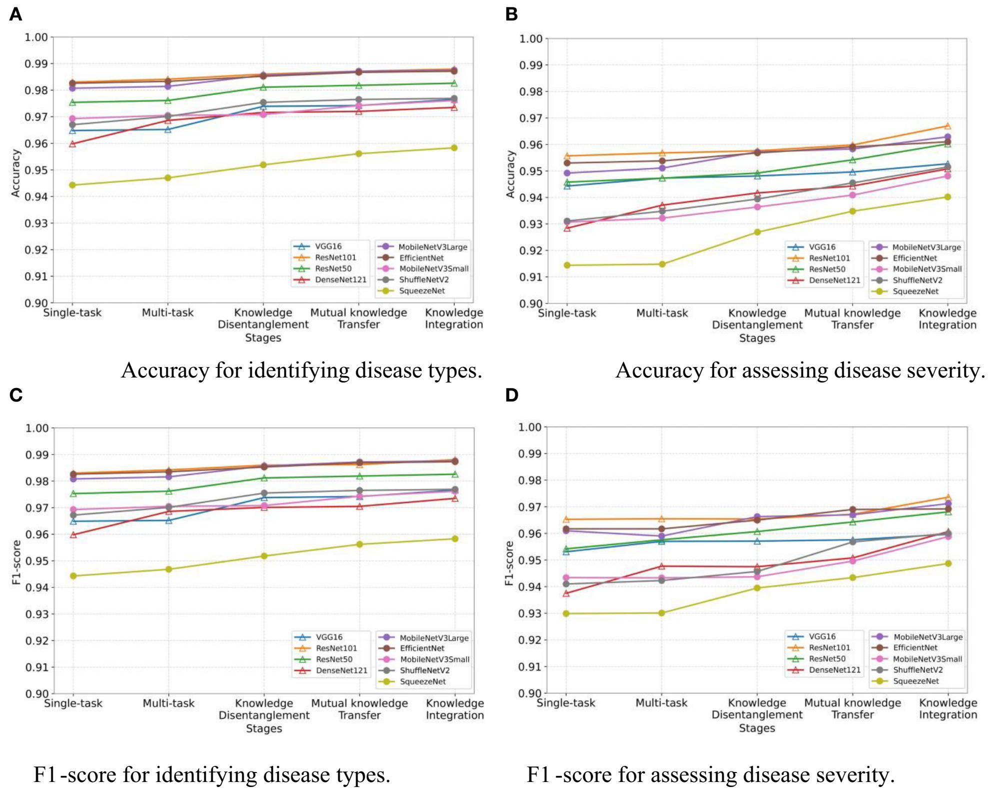
Task	Model	vs VGG16	vs ResNet101	vs ResNet50	vs DenseNet121	
Disease Classification	MTDL	1.953 × 10–^{3}	1.367×10^{-2}	1.953×10^{-3}	1.172×10^{-2}	
	MTDL-PTF	1.953×10^{-3}	1.065×10^{-2}	1.953×10^{-3}	1.953×10^{-3}	
	MTDL-TF	1.953×10^{-3}	2.066 × 10–^{2}	4.980×10^{-2}	1.953×10^{-3}	
Severity Prediction	MTDL	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	
	MTDL-PTF	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	
	MTDL-TF	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	
Task	Model	vs MobileNetV3Large	vs EfficientNet	vs MobileNetV3Small	vs ShuffleNetV2	vs SqueezeNet
Disease Classification	MTDL	1.151×10^{-2}	1.953×10^{-3}	3.906×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL-PTF	1.172×10^{-1}	1.079×10^{-2}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL-TF	4.206×10^{-2}	1.065×10^{-2}	3.906×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
Severity Prediction	MTDL	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL-PTF	1.953×10^{-3}	3.906×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL-TF	1.278×10^{-2}	2.734 × 10–2	1.079×10^{-2}	1.953×10^{-3}	1.953×10^{-3}

TABLE 7 Wilcoxon Signed-Rank Test results for MTDL variants' Accuracy under heterogeneous settings ('()' indicate teacher models).

Task	Model	vs MobileNetV3Large	vs EfficientNet	vs MobileNetV3Small	vs ShuffleNetV2	vs SqueezeNet
Disease Classification	MTDL (VGG16)	7.632×10^{-3}	1.162×10^{-2}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (ResNet101)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (ResNet50)	1.953×10^{-3}	1.953×10^{-3}	3.906×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (DenseNet121)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
Severity Prediction	MTDL (VGG16)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (ResNet101)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (ResNet50)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (DenseNet121)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}

TABLE 8 Results of the Wilcoxon Signed-Rank Test for MTDL and its variants versus other methods (The first in '()' is the teacher model and the second is the student model).

Task	Model	vs DHBP	vs KD	vs DKD	vs AT
Disease Classification	MTDL (ResNet101-ResNet101)	1.507×10^{-2}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
	MTDL (ResNet101-MobileNetV3Small)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}
Severity Prediction	MTDL (ResNet101-ResNet101)	1.953×10^{-3}	1.953×10^{-3}	1.953×10^{-3}	1.953×10–3
	MTDL (ResNet101-MobileNetV3Small)	9.219×10–2	1.953×10^{-3}	1.953×10–3	1.953×10–3



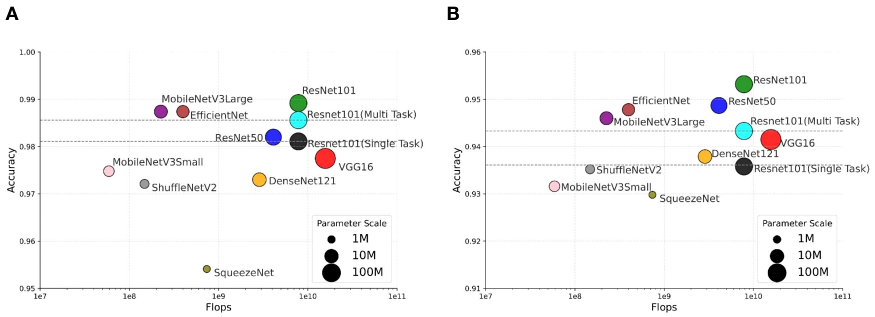
Accuracy for identifying disease types.

Accuracy for assessing disease severity.

F1-score for identifying disease types.

F1-score for assessing disease severity.

FIGURE 5
Performance improvement through multi-stage distillation in MTDL. **(A)** Accuracy for identifying disease types, **(B)** Accuracy for assessing disease severity, **(C)** F1-score for identifying disease types, **(D)** F1-score for assessing disease severity.



Task for disease classification.

Task for disease estimation.

FIGURE 6
Trade-off between performance and efficiency. **(A)** Task for disease classification, **(B)** Task for disease estimation.

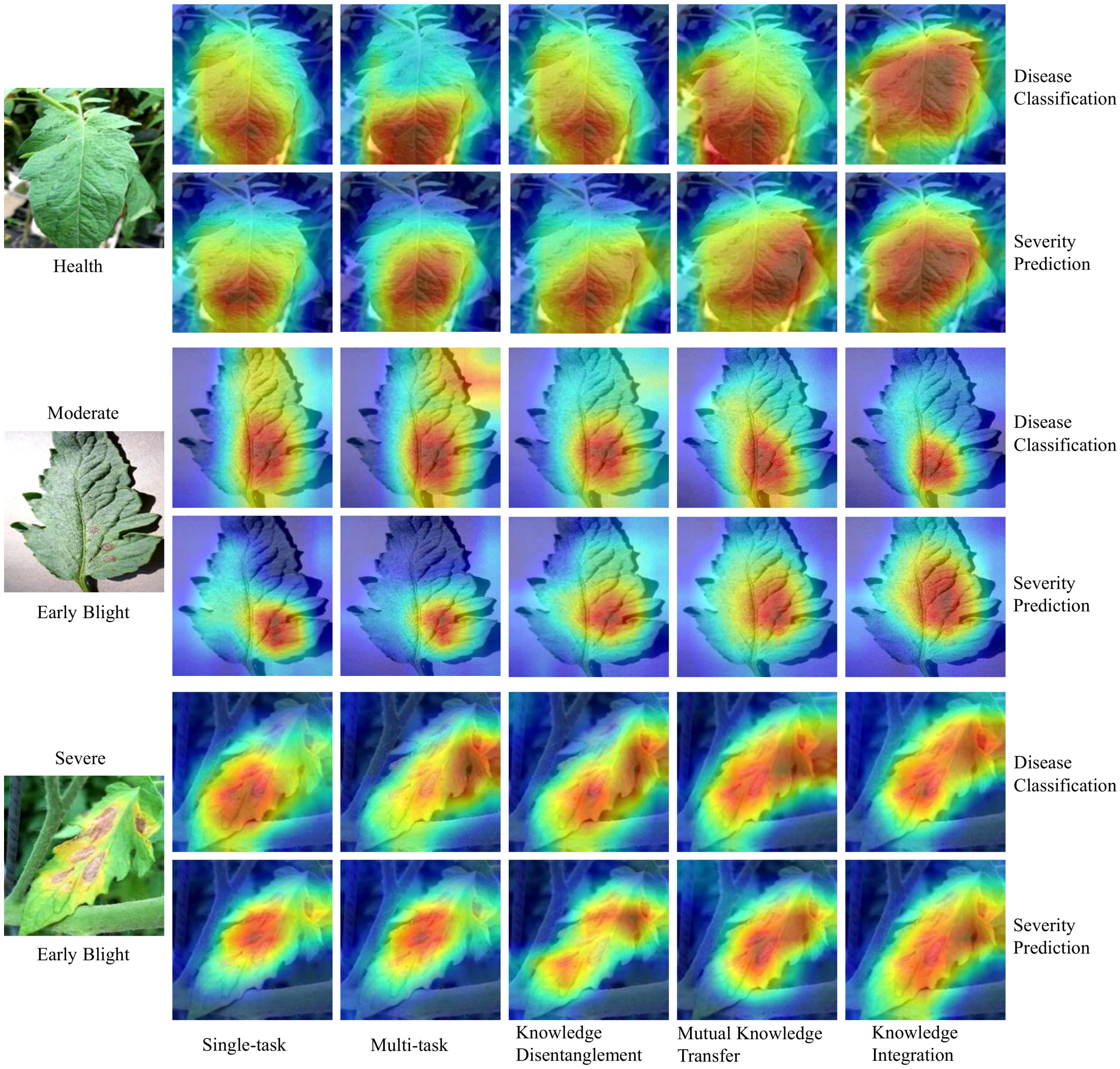
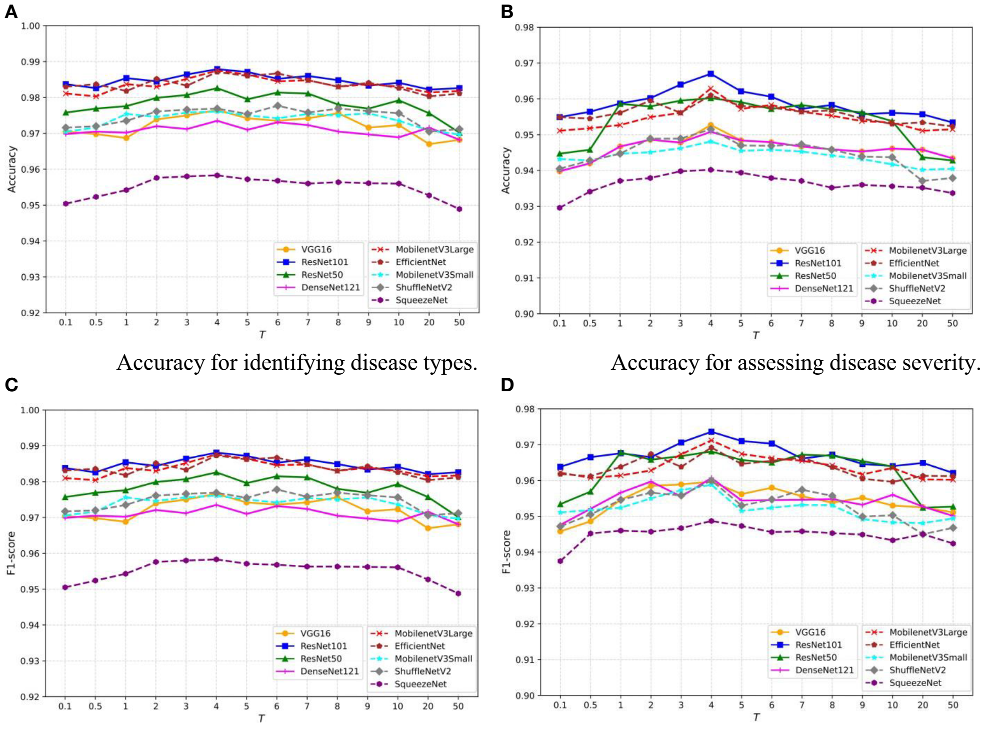


FIGURE 7
Visual analysis of attention shifts in MTDL framework across severity levels.



Accuracy for identifying disease types.

Accuracy for assessing disease severity.

F1-score for identifying disease types.

F1-score for assessing disease severity.

FIGURE 8
Sensitivity analysis of temperature hyperparameter *T* in MTDL framework. **(A)** Accuracy for identifying disease types, **(B)** Accuracy for assessing disease severity, **(C)** F1-score for identifying disease types, **(D)** F1-score for assessing disease severity.

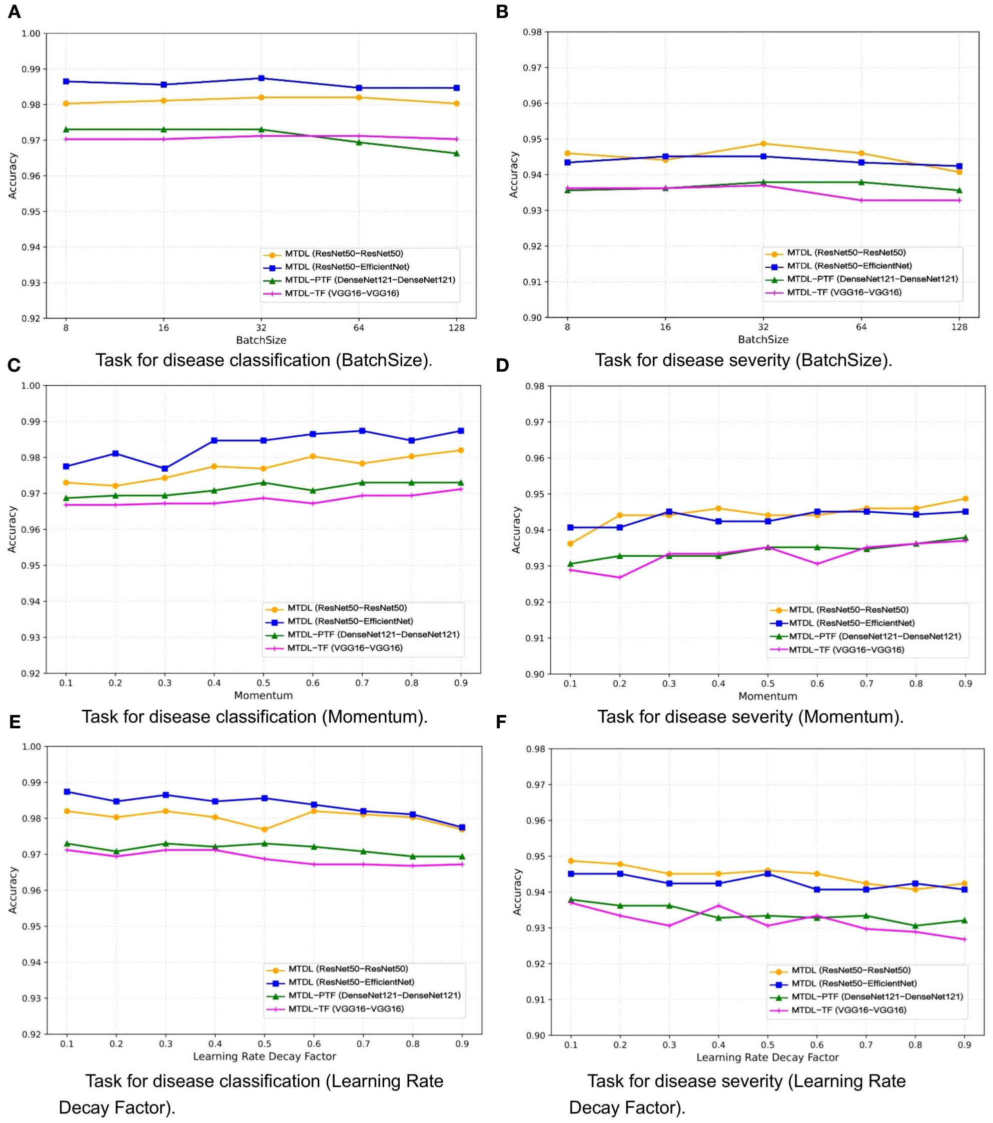
performance remains relatively stable for *T* within the interval [3,8]. This indicates that our model is robust to the choice of *T* within this range, providing flexibility in practical applications.

One the other hand, the selection of a batch size of 32, momentum of 0.9, and learning rate decay factor of 0.1 was guided by a combination of empirical conventions and experimental validation aimed at striking a balance between computational efficiency and model performance. To validate the impact of different parameter settings on performance, we analyzed MTDL and its variants on the validation set for varying batch sizes (Figures 9A, B), momentum (Figures 9C, D), and learning rate decay factors (Figures 9E, F), detailing their effects on Accuracy. We can see that Accuracy remains relatively stable across batch sizes that varies (8, 16, 32, 64, 128), with the optimal average Accuracy achieved at 32. This is likely because a moderate batch size balances gradient estimation Accuracy and the beneficial noise of stochasticity, optimizing learning. As momentum increases from 0.1 to 0.9, Accuracy generally improves. A higher momentum, like 0.9, effectively uses past gradients to accelerate convergence and navigate through local minima, leading to better performance compared to a lower setting like 0.1. Moreover, increasing decay factors tend to lower Accuracy, potentially due to a swift reduction in the learning rate and premature convergence. An optimal decay factor is one that slowly decreases the learning rate, facilitating precise adjustments as the model converges to the best solution.

4 Conclusion

In this work, we present the multi-task distillation learning (MTDL) framework, a specialized solution for diagnosing tomato diseases. The framework comprises three key stages: knowledge disentanglement, mutual knowledge transfer, and knowledge integration. Using this staged learning approach, we leverage the complementary aspects of different tasks to enhance performance across various network architectures. Moreover, our framework adeptly balances performance with efficiency, underlining its potential for practical applications. Although MTDL enhances traditional knowledge distillation with bidirectional knowledge transfer between teacher and student models, it extends training time due to a progressive, multi-stage learning approach. To mitigate this, we introduce MTDL-PTF and MTDL-TF variants for efficiency, though they may slightly underperform compared to the original MTDL.

Furthermore, our current framework has some limitations. First, although the framework is designed for outdoor environments, it has stringent requirements for the subject being



Task for disease classification (BatchSize).

Task for disease severity (BatchSize).

Task for disease classification (Momentum).

Task for disease severity (Momentum).

Task for disease classification (Learning Rate Decay Factor).

Task for disease severity (Learning Rate Decay Factor).

FIGURE 9
Effect of different parameters on model performance. **(A)** Task for disease classification (BatchSize), **(B)** Task for disease severity (BatchSize), **(C)** Task for disease classification (Momentum), **(D)** Task for disease severity (Momentum), **(E)** Task for disease classification (Learning Rate Decay Factor), **(F)** Task for disease severity (Learning Rate Decay Factor).

photographed, focusing mainly on recognizing single subjects in images. Second, the severity level classification is relatively basic, encompassing only three levels, including a healthy state. In future work, we plan to integrate object localization techniques into the distillation process to facilitate the identification of multiple leaves in images. Additionally, we aim to refine the classification of disease severity levels, focusing especially on the early detection of diseases. These planned enhancements will contribute to the development of more sophisticated and nuanced solutions in the field of tomato disease diagnosis, offering a robust framework for sustainable and intelligent agriculture.

Data availability statement

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

Author contributions

BL: Conceptualization, Methodology, Writing – original draft. SW: Data curation, Formal analysis, Software, Writing – review & editing. FZ: Funding acquisition, Methodology, Writing – review & editing.

NG: Formal analysis, Validation, Writing – review & editing. HF: Data curation, Formal analysis, Validation, Writing – review & editing. WY: Project administration, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the National Natural Science Foundation of China (No. 61972132), the S&T Program of Hebei (Nos. 20327404D, 21327404D, 21327407D), the Natural Science Foundation of Hebei Province, China (Nos. F2020204009, C2023204069), and the Research Project for Self-cultivating Talents of Hebei Agricultural University (No. PY201810).

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

Albahli, S., and Nawaz, M. (2022). Dcnet: Densenet-77-based cornernet model for the tomato plant leaf disease detection and classification. *Front. Plant Sci.* 13, 957961. doi: 10.3389/fpls.2022.957961

Atila, Ü, Ucar, M., Akyol, K., and Ucar, E. (2021). Plant leaf disease classification using efficientnet deep learning model. *Ecol. Inf.* 61, 101182. doi: 10.1016/j.ecoinf.2020.101182

Barbedo, J. G. A. (2019). Plant disease identification from individual lesions and spots using deep learning. *Biosyst. Eng.* 180, 96–107. doi: 10.1016/j.biosystemseng.2019.02.002

Basavaiah, J., and Arlene Anthony, A. (2020). Tomato leaf disease classification using multiple feature extraction techniques. *Wireless Pers. Commun.* 115, 633–651. doi: 10.1007/s11277-020-07590-x

Bhujel, A., Kim, N.-E., Arulmozhi, E., Basak, J. K., and Kim, H.-T. (2022). A lightweight attention-based convolutional neural networks for tomato leaf disease classification. *Agriculture* 12, 228. doi: 10.3390/agriculture12020228

Bi, C., Wang, J., Duan, Y., Fu, B., Kang, J.-R., and Shi, Y. (2022). Mobilenet based apple leaf diseases identification. *Mobile Networks Appl.* 27, 172–180. doi: 10.1007/s11036-020-01640-1

Botineştean, C., Gruia, A. T., and Jianu, I. (2015). Utilization of seeds from tomato processing wastes as raw material for oil production. *J. Material Cycles Waste Manage.* 17, 118–124. doi: 10.1007/s10163-014-0231-4

Boulent, J., Foucher, S., Theau, J., and St-Charles, P.-L. (2019). Convolutional neural networks for the automatic identification of plant diseases. *Front. Plant Sci.* 10, 941. doi: 10.3389/fpls.2019.00941

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

Corder, G. W., and Foreman, D. I. (2014). *Nonparametric statistics: A step-by-step approach* (John Wiley & Sons).

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

Dataset AI Challenger (2018) *AI Challenger 2018 Datasets*. Available at: https://github.com/AIChallenger/AI_Challenger_2018 (Accessed Nov. 1, 2022).

Deng, Y., Xi, H., Zhou, G., Chen, A., Wang, Y., Li, L., et al. (2023). An effective image-based tomato leaf disease segmentation method using mc-unet. *Plant Phenomics* 5, 0049. doi: 10.34133/plantphenomics.0049

Ghofrani, A., and Toroghi, R. M. (2022). Knowledge distillation in plant disease recognition. *Neural Computing Appl.* doi: 10.1007/s00521-021-06882-y

Gupta, A. (2022). A segmentation algorithm for the leaf area identification in plant's images. *Sci. Technol. Asia*, 171–178. doi: 10.14456/scitechasia.2022.33

Habib, M. T., Majumder, A., Jakaria, A., Akter, M., Uddin, M. S., and Ahmed, F. (2020). Machine vision based papaya disease recognition. *J. King Saud University-Computer Inf. Sci.* 32, 300–309. doi: 10.1016/j.jksuci.2018.06.006

Harakannanavar, S. S., Rudagi, J. M., Puranikmath, V. I., Siddiqua, A., and Pramodhini, R. (2022). Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proc.* 3, 305–310. doi: 10.1016/j.gltp.2022.03.016

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. (Piscataway, NJ: IEEE), 770–778.

Hinton, G. E., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. Available at: https://arxiv.org/abs/1503.02531.

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., et al. (2019). Searching for mobilenetv3. *Proc. IEEE/CVF Int. Conf. Comput. vision.* 2019, 1314–1324. doi: 10.1109/ICCV.2019.00140

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). "Mobilenets: Efficient convolutional neural networks for mobile vision applications," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seoul, Korea (South). (Piscataway, NJ: IEEE).

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, USA. (Piscataway, NJ: IEEE).

Huang, M.-L., and Chang, Y.-H. (2020). Dataset of tomato leaves. *Mendeley Data* 1.

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*, Hawaii, USA. (Munich, Germany), 4700–4708.

Hughes, D., and Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. Available at: https://arxiv.org/abs/1511.08060.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2016). Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size. Available at: https://arxiv.org/abs/1602.07360.

Ilyas, T., Jin, H., Siddique, M. I., Lee, S. J., Kim, H., and Chua, L. (2022). Diana: A deep learning-based paprika plant disease and pest phenotyping system with disease severity analysis. *Front. Plant Sci.* 13, 983625. doi: 10.3389/fpls.2022.983625

Janarthan, S., Thuseethan, S., Rajasegarar, S., and Yearwood, J. (2022). P2op—plant pathology on palms: A deep learning-based mobile solution for in-field plant disease detection. *Comput. Electron. Agric.* 202, 107371. doi: 10.1016/j.compag.2022.107371

Ji, M., and Wu, Z. (2022). Automatic detection and severity analysis of grape black measles disease based on deep learning and fuzzy logic. *Comput. Electron. Agric.* 193, 106718. doi: 10.1016/j.compag.2022.106718

Ji, M., Zhang, K., Wu, Q., and Deng, Z. (2020). Multi-label learning for crop leaf diseases recognition and severity estimation based on convolutional neural networks. *Soft Computing* 24, 15327–15340. doi: 10.1007/s00500-020-04866-z

Jiang, Z., Dong, Z., Jiang, W., and Yang, Y. (2021). Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. *Comput. Electron. Agric.* 186, 106184. doi: 10.1016/j.compag.2021.106184

Karlekar, A., and Seal, A. (2020). Soynet: Soybean leaf diseases classification. *Comput. Electron. Agric.* 172, 105342. doi: 10.1016/j.compag.2020.105342

Komodakis, N., and Zagoruyko, S. (2016). Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In. Available at: https://arxiv.org/abs/1612.03928.

Kumar, M., Chandran, D., Tomar, M., Bhuyan, D. J., Grasso, S., Sá, A. G. A., et al. (2022). Valorization potential of tomato (solanum lycopersicum l.) seed: nutraceutical quality, food properties, safety aspects, and application as a health-promoting ingredient in foods. *Horticulturae* 8, 265. doi: 10.3390/horticulturae8030265

Kumar, K. S., Paswan, S., and Srivastava, S. (2012). Tomato-a natural medicine and its health benefits. *J. Pharmacognosy Phytochem.* 1, 33–43.

Li, X., Li, X., Zhang, S., Zhang, G., Zhang, M., and Shang, H. (2023). Slvit: Shuffle-convolution-based lightweight vision transformer for effective diagnosis of sugarcane leaf diseases. *J. King Saud University-Computer Inf. Sci.* 35, 101401. doi: 10.1016/j.jksuci.2022.09.013

Liu, B.-Y., Fan, K.-J., Su, W.-H., and Peng, Y. (2022). Two-stage convolutional neural networks for diagnosing the severity of alternaria leaf blotch disease of the apple tree. *Remote Sens.* 14, 2519. doi: 10.3390/rs14112519

Meenakshi, K., Swaraja, K., and Ch, U. K. (2019). "Grading of quality in tomatoes using multi-class svm," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)* (Erode, India: Surya Engineering College, IEEE). 104–107.

Mokhtar, U., Ali, M. A., Hassanien, A. E., and Hefny, H. (2015). "Identifying two of tomatoes leaf viruses using support vector machine," in *Information Systems Design and Intelligent Applications: Proceedings of Second International Conference INDIA 2015* (Kalyani, India), Vol. 1. 771–782.

Nanehkaran, Y., Zhang, D., Chen, J., Tian, Y., and Al-Nabhan, N. (2020). Recognition of plant leaf diseases based on computer vision. *J. Ambient Intell. Humanized Computing* 2020, 1–18. doi: 10.1007/s12652-020-02505-x

Ozguven, M. M., and Adem, K. (2019). Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. *Physica A: Stat. Mechanics its Appl.* 535, 122537. doi: 10.1016/j.physa.2019.122537

Pal, A., and Kumar, V. (2023). Agridet: Plant leaf disease severity classification using agriculture detection framework. *Eng. Appl. Artif. Intell.* 119, 105754. doi: 10.1016/j.engappai.2022.105754

Patil, P., Yaligar, N., and Meena, S. (2017). "Comparision of performance of classifiers-svm, rf and ann in potato blight disease detection using leaf images," in *2017 IEEE International Conference on Computational Intelligence and Computing research (ICCIC)* Tamil Nadu, India. (Piscataway, NJ: IEEE), 1–5.

Rahman, S. U., Alam, F., Ahmad, N., and Arshad, S. (2022). Image processing based system for the detection, identification and treatment of tomato leaf diseases. *Multimedia Tools Appl.* 82, 9431–9445. doi: 10.1007/s11042-022-13715-0

Roy, A. M., and Bhaduri, J. (2021). A deep learning enabled multi-class plant disease detection model based on computer vision. *AI.* 2 (3), 413–428. doi: 10.3390/ai2030026

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. *Proc. IEEE Int. Conf. Comput. Vision* 2017, 618–626. doi: 10.1109/ICCV.2017.74

Septiyanti, M., and Meliana, Y. (2020). Characterization of nanoemulsion gotukola, mangosteen rind, cucumber and tomato extract for cosmetic raw material. *J. Physics: Conf. Ser. (IOP Publishing)* 1442, 012046. doi: 10.1088/1742-6596/1442/1/012046

Sharif, M., Khan, M. A., Iqbal, Z., Azam, M. F., Lali, M. I. U., and Javed, M. Y. (2018). Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. *Comput. Electron. Agric.* 150, 220–234. doi: 10.1016/j.compag.2018.04.023

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, 1031748. doi: 10.3389/fpls.2022.1031748

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. Available at: https://arxiv.org/abs/1409.1556.

Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., and Batra, N. (2020). Plantdoc: A dataset for visual plant disease detection. *Proc. 7th ACM IKDD CoDS 25th COMAD*, 249–253. doi: 10.1145/3371158.3371196

Singh, P., Verma, A., and Alex, J. S. R. (2021). Disease and pest infection detection in coconut tree through deep learning techniques. *Comput. Electron. Agric.* 182, 105986. doi: 10.1016/j.compag.2021.105986

Sujatha, R., Chatterjee, J. M., Jhanjhi, N., and Brohi, S. N. (2021). Performance of deep learning vs machine learning in plant leaf disease detection. *Microprocessors Microsystems* 80, 103615. doi: 10.1016/j.micpro.2020.103615

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

Thai, H.-T., Le, K.-H., and Nguyen, N. L.-T. (2023). Formerleaf: An efficient vision transformer for cassava leaf disease detection. *Comput. Electron. Agric.* 204, 107518. doi: 10.1016/j.compag.2022.107518

Thuseethan, S., Vigneshwaran, P., Charles, J., and Wimalasooriya, C. (2022). Siamese network-based lightweight framework for tomato leaf disease recognition. Available at: https://arxiv.org/abs/2209.11214.

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., and Zhu, H. (2021). A cucumber leaf disease severity classification method based on the fusion of deeplabv3+ and u-net. *Comput. Electron. Agric.* 189, 106373. doi: 10.1016/j.compag.2021.106373

Wang, G., Sun, Y., and Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. *Comput. Intell. Neurosci.* 2017, 2917536. doi: 10.1155/2017/2917536

Wang, D., Wang, J., Ren, Z., and Li, W. (2022). Dhbp: A dual-stream hierarchical bilinear pooling model for plant disease multi-task classification. *Comput. Electron. Agric.* 195, 106788. doi: 10.1016/j.compag.2022.106788

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, Berlin: Springer), 3–19.

Wu, J., Wen, C., Chen, H., Ma, Z., Zhang, T., Su, H., et al. (2022). Ds-detr: A model for tomato leaf disease segmentation and damage evaluation. *Agronomy* 12, 2023. doi: 10.3390/agronomy12092023

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. (2020). "Revisiting knowledge distillation via label smoothing regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. (Virtual, Piscataway, NJ: IEEE), 3903–3911.

Zeng, W., Li, H., Hu, G., and Liang, D. (2022). Lightweight dense-scale network (ldsnet) for corn leaf disease identification. *Comput. Electron. Agric.* 197, 106943. doi: 10.1016/j.compag.2022.106943

Zhang, S., Griffiths, J. S., Marchand, G., Bernards, M. A., and Wang, A. (2022). Tomato brown rugose fruit virus: An emerging and rapidly spreading plant rna virus that threatens tomato production worldwide. *Mol. Plant Pathol.* 23, 1262–1277. doi: 10.1111/mpp.13229

Zhang, J.-H., Kong, F.-T., Wu, J.-Z., Han, S.-Q., and Zhai, Z.-F. (2018a). Automatic image segmentation method for cotton leaves with disease under natural environment. *J. Integr. Agric.* 17, 1800–1814. doi: 10.1016/S2095-3119(18)61915-X

Zhang, J., Rao, Y., Man, C., Jiang, Z., and Li, S. (2021). Identification of cucumber leaf diseases using deep learning and small sample size for agricultural internet of things. *Int. J. Distributed Sensor Networks* 17, 15501477211007407. doi: 10.1177/15501477211007407

Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018b). "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Munich, Germany, (Munich, Germany). 6848–6856.

Zhao, B., Cui, Q., Song, R., Qiu, Y., and Liang, J. (2022). "Decoupled knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, USA. (Piscataway, NJ: IEEE), 11953–11962.

Zhao, S., Peng, Y., Liu, J., and Wu, S. (2021). Tomato leaf disease diagnosis based on improved convolution neural network by attention module. *Agriculture* 11, 651. doi: 10.3390/agriculture11070651

TYPE Original Research
PUBLISHED 06 June 2024
DOI 10.3389/fpls.2024.1383863

Check for updates

OPEN ACCESS

EDITED BY
Chunlei Xia,
Chinese Academy of Sciences (CAS), China

REVIEWED BY
Yunchao Tang,
Dongguan University of Technology, China
Aibin Chen,
Central South University Forestry and Technology, China
Ronnie Concepcion II,
De La Salle University, Philippines

*CORRESPONDENCE
Jianhua Zhang
✉ zhangjianhua@caas.cn
Lin Hu
✉ hulin@caas.cn

RECEIVED 08 February 2024
ACCEPTED 16 May 2024
PUBLISHED 06 June 2024

CITATION
Pan P, Shao M, He P, Hu L, Zhao S, Huang L, Zhou G and Zhang J (2024) Lightweight cotton diseases real-time detection model for resource-constrained devices in natural environments.
Front. Plant Sci. 15:1383863.
doi: 10.3389/fpls.2024.1383863

Lightweight cotton diseases real-time detection model for resource-constrained devices in natural environments

Pan Pan[1,2,3], Mingyue Shao[1,2,3], Peitong He[1,2,3], Lin Hu[1,2,3*], Sijian Zhao[1], Longyu Huang[3,4], Guomin Zhou[1,2,3,5] and Jianhua Zhang[1,2,3*]

[1]Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing, China, [2]National Agriculture Science Data Center, Beijing, China, [3]National Nanfan Research Institute (Sanya), Chinese Academy of Agricultural Sciences, Sanya, China, [4]Institute of Cotton Research, Chinese Academy of Agricultural Sciences, Anyang, China, [5]Farmland Irrigation Research Institute, Chinese Academy of Agricultural Sciences, Xinxiang, China

Cotton, a vital textile raw material, is intricately linked to people's livelihoods. Throughout the cotton cultivation process, various diseases threaten cotton crops, significantly impacting both cotton quality and yield. Deep learning has emerged as a crucial tool for detecting these diseases. However, deep learning models with high accuracy often come with redundant parameters, making them challenging to deploy on resource-constrained devices. Existing detection models struggle to strike the right balance between accuracy and speed, limiting their utility in this context. This study introduces the CDDLite-YOLO model, an innovation based on the YOLOv8 model, designed for detecting cotton diseases in natural field conditions. The C2f-Faster module replaces the Bottleneck structure in the C2f module within the backbone network, using partial convolution. The neck network adopts Slim-neck structure by replacing the C2f module with the GSConv and VoVGSCSP modules, based on GSConv. In the head, we introduce the MPDIoU loss function, addressing limitations in existing loss functions. Additionally, we designed the PCDetect detection head, integrating the PCD module and replacing some CBS modules with PCDetect. Our experimental results demonstrate the effectiveness of the CDDLite-YOLO model, achieving a remarkable mean average precision (mAP) of 90.6%. With a mere 1.8M parameters, 3.6G FLOPS, and a rapid detection speed of 222.22 FPS, it outperforms other models, showcasing its superiority. It successfully strikes a harmonious balance between detection speed, accuracy, and model size, positioning it as a promising candidate for deployment on an embedded GPU chip without sacrificing performance. Our model serves as a pivotal technical advancement, facilitating timely cotton disease detection and providing valuable insights for the design of detection models for agricultural inspection robots and other resource-constrained agricultural devices.

KEYWORDS

cotton diseases detection, natural environment, deep learning, lightweight, YOLOv8

1 Introduction

Cotton, a member of the Malvaceae family (Chohan et al., 2020), holds the top position among natural fibers, thanks to its simplicity of cultivation and its wide range of uses in clothing and home textiles. It satisfies nearly 35% of the global annual fiber demand (Huang et al., 2021). Beyond the textile industry, cotton plays a crucial role in the production of animal feed and edible oil (Townsend, 2020; Zaidi et al., 2020). In 75 countries, cotton crop production supports the livelihoods of over 250 million people (Wang et al., 2020).

Throughout the cotton growth cycle, diseases can significantly hinder both yield and quality, posing a substantial threat to the economic viability of farmers (Chi et al., 2021). According to statistics, estimates of the total cotton disease losses ranged from 6% to 12% of the yield lost due to disease (Lawrence et al., 2022). Among cotton diseases, verticillium wilt (Cai et al., 2009), fusarium wilt (Wang et al., 2009), and anthracnose (Nawaz et al., 2018) are particularly significant (Toscano-Miranda et al., 2022). They are often referred to as the 'cancer' of cotton crops due to their ability to substantially reduce cotton production.

The battle against cotton diseases endures, with ongoing efforts to avert crop losses by early and effective disease detection, followed by timely intervention (Mohanty et al., 2016; Guo et al., 2022). While manual disease detection is the prevailing approach, it is hampered by reliability issues and is impractical for large-scale monitoring due to time and cost constraints (Peyal et al., 2022). The quest for automated cotton disease detection methods is becoming increasingly urgent, particularly given the rapid growth of the cotton industry (Pan et al., 2023b).

Over the past two decades, image-processing techniques for identifying plant diseases have yielded promising results (Thakur et al., 2022). With recent advancements in machine learning, these techniques offer the potential to reduce labor costs, minimize time wastage, and enhance plant quality (Wani et al., 2022). However, traditional machine learning algorithms predominantly rely on manually crafted, low-level visual features based on engineering experience. This limitation often leads to subpar performance when dealing with complex scenes (Wang et al., 2022b). Consequently, further research is required to develop more efficient and automated methods (Zhang et al., 2023e).

Deep learning algorithms exhibit the capability to autonomously extract and learn complex high-level features through deeply structured convolutional neural networks. Due to its rapid evolution, deep learning models have been constructed for the detection of plant diseases (Pan et al., 2023a). These models not only excel in disease classification but also accurately determine disease locations on plant leaves within images (Liu and Wang, 2021). Much like other research domains such as medical science, mechanical automation, and logistics, the integration of robotics and deep learning into agriculture has sparked a revolution in the way plants are cultivated and safeguarded (Balaska et al., 2023). This transformative approach allows for the intelligent application of chemical sprays, including fungicides, herbicides, and pesticides, following successful robotic disease detection. This intelligent strategy offers the promise of establishing a cost-effective crop protection system (Saleem et al., 2021). This innovative approach has been applied to a wide range of crops, including cucumber (Li et al., 2023b), maize (Leng et al., 2023), potato (Johnson et al., 2021; Dai et al., 2022), rice (Jia et al., 2023), soybeans (Zhang et al., 2021), strawberry (Zhao et al., 2022), tomato (Tang et al., 2023b), and wheat (Zhang et al., 2023a), on a global scale for disease detection using deep learning techniques.

In recent years, researchers have harnessed deep learning techniques for the detection of cotton diseases. Several noteworthy studies have been conducted: Susa et al (Susa et al., 2022). applied the YOLOv3 algorithm to detect and classify cotton plants and leaves, achieving a remarkable mean Average Precision (mAP) score of 95.09%. Zhang et al (Zhang et al., 2023b). optimized the YOLOv5 algorithm to address the issue of subpar small target detection in the context of cotton wilt disease. They introduced a small target detection layer and incorporated an attention mechanism, resulting in an impressive mAP score of 91.13%. PRIYA et al (Priya et al., 2021). utilized Faster R-CNN with Region Proposal Network (RPN) to detect and classify images containing both healthy and diseased cotton plant leaves. Their approach demonstrated an average accuracy of 96% in disease identification. R. Devi Priya et al (Devi Priya et al., 2022). proposed the Augmented Faster R-CNN (AFR-CNN) algorithm by amalgamating Faster R-CNN, an efficient deep learning algorithm, with effective data augmentation techniques such as rotation, blur transformation, flipping, and GAN. The model achieved a noteworthy mAP score of 90.2%. Zhang et al (Zhang et al., 2022). introduced a real-time, high-performance detection model based on an improved YOLOX. Their model incorporated features like Efficient Channel Attention (ECA), a hard-Swish activation function, and Focal Loss into YOLOX, resulting in an mAP of 94.60% for cotton disease and pest detection, with a precision rate of 94.04%. Zhang et al (Zhang et al., 2023c). proposed an enhanced attention mechanism YOLOv7 algorithm (CBAM-YOLOv7) for the image detection of diseases and pests like cotton wilt disease. Their model achieved an impressive mean Average Precision (mAP) score of 90.2%.

The endeavors of the researchers mentioned above have undeniably advanced the field of cotton disease detection, providing valuable insights into areas such as dataset augmentation and the optimization of detection algorithms. Nonetheless, the deployment of mobile robots and various edge AI devices often necessitates a trade-off between computational power, power consumption, battery size, and the time between charges. These devices typically operate with significantly less computational power compared to the robust GPU-based systems commonly employed for training and assessing deep neural networks (Yao et al., 2022). Moreover, it has become evident that certain deep learning models with high detection accuracy tend to possess redundant model parameters. This redundancy poses challenges when it comes to deploying these models on mobile agricultural inspection robots. Existing detection models struggle to strike a balance between detection accuracy and speed, hindering their application in this context. Furthermore, it's worth acknowledging that, in some of these studies, cotton disease detection was conducted within controlled environments, and this gap in achieving reliable detection in natural agricultural settings remains (Tang et al., 2023a). This limitation has, to a certain extent, constrained the development of agricultural inspection robots (Wang et al., 2022a; Ye et al., 2023).

Consequently, this study centered on cotton disease as the focal point of research and proposed CDDLite-YOLO detection algorithm to detect cotton disease quickly and accurately under natural field conditions. The model introduced in this paper is built upon the most recent advancements in object detection algorithms with the specific features of cotton diseases. It successfully strikes a harmonious balance between detection speed, accuracy, and model size, making it a promising candidate for deployment on an embedded GPU chip without compromising performance.

The significant contributions of this paper can be summarized as follows:

(1) We collected a dataset of cotton disease images from natural environments for training, validation, and testing of the model.

(2) To enhance detection accuracy while minimizing parameter calculations, we designed the C2f-Faster module as a replacement for the C2f module in the backbone network and introduced a novel Slim-neck structure by substituting the C2f module with the GSConv module and the VoVGSCSP module in the neck network.

(3) We introduced MPDIoU, an IoU loss measure, to address limitations for cotton disease detection that existing loss functions when predicted and ground truth bounding boxes have the same aspect ratio but varying width and height values.

(4) We designed the PCDetect detection head to reduce model parameters and computations while maintaining exceptional detection performance.

(5) Through experiments, we validated the CDDLite-YOLO model. Compared to other models, CDDLite-YOLO achieves higher mAP and detection speed, with lower FLOPs and a smaller model size.

The subsequent sections of this study are structured as follows: Section II explores critical aspects, including image acquisition, preprocessing, and model structure enhancements. Section III presents the experimental results alongside a detailed analysis, while Section IV offers a comprehensive discussion of this study. Section V encapsulates our efforts with a summary of the conclusions reached.

2 Materials and methods

2.1 Materials

2.1.1 Image data acquisition

The image dataset was collected from two specific locations: the cotton fields at the Langfang Research Base of the Chinese Academy of Agricultural Sciences, Hebei Province, China (N: 39°27′55.59″, E: 116°45′28.54″), and the Potianyang Base in Yazhou District, Sanya City, Hainan Province, China (N: 18°23′49.71″, E: 109°10′39.84″). This data collection took place from September 2020 to December 2022.The focus of our image collection comprised three primary types of cotton diseases: verticillium wilt, fusarium wilt, and anthracnose. To ensure the quality and accuracy of the dataset, all images underwent a meticulous identification and confirmation process carried out by two expert cotton pathologists.

Images were captured during different weather conditions, including clear and overcast skies, at various times of the day, covering the morning, noon, and evening. Image capture was carried out using a Canon EOS 850D digital camera (Canon Inc., Tokyo, Japan) and a Huawei P40 Pro smartphone (Huawei Technologies Co., Ltd., Shenzhen, China). The images were captured from a distance of 20–50 cm from the cotton leaves, using automatic exposure mode. They have a resolution of 4608 × 3456 pixels and were saved in JPG format.

To ensure the diversity and richness of our image dataset, a randomized approach was employed during the collection process. This involved capturing images from various angles, under different lighting conditions, and against diverse backgrounds. To accurately reflect natural field conditions, images were taken during different weather conditions, including sunny, cloudy, and overcast weather, across different times of the day, encompassing various growth stages of the cotton crop. The images also include the presence of soil, as well as potential field clutter such as weeds, plastic film, and dried leaves.

2.1.2 Images processing and dataset production

To enhance data collection efficiency, we concurrently captured images and recorded videos. Later, we employed video frame extraction to augment the image count. The recorded videos ranged from 15 to 30 seconds, and frames were extracted at a rate of 15 frames per second, resulting in a range of 225 to 450 frames, and the image resolution is 4608 × 3456, which is saved in JPG format. These frames were then carefully curated for selection. In order to prevent redundancy within the dataset, we adhered to three guiding principles for image selection: (1) ensuring each diseased leaf was represented only once, (2) avoiding multiple images from the one or neighboring cotton plants, and (3) prioritizing images with different angles, various lighting conditions, and diverse backgrounds. Consequently, we curated a dataset for cotton disease detection under natural conditions, comprising 591 images of cotton with verticillium wilt, 435 images of cotton with fusarium wilt, and 504 images of cotton with anthracnose, totaling 1,530 images. For specific details regarding the types of cotton diseases, the number of images in each category, and key disease features within the dataset, please refer to Table 1.

We employed the Make Sense tool (https://makesense.ai) for labeling the types of diseased leaves and their respective positions in the images. The labeling area was defined as the smallest rectangle encompassing the diseased leaf, minimizing background inclusion. The dataset was partitioned into three subsets in an 8:1:1 ratio, with 1224 images allocated to the training set, and 153 images each for both the validation and test sets. Additionally, mosaic augmentation was incorporated into the training process. Mosaic augmentation randomly selects four images, extracting segments of content and their corresponding detection box information. These segments are

TABLE 1 The types, figures, image samples, and key features of each cotton disease in the dataset.

Type of Disease	Figures	Image	Key Features
Verticillium wilt	591		Pale yellow patches develop between leaf margins and veins, gradually expanding and causing the loss of green color in the leaves.
Fusarium wilt	435		Lower leaves exhibit yellowing and wilting. The stem displays brown discoloration and often splits open, revealing red-brown vascular tissue.
Anthracnose	504		Small, circular lesions appear on leaves, stems, and bolls. These lesions start as water-soaked areas and become sunken with dark centers over time.
Total	1530		

then fused into a single image for network input. This method substantially enhances training data diversity, mitigating the risk of overfitting by introducing greater variability into the learning process.

2.2 Methods

2.2.1 Overall model

Object detection algorithms can be categorized into one-stage and two-stage algorithms. The two-stage algorithm relies on region proposals, represented by Faster R-CNN, which is known for its slower processing speed, which makes it unsuitable for real-time detection and deployed on an embedded GPU chip. On the other hand, the one-stage model is based on regression, which includes the YOLO series. offers a significant advantage in speed compared to the two-stage model, making it better suited for real-time detection requirements. Hence, this study opts for the YOLO model as the baseline model. This model is an enhancement of the YOLOv8 model specifically tailored for the task of detecting cotton diseases in natural environments and designed for deployment on agricultural inspection robots and other devices with limited memory and computational resources. The architecture of CDDLite-YOLO is visualized in Figure 1.

The model comprises four key components: Input, Backbone, Neck, and Head. The enhancements are summarized as follows:

(1) We designed the Faster Block structure using partial convolution to replace the Bottleneck structure in the C2f module within the backbone network, resulting in the upgraded C2f module termed C2f-Faster.

(2) In the neck network, we introduce an innovative Slim-neck structure by replacing the C2f module with the GSConv module. Additionally, the C2f modules are enhanced by integrating the VoVGSCSP module. This module is an iterative fusion of the GS bottleneck, built upon GSConv.

(3) We introduced MPDIoU, an IoU loss function based on minimum points distance, to address limitations in existing loss functions in YOLOv8, particularly when dealing with predicted and ground truth bounding boxes of the same aspect ratio but varying width and height values.

(4) We designed the PCDetect detection head, incorporating the PCD module into the detection head and replacing specific CBS modules with PCDetect.

By integrating these advancements, CDDLite-YOLO effectively balances detection speed, accuracy, and model size. It significantly reduces the model's size, accelerates detection speed, and achieves higher detection accuracy, providing a harmonious synergy of performance improvements.

2.2.2 YOLOv8

YOLOv8, the latest addition to the YOLO series, was introduced by Ultralytics in January 2023. It maintains the established YOLO series structure while undergoing significant optimization, resulting in notable improvements in both speed and accuracy (Kang and Kim, 2023).

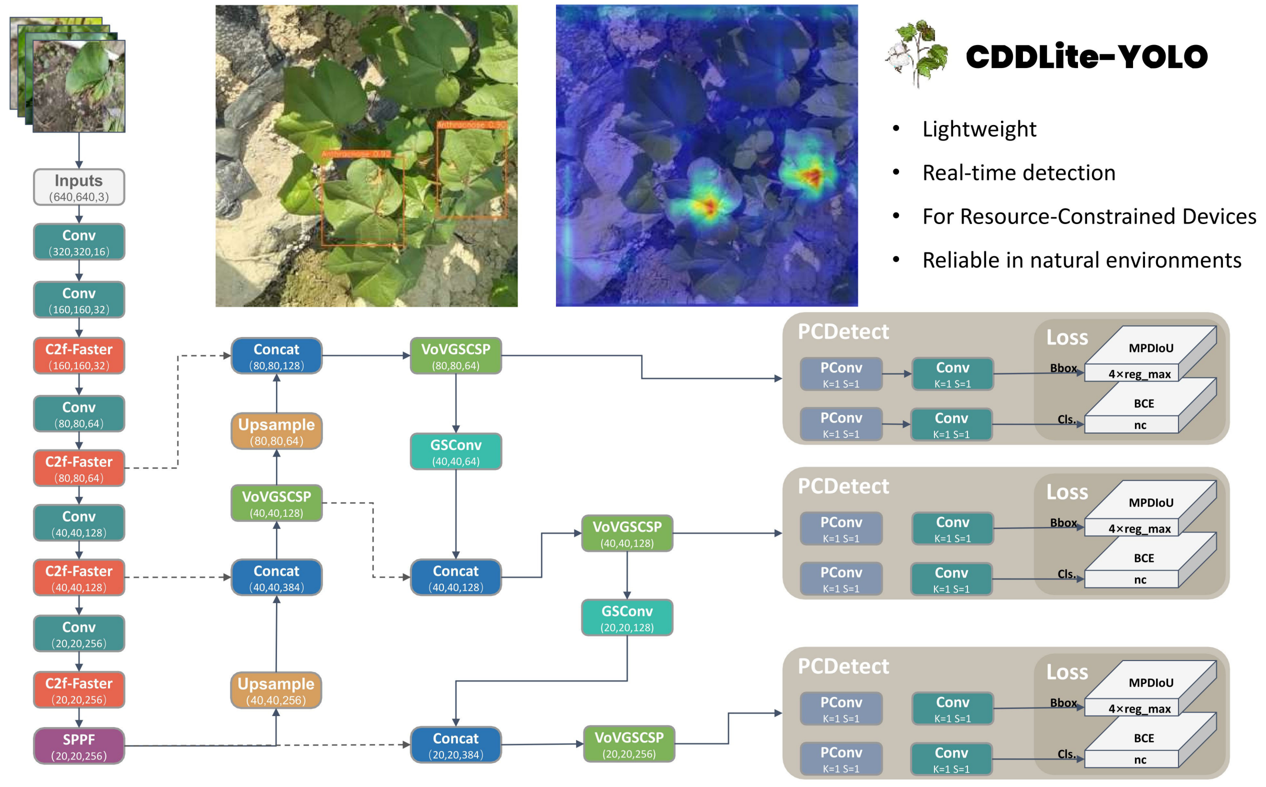


FIGURE 1
Overall model architecture diagram.

YOLOv8 consists of three core components: Backbone, Neck, and Head. The Backbone in YOLOv8 closely mirrors YOLOv5's architecture, with notable refinements to the CSPLayer, now referred to as the C2f module. This C2f module seamlessly integrates high-level features with contextual information, resulting in improved detection accuracy. The Neck of YOLOv8 combines an FPN (Feature Pyramid Network) and PAN (Path Aggregation Network) to facilitate feature fusion among the three effective feature layers obtained in the Backbone. In the Head of YOLOv8, a shift occurs from an anchor-based to an anchor-free approach (Terven and Cordova-Esparza, 2023). This transition abandons IOU matching and single-scale assignment, opting instead for a task-aligned assigner to match positive and negative samples.

YOLOv8n, the smallest model in the YOLOv8 series, is distinguished by its compact model parameters and minimal hardware requirements. When trained on the cotton diseases dataset, YOLOv8n surpasses the performance of YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x, yielding notably superior results. Although its mAP value is slightly lower compared to the other four models, YOLOv8n shines with significantly reduced computational costs and fewer parameters. This renders it an optimal choice for deployment on resource-constrained devices.

In this article, we present the CDDLite-YOLO model, built upon YOLOv8n. Our objective is to cater to real-time and resource-constrained device development requirements while upholding detection accuracy in natural field environments.

2.2.3 C2f-faster

In object detection models, the main objective is to extract spatial information from images, which demands a substantial number of convolutional operations. In contrast to YOLOv5's C3 module, YOLOv8's new C2f module incorporates additional Bottleneck structures and cross-layer connections, enhancing gradient flow. However, this also brings about excessive convolution operations and heightened computational load, presenting deployment challenges on resource-limited embedded devices.

To meet the requirements of embedded devices for cotton disease detection, reduce computational complexity, and minimize parameter size, thus achieving a lightweight network model, enhancing the convolution operator within the C2f module stands out as a highly effective and worthwhile approach.

The feature maps exhibit significant similarities across various channels. FasterNet (Chen et al., 2023) introduced the concept of partial convolution, where it applies a regular Conv operation to only a subset of the input channels for spatial feature extraction, leaving the rest unchanged. This approach reduces computational redundancy and memory usage simultaneously, resulting in efficient performance on a wide range of devices. The C2f-faster module is inspired by the lightweight design principles of FasterNet. It utilizes the Faster Block to replace the Bottleneck within the C2f module, as illustrated in the Figure 2.

The Faster Block encompasses three types of blocks: PConv, CBS, and 1×1 Conv. PConv stands for Partial Convolution, and utilizes only 1/4 of the input channels for convolution, leaving the remaining 3/4 channels untouched. The outputs of the convolved 1/ 4 channels are then merged with the untouched 3/4 channels. For contiguous or regular memory access, the first or last consecutive cp channels as the representatives of the whole feature maps for computation. Without loss of generality, we assume the input and output feature maps have the same number of channels, which aims to reduce redundant calculations while preserving the original channel information. Despite 3/4 of the channels not being involved in convolution, they are not discarded. Instead, valuable information can be extracted from these channels in subsequent 1×1 convolutions. This approach enhances the efficiency of spatial feature extraction by reducing redundant computation and memory access concurrently. Additionally, CBS is composed of Conv, batch normalization, and a SILU activation function. To ensure that the processed feature maps maintain their original dimensions and size, the 1×1 Conv layer is utilized to restore the output of the preceding layer.

2.2.4 Slim-neck

The standard convolution (SC) module in YOLOv8 utilizes different convolutional kernels across multiple channels simultaneously, resulting in a higher parameter count and increased computational requirements (FLOP). While lightweight networks like MobileNet (Howard et al., 2017) and ShuffleNet (Zhang et al., 2018) effectively address this issue using Depth-wise Separable Convolutions (DSC), they suffer from reduced feature extraction and fusion capabilities, hindering model detection performance. Such limitations make them unsuitable for real-time cotton disease detection.

To address these challenges, the CDDLite-YOLO model introduces the GSConv module (Li et al., 2022), a lightweight convolution, into the neck section, resulting in a novel Slim- neck structure. The GSConv module utilizes the shuffle operation to seamlessly integrate information from SC into DSC-generated data. In contrast to DSC, GSConv excels at preserving hidden connections while still keeping complexity low, achieving a balanced trade-off between model accuracy and speed.

The GSConv module is primarily constituted by Conv, DWConv, Concat, and Shuffle operations, visually represented in the Figure 3. The construction unfolds as follows:

(1) The input feature map consists of C1 channels.

(2) Half of the channels undergo Standard Convolution (SC), and the remaining half undergo Depthwise Separable Convolution (DSC).

(3) Concatenate the resulting two output feature maps along the channel dimension.

(4) Subject the concatenated feature map to a shuffle operation, resulting in the final output.

(5) The final output feature map now contains C2 channels in total.

VoVGSCSP (Xu et al., 2023) represents an iterative integration that builds upon the GS bottleneck using the foundation of GSConv, as depicted in Figure 3. This process involves segmenting the input feature map's channel count into two portions. The initial segment undergoes Convolution (Conv) for processing, followed by consecutive GS bottleneck modules for feature extraction. Simultaneously, the remaining segment serves as residuals and undergoes a single Convolution operation. The resulting two output feature maps are then concatenated and subjected to an additional Convolution, resulting in the final output. The ultimate output feature map contains a total of C2 channels. This module effectively strikes a balance between model accuracy and speed, concurrently reducing computational load and complexity while preserving commendable accuracy.

We envisioned integrating GSConv and VoVGSCSP into the neck network to create a lightweight model without compromising detection performance, as illustrated in the Figure 3. This enhancement led to a reduction in model parameter calculations,

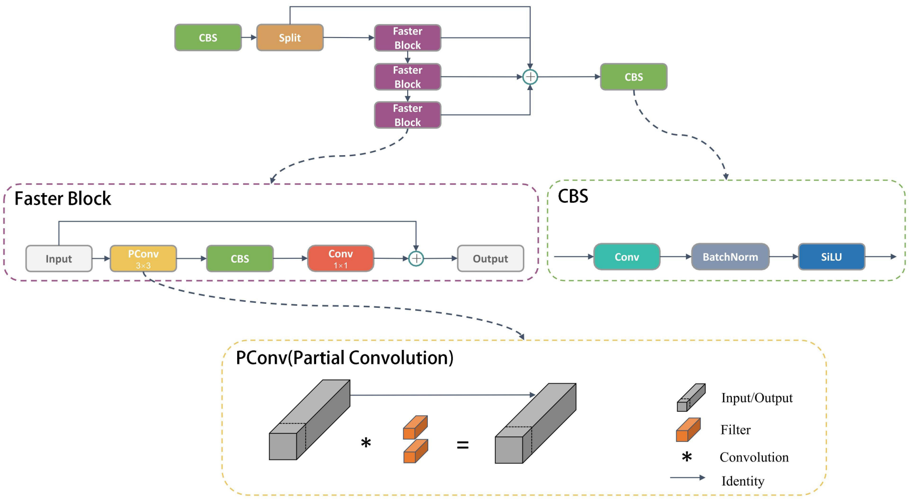


FIGURE 2
Structural diagram of C2f-Faster.

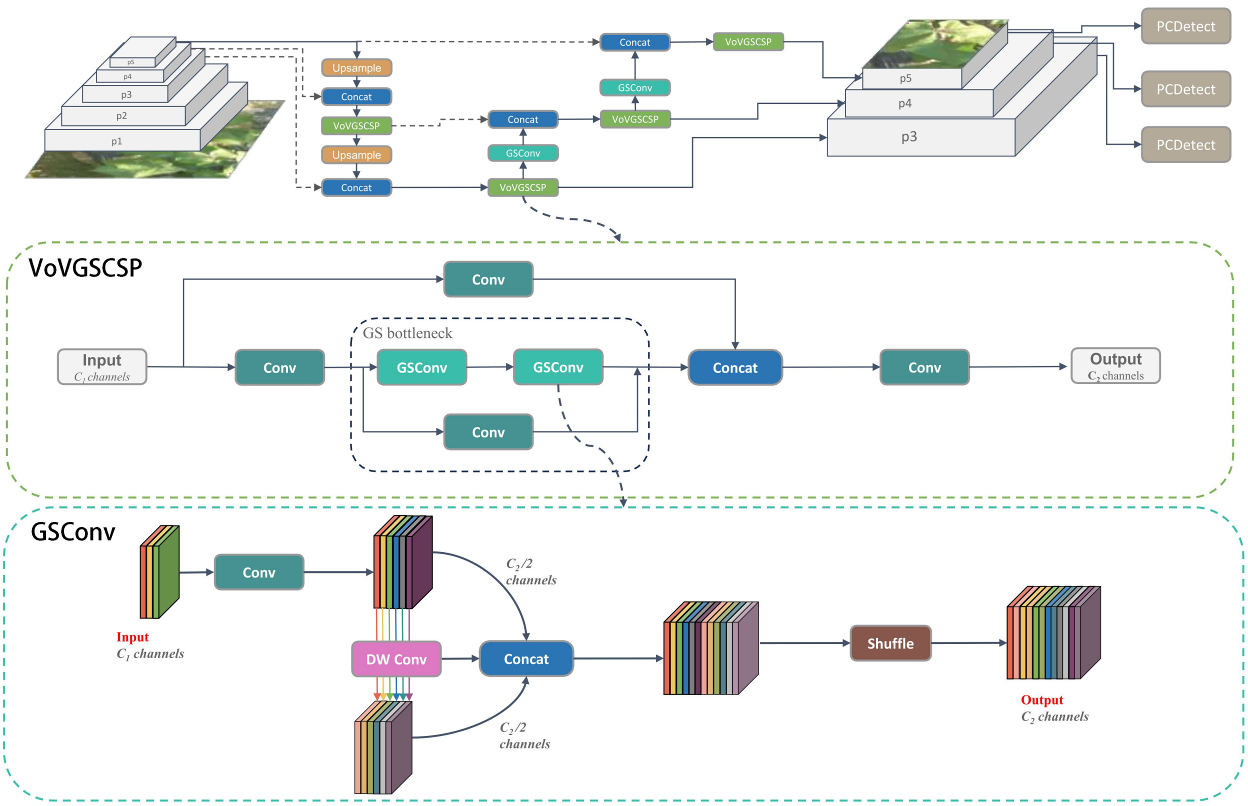


FIGURE 3
Structural diagram of neck.

fostering high detection accuracy and a notable improvement in the balance between the model's accuracy and speed.

2.2.5 MPDIoU

Computing the loss involves comparing the network's predicted results with the groundtruth (Tian et al., 2022). Our model's loss function aligns with YOLOv8, encompassing regression and classification components. YOLOv8 utilizes DFL and CIoU for bounding box regression (Xiao et al., 2023).

The training dataset comprises precisely ground truth bounding boxes that accurately delineate diseased areas. In cotton disease detection, the diverse range of diseases, variations across growth stages, and the influence of factors such as camera angles, lighting conditions, and obstructions can introduce discrepancies in disease localization.However, the aspect ratio definition in CIoU is relative rather than absolute. In instances where predicted and ground truth bounding boxes share the same aspect ratio but differ in width and height, the model may generate boxes with slight deviations (Zhang et al., 2023d). CIoU's sensitivity to such nuances poses challenges for precise learning and prediction, impacting convergence speed and accuracy. To mitigate this, we introduced a novel bounding box similarity comparison metric, MPDIoU (Siliang and Yong, 2023), based on the minimum point distance.

MPDIoU incorporates three key factors: overlapping or non-overlapping area, central points distance, and width and height deviation. It streamlines calculations by minimizing the distance between top-left and bottom-right points in predicted and ground truth bounding boxes. This adaptable metric accommodates overlapping or non-overlapping bounding box regression. Equation 1 shows the computation method for MPDIoU.

$$\begin{aligned} d_1^2 &= (x_1^B - x_1^A)^2 + (y_1^B - y_1^A)^2 \\ d_2^2 &= (x_2^B - x_2^A)^2 + (y_2^B - y_2^A)^2 \\ MPDIoU &= \tfrac{A \cap B}{A \cup B} - \tfrac{d_1^2}{w^2+h^2} - \tfrac{d_2^2}{w^2+h^2} \end{aligned} \tag{1}$$

In the formulation, d_1 and d_2 represent the intersection and minimum point distance. Shapes A and B are two arbitrary convex entities, with w and h signifying the width and height of the input image. The coordinates (x_1^A, y_1^A) and (x_2^A, y_2^A) denote the top-left and bottom-right points of shape A, respectively, and (x_1^B, y_1^B) and (x_2^B, y_2^B) represent the top-left and bottom-right points of shape B.

Benefiting from the implementation of MPDIoU to replace CIoU in YOLOv8, our model has demonstrated competitive results. The subsequent section detailing illustrates that our proposed MPDIoU surpasses the original CIoU and other loss functions.

2.2.6 PCDetect

YOLOv8 introduces the decoupled head mechanism, separating convolutional layers from fully connected layers. This technique leverages neck network output features to predict category and location via distinct branches. While enhancing model convergence and accuracy, the decoupling head introduces additional parameters and computational costs.

To boost computational efficiency, we propose the PCD module, building on PConv from Section 2.2.3. The PCD module features a 3 × 3 PConv layer for extraction, augmented by a CBS module using a 1×1 convolutional kernel for channel adjustment. This enhancement improves feature fusion and cross-channel perception without a substantial parameter increase, enhancing model expressiveness.

The PCD module replaces some CBS modules in the detection head, forming PCDetect. Input and output feature maps are H × W × C. Equation 2 shows the FLOPs ratio of PCD to traditional convolution is only 1/5–1/6 when k = 3, r = 4 (Jiang et al., 2023).

$$\begin{aligned} s &= \frac{FLOPs_{PCD}}{FLOPs_{Conv}} \\ &= \frac{k \times k \times C/r \times W \times H \times C/r + C \times W \times H \times C}{k \times k \times C \times H \times W \times C} \\ &= \frac{1}{r^2} + \frac{1}{k^2} \end{aligned} \tag{2}$$

Substituting PCDetect for the Detection module in YOLOv8 significantly reduces model parameters while maintaining similar detection accuracy. This effectively resolves conflicts between accuracy and detection speed.

3 Experiments and analysis of results

3.1 Experiment settings

3.1.1 Experimental parameter settings

The experimental setup utilized a Dell tower workstation (Dell, Inc., Round Rock, Texas, USA) running Windows 11. It was equipped with a 12th Gen Intel(R) Core(TM) i5–12500 processor operating at 3.00 GHz, 32GB of RAM, a 1TB solid-state drive, and an NVIDIA GeForce RTX 3080 graphics card with 10GB of video memory for GPU-accelerated computing. The software environment included Python 3.8.17, PyTorch 1.13.0, Torchvision 0.14.0, and CUDA 11.7.

The experiment comprised 300 iterations with a batch size of 4. The optimization algorithm used was Adam, with an initial learning rate of 1e-3, a maximum learning rate of 1e-5, a momentum of 0.937, a weight decay of 5e-4, and an input image resolution of 640×640. These training parameters and dataset were consistent across all models during the training process.

3.1.2 Evaluation indicators

To assess the model's performance, various evaluation metrics were used, including Precision, Recall, mAP@0.5, mAP@0.5:0.95, Speed (measured in frames per second or FPS), the number of parameters (Params), and computation costs (FLOPS).

Precision measures the ratio of correctly classified positive samples to all samples predicted as positive, calculated using the formula in Equation 3:

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FP}} \tag{3}$$

Where TP is the true positive samples, and FP is the false positive samples.

Recall quantifies the proportion of actual positive samples correctly identified by the model, calculated using Equation 4:

$$\mathbf{Recall} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FN}} \tag{4}$$

mAP, which stands for mean Average Precision, is determined through the precision-recall (PR) curve and is calculated using Equation 5:

$$\boldsymbol{mAP} = \frac{\sum_{i=1}^{N} \boldsymbol{APi}}{\boldsymbol{N}} \tag{5}$$

Where mAP@0.5 is the average AP with an IoU of 0.5, and mAP@0.5:0.95 is the average AP with IoU values ranging from 0.5 to 0.95 in steps of 0.05.

The number of parameters (Params) reflects the model's complexity and its capacity to learn and represent features. It's calculated using Equation 6:

$$\mathbf{Param} = \sum (\boldsymbol{K} \times \boldsymbol{K} \times \boldsymbol{C}_{\mathbf{in}} \times \boldsymbol{C}_{\mathbf{out}}) \tag{6}$$

Where K represents the convolution kernel size, C_{in} is the input size, and C_{out} is the output size.

Speed is measured in frames per second (FPS), calculated using Equation 7:

$$\mathbf{speed{=}frames/time} \tag{7}$$

FLOPS (Floating-Point Operations Per Second) represents the model's computation costs, and its calculation is detailed in Equation 8:

$$\mathbf{FLOPs} = \sum (\mathbf{K} \times \mathbf{K} \times \mathbf{C}_{\mathbf{in}} \times \mathbf{C}_{\mathbf{out}} \times \mathbf{H} \times \mathbf{W}) \tag{8}$$

Where H × W is the size of the outputted feature map.

3.2 Analysis of results

3.2.1 Ablation experiments

For a more in-depth evaluation of the effectiveness of the enhancement technique in the CDDLite-YOLO model, we performed a series of ablation experiments. We used YOLOv8 as the baseline model for comparison, and the results can be found in Table 2.

(1) Effects of C2f-Faster: A comparative analysis between YOLOv8 and experiments involving the gradual addition of the C2f-Faster module highlights its effectiveness. The incorporation of C2f-Faster significantly reduces computational costs, with a 13.41% reduction in FLOPS and a 13.33% decrease in Params. Simultaneously, it modestly enhances feature extraction capabilities, leading to a 1.3% increase in mAP@0.5. This demonstrates that C2f-Faster not only significantly reduces parameters but

TABLE 2 Comparisons of ablation experiments.

BaseLine	C2f-Faster	Slim- neck	MPDIoU	PCDetect	mAP@0.5	FLOPS/G	Params/M
✓					88.6%	8.2	3.0
✓	✓				89.9%	7.1	2.6
✓	✓	✓			89.3%	6.2	2.4
✓	✓	✓	✓		90.2%	6.2	2.4
✓		✓			90.0%	7.3	2.7
✓		✓	✓		90.1%	7.3	2.7
✓		✓	✓	✓	89.6%	4.7	2.2
✓			✓		90.7%	8.2	3.0
✓			✓	✓	89.4%	5.6	2.4
✓				✓	89.0%	5.6	2.4
✓	✓	✓	✓	✓	**90.6%**	**3.6**	**1.8**

The bold values in Table 2 represent the model proposed in this paper.

also reduces computational costs without compromising detection accuracy.

(2) Effects of Slim-neck: A comparison between YOLOv8 and experiments involving the gradual integration of the Slim-neck module reveals that the inclusion of the Slim-neck contributes to a reduction in computational costs. It leads to a notable 10.98% reduction in FLOPS and a 10.00% decrease in Params. Simultaneously, it provides a modest enhancement in feature extraction capabilities, resulting in a 1.4% increase in mAP@0.5. When both C2f-Faster and Slim-neck are added, computational costs experience a significant decrease, with FLOPS and Params decreasing by 24.39% and 20.00%, while mAP@0.5 remains stable. This achieves model lightweight without compromising mAP@0.5. This outcome can be primarily attributed to the incorporation of the GSConv and VoVGSCSP module, which utilizes depthwise separable convolution to significantly reduce the number of computed parameters. Additionally, it reshuffles the connections between channels to ensure information multiplexing, thereby maintaining detection accuracy. The deliberate decision to integrate the GSConv module into the neck was made with careful consideration. However, it was intentionally omitted from the backbone to prevent an excessive presence of GSConv modules. This choice aimed to avoid over-complicating the network architecture, which could hinder the flow of spatial information and substantially increase inference times.

(3) Effects of MPDIoU: A comparative analysis between YOLOv8 and experiments gradually introducing the MPDIoU module highlights the efficacy of its integration. The addition of MPDIoU notably enhances model accuracy, achieving a mAP@0.5 of up to 90.7% and showing improvements of 2.10%, with no additional parameters and speed costs. It also achieves high accuracy when integrated with other improvements. This substantiates that MPDIoU indeed contributes to improved model performance by calculating the IoU based on minimizing the point distance between the predicted bounding box and the ground truth bounding box.

(4) Effects of PCDetect: A comparative analysis between YOLOv8 and experiments involving the gradual addition of the PCDetect module highlights its effectiveness. The incorporation of PCDetect contributes to a reduction in computational costs, with FLOPS and Params experiencing reductions of 31.71% and 20.00%, respectively. It maintains accuracy while achieving these reductions when integrated with other improvements.

(5) Effects of integrating together: CDDLite-YOLO seamlessly combines the strengths of C2f-Faster, Slim-neck, MPDIoU, and PCDetect. The result is a model with a 56.10% reduction in parameters, a 40.00% decrease in computational demand, and a noteworthy 2.00% improvement in mAP@0.5 compared to YOLOv8.

The CDDLite-YOLO model significantly reduces both model size and computational costs while maintaining a comparable detection accuracy. This emphasizes a harmonious balance between enhancing accuracy and streamlining model efficiency, underscoring the significance of our proposed improvements.

3.2.2 Performance comparison with the state-of-the-art detection models

To evaluate the model's effectiveness, we conducted comparative experiments, comparing our proposed model against well-known lightweight models such as YOLOv5n, YOLOv6n, YOLOv7-tiny, and YOLOv8n. All experiments utilized the same cotton diseases dataset, which consists of 1224 training images, 153 validation images, and 153 test images. We maintained identical experimental conditions throughout to ensure a fair comparison.

The comparison results are shown in Figure 4 and Table 3.

CDDLite-YOLO outperforms other mainstream lightweight models in terms of detection accuracy. In this paper, CDDLite-YOLO achieves mAP@0.5 and mAP@0.5:0.95 scores of 90.6% and 73.7%, surpassing the performance of YOLOv5n, YOLOv6n, YOLOv7-tiny, Faster R-CNN, SSD, RetinaNet, FCOS and YOLOv8n. Several factors contribute to this superior performance. Firstly, the C2f-Faster module utilizes only 1/4 of the input channels for convolution and processing 3/4 of the channels extracted from these channels in subsequent 1×1 convolutions. This approach enhances spatial feature extraction by reducing redundant computation and memory access simultaneously. Secondly, Slim-neck utilizes the shuffle operation to seamlessly integrate information from SC into DSC-generated data while preserving hidden connections. This approach effectively achieves a balanced trade-off between model accuracy and speed, keeping complexity low. Additionally, the PCDetect module employs a 1×1 convolutional kernel for channel adjustment, enhancing feature fusion and cross-channel perception without substantially increasing parameters. The integration of the C2f-Faster module, Slim-neck, and PCDetect module significantly reduces operational parameters while maintaining inference speed, without compromising detection accuracy. Furthermore, the inclusion of MPDIoU is pivotal in enhancing model accuracy. It addresses limitations in existing loss functions by considering the minimum point distance between predicted and ground truth bounding boxes, particularly when they share the same aspect ratio but possess varying width and height values. These factors collectively enhance the effectiveness of the CDDLite-YOLO model in detecting cotton diseases.

The CDDLite-YOLO model excels in reducing parameter count and computational complexity. Compared to YOLOv5n, YOLOv6n, YOLOv7-tiny, Faster R-CNN, SSD, RetinaNet, FCOS and YOLOv8n, our proposed CDDLite-YOLO model offers lower FLOPS and Params, specifically 3.6G and 1.8M. This reduction can be mainly attributed to the incorporation of the C2f-Faster module, Slim-neck, and PCDetect module.

Upon analyzing the results, we observe that the Params of the YOLOv5n model are slightly lower than those of our proposed model, albeit by only 0.1. However, what sets CDDLite-YOLO apart is its superior performance in terms of Precision, Recall, mAP@0.5, mAP@0.5:0.95, and speed. The CDDLite-YOLO model outperforms YOLOv5n with a 0.5% increase in Precision, 4.9% in Recall, 3.1% in mAP@0.5, 7.1% in mAP@0.5:0.95, and a remarkable 107.28 FPS boost in speed.

The results unequivocally establish the superiority of our proposed model over the current mainstream lightweight algorithms in three key aspects: model size, detection accuracy, and detection speed. To further substantiate the performance of the CDDLite-YOLO model, we randomly selected detection results from a variety of environmental conditions among all testing samples, as displayed in Figure 5.

3.2.3 Performance comparison of loss function

We experimented with various IoU loss functions to determine their impact on performance. The tested loss functions include CIoU loss, GIoU loss (Rezatofighi et al., 2019), SIoU loss (Gevorgyan, 2022), WIoU loss (Cho, 2021), and MPDIoU loss, while the remaining aspects of the YOLOv8 model were kept constant. The comparative results are presented in the Table 4.

Notably, when using MPDIoU as the loss function for YOLOv8, the highest mAP is achieved. This can be attributed to its adaptability to diseases of various shapes and sizes in field environments, distinguishing it as the most suitable choice for our model in comparison with the other tested loss functions, particularly when compared to the original IoU loss.

3.2.4 Performance comparison of detection head optimization

To evaluate the impact of the PCDetect detection head on cotton disease detection, we conducted experiments to determine the most effective detection head. We tested several detection heads, including Origin YOLOv8 (featuring two 3x3 Conv layers), a detection head with one 1x1 ScConv (Li et al., 2023a) + one 1x1 Conv, a detection head with two 3x3 RepConv (Soudy et al., 2022), and PCDetect (comprising 1x1 PConv + one 1x1 Conv). The

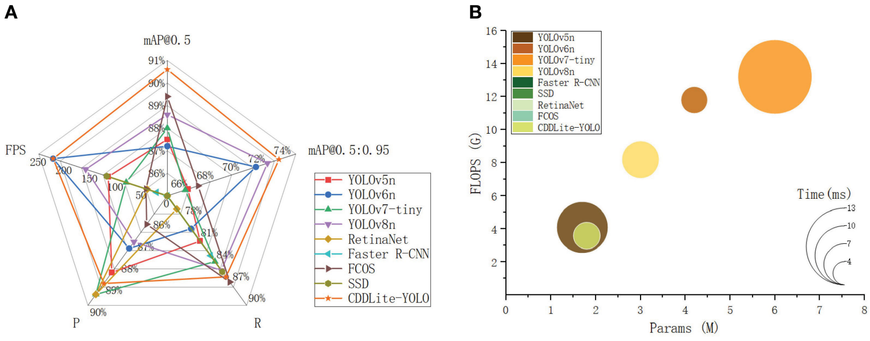


FIGURE 4
Comparison of detection results between different models. **(A)** Detection performance. **(B)** Computational complexity, parameter, and detection time.

TABLE 3 Comparison of detection performance of different models.

Models	Precision	Recall	mAP@0.5	mAP@0.5:0.95	FLOPS/G	Params/M	Speed
YOLOv5n	88.5%	81.2%	87.5%	66.6%	4.1	1.7	114.9
YOLOv6n	87.4%	79.5%	87.2%	71.9%	11.8	4.2	220.3
YOLOv7-tiny	89.5%	84.0%	88.0%	66.4%	13.2	6.0	80.0
YOLOv8n	87.1%	85.2%	88.6%	72.8%	8.2	3.0	158.7
Faster R-CNN	43.9%	83.2%	74.3%	46.3%	370.2	137.1	21.0
SSD	74.4%	85.4%	82.1%	58.4%	62.8	26.3	117.6
RetinaNet	89.5%	76.8%	84.4%	60.8%	170.1	38.0	39.3
FCOS	86.3%	86.8%	89.4%	67.4%	161.9	32.2	41.5
CDDLite-YOLO	89.0%	86.1%	90.6%	73.7%	3.6	1.8	222.2

Comparison of different detection heads on YOLOv8 is shown in Table 5.

Comparing PCDetect with Origin YOLOv8 and the detection head with one 1x1 ScConv + one 1x1 Conv, we observed that the mAP@0.5 of the PCDetect detection head remained stable. However, the number of parameters decreased by 20% and 4%, while computational complexity increased by 31.71% and 1.75%. It's worth noting that although the mAP@0.5 of the detection head with two 3x3 RepConv was 0.6 higher than that of PCDetect, the computational costs and parameter count increased by 44.64% and 58.33% compared to PCDetect, even surpassing those of the Origin YOLOv8 model.

Our experimental results unequivocally confirm that using the PCDetect detection head outperforms other options, maintaining detection accuracy while requiring fewer parameters and lower computational complexity.

4 Discussion

4.1 The importance of model lightweight

In recent years, advances in deep learning and convolutional networks have significantly enhanced object detection capabilities. Embedded computing devices have emerged as the preferred computational core for cost-effective and portable agricultural equipment. However, a graphics card's performance depends on its single-precision floating-point capabilities, CUDA core count, and overall computing power, creating a noticeable power gap between embedded devices and professional computing cards (Cui et al., 2023). Consider the NVIDIA H100, a pinnacle in professional computing, with an impressive 1200.00 TFlops in single-precision floating-point performance and a substantial 18432 CUDA cores. Meanwhile, the NVIDIA A100, another powerhouse in professional computing, maintains a balanced profile with 312.00 TFlops and 6912 CUDA cores. On the other hand, the NVIDIA GeForce RTX 4090, a robust GPU not specifically tailored for professional computing, emphasizes a different performance profile with 82.58 TFlops and 16384 CUDA cores. In contrast, embedded devices like the NVIDIA Jetson AGX Orin and Jetson TX2, efficient in their own right, demonstrate more modest capabilities with 5.30 TFlops/2560 CUDA cores and 1.36 TFlops/256 CUDA cores, respectively.

Deep learning models demand a considerable number of multiplicative operations for accurate feature extraction. Deploying detection models on embedded devices presents a significant challenge due to their constrained computational resources. Unfortunately, the computing power of the NVIDIA Jetson TX2 is only 1/882nd of that of the NVIDIA H100, highlighting the embedded devices' inability to handle such demanding calculations within a reasonable timeframe.

In the context of deployment on agricultural inspection robots and resource-constrained devices, while some detection networks boast high accuracy, their extensive parameters and computations strain devices. Conversely, the most lightweight detection models offer faster detection but often sacrifice accuracy, posing challenges for application. Thus, ensuring the model lightweight while maintaining detection accuracy is a fundamental requirement for deploying the cotton disease detection model on agricultural inspection robots and other resource-constrained devices. The CDDLite-YOLO model adeptly amalgamates the strengths of various lightweight modules such as C2f-Faster, Slim-neck, and PCDetect. By doing so, it achieves a harmonious balance between enhancing accuracy and streamlining model efficiency, rendering it well-suited for deployment on agricultural inspection robots and other resource-constrained agricultural devices.

4.2 Discussions of the detection results

Extensive research has been conducted on detecting cotton diseases using deep learning. However, previous studies, such as those by (Priya et al., 2021; Devi Priya et al., 2022; Susa et al., 2022; Zhang et al., 2022; Zhang et al., 2023b, Zhang et al., 2023c), did not fully consider the requirement for fast detection in applications involving agricultural inspection robots or detection conducted within controlled environments. This study addresses these specific needs.

The advantages of the CDDLite-YOLO model are as follows:

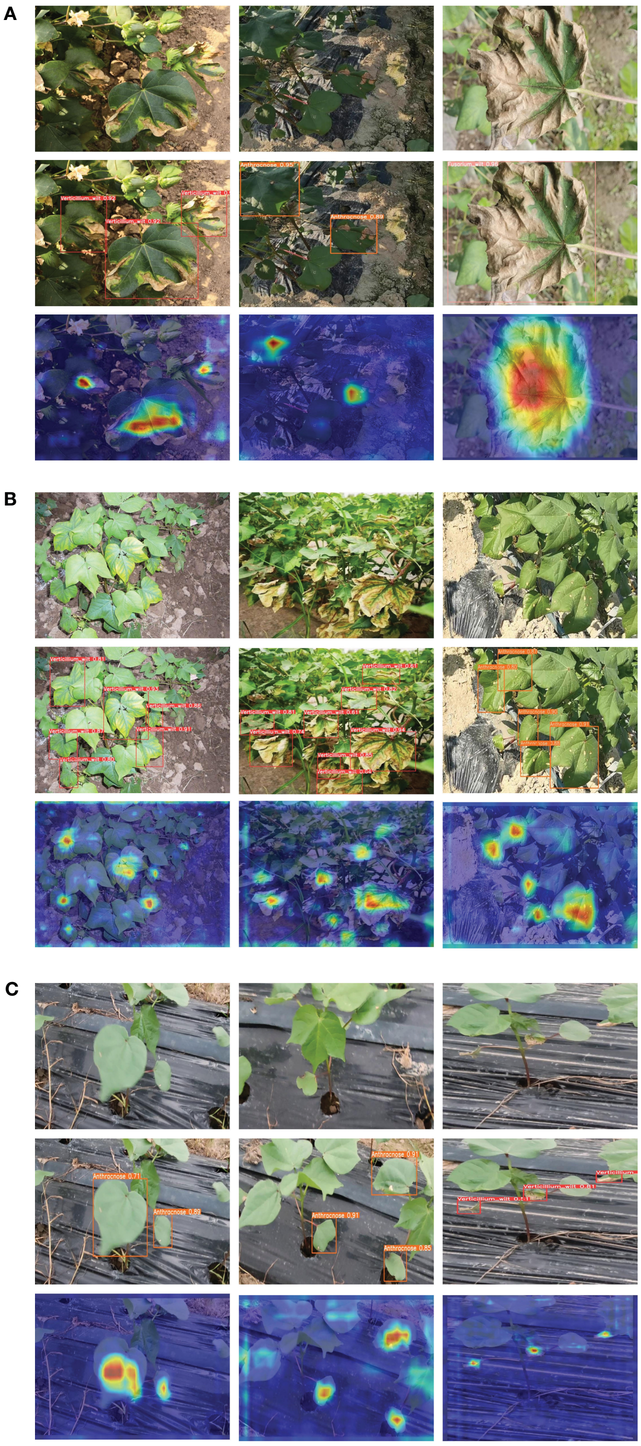


FIGURE 5
Prediction results of the proposed method. **(A)** Under complex backgrounds such as plastic film, water pipes, and soil in the field. **(B)** Under dense disease conditions. **(C)** Under the conditions of image blurriness generated during the agricultural inspection robot movement and collection process.

TABLE 4 Comparison of different loss functions onYOLOv8.

Loss Functions	CIoU (Origin YOLOv8)	GIoU	SIoU	WIoU	MPDIoU
mAP@0.5	88.6%	89.3%	90.1%	90.0%	90.7%

TABLE 5 Comparison of different detection heads onYOLOv8.

Detection head	Origin YOLOv8	one 1x1 ScConv + one 1x1 Conv	two 3x3 RepConv	PCDetect
mAP@0.5	88.6%	88.7%	89.6%	89.0%
FLOPS/G	8.2	5.7	8.1	5.6
Params/M	3.0	2.5	3.8	2.4

(1) Lightweight and Speed: The CDDLite-YOLO model exhibits lightweight characteristics and reduces model size, making it well-suited for deployment on agricultural inspection robots and other resource-constrained agricultural devices.

(2) Balance of Accuracy and Efficiency: The CDDLite-YOLO model strikes a harmonious balance between detection speed, accuracy, and model size, positioning it as a promising candidate for deployment on an embedded GPU chip without compromising performance.

4.3 Limitations and future prospects

While our proposed method has demonstrated encouraging results, there are still certain limitations that need to be addressed in future research.

The mAP@0.5 of the CDDLite-YOLO model for detecting cotton verticillium wilt diseases currently stands at 78.1%, leaving room for improvement. This lower accuracy may be attributed to factors such as background interference, as the color of cotton verticillium wilt diseases closely resembles that of the soil, making them easily blend into the background. Additionally, cotton verticillium wilt diseases and cotton Fusarium wilt diseases share a similar color, leading to occasional misdetections. To address these limitations, future experiments will explore the use of spectral imaging or hyperspectral imaging to capture more detailed information about the spectral characteristics of cotton verticillium wilt diseases. This can aid in distinguishing them from the soil background. Moreover, we will enrich our dataset by gathering and analyzing images of cotton diseases from various varieties and regions captured by agricultural inspection robots during their operation. This initiative will further validate the applicability of the model proposed in this study. Furthermore, we intend to implement systems that integrate human expertise to validate and refine model predictions, thus strengthening the accuracy of disease detection.

Regarding model deployment, we have successfully deployed the CDDLite-YOLO model on embedded devices such as the NVIDIA Jetson AGX Orin, NVIDIA Jetson TX2, and NVIDIA Jetson Nano. It performs well and fulfills the requirements for low computational power embedded devices in detecting cotton diseases in natural field environments. It achieves a balance between detection speed, accuracy, and model size, allowing deployment on these embedded GPU chips without sacrificing performance. Additionally, the CDDLite-YOLO model has been applied on agricultural inspection robots equipped with NVIDIA Jetson AGX Orin, demonstrating excellent performance in rapidly inspecting. We hope to deploy it on more cost-effective agricultural inspection robots in the future. However, our lab currently lacks access to agricultural inspection robots which are equipped with more cost-effective devices like NVIDIA Jetson Nano, which will be the focus of our future research.

Despite its limitations, CDDLite-YOLO serves as a valuable technical reference for detecting cotton diseases in natural field conditions. The application of the CDDLite-YOLO model in agricultural inspection robots for cotton disease detection holds the promise of validating its reliability.

5 Conclusions

Cotton, a crucial global source of natural textile fibers, is highly susceptible to cotton diseases, which significantly impact both cotton quality and yield. The use of deep learning has become an integral approach to cotton disease detection. However, current detection models often suffer from an overabundance of model parameters, making them unsuitable for resource-constrained devices and hindering the delicate balance between detection accuracy and speed. To address these challenges, our research establishes a dedicated dataset for cotton disease detection. Building upon the YOLOv8 model, we introduce significant improvements, resulting in the CDDLite-YOLO model that meets the demands for accuracy, lightweight design, and real-time performance in agricultural inspection robots and resource-constrained agricultural devices. These enhancements encompass the introduction of the C2f-Faster module, Slim-neck structure, the PCDetect detection head, and the MPDIoU loss function. These innovations enable automatic cotton disease detection in natural environments, even on resource-constrained agricultural devices. Our experimental results validate the model's effectiveness, achieving an impressive mAP@0.5 of 90.6%. It outperforms comparable models in mAP@50–95, precision, and recall. The model excels in computational efficiency, with parameters totaling 1.8M, FLOPS at 3.6G, and a rapid detection speed of 222.22ms. These advancements represent a significant leap compared to mainstream lightweight detection models like YOLOv5n, YOLOv6n, YOLOv7-tiny, and YOLOv8n, rendering them highly suitable for deployment on agricultural inspection robots. This study provides innovative methods for developing lightweight cotton disease detection models and deploying them on agricultural inspection robots and other resource-constrained agricultural devices. Additionally, it is also a reference for crop loss estimation, pesticidal management practices, and understanding symptom-environment relationships. the CDDLite-YOLO model for detecting cotton verticillium wilt indicates room for improvement. This limitation could potentially be addressed by exploring the use of spectral imaging or hyperspectral imaging to capture more detailed information about the spectral characteristics of cotton verticillium wilt diseases.

Data availability statement

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

Author contributions

PP: Conceptualization, Methodology, Writing – original draft, Writing – review & editing. MS: Data curation, Software, Writing – review & editing. PH: Data curation, Writing – review & editing. LiH: Funding acquisition, Supervision, Writing – review & editing. SZ: Validation, Writing – review & editing. LoH: Formal analysis, Writing – review & editing. GZ: Funding acquisition, Project administration, Writing – review & editing. JZ: Funding acquisition, Project administration, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the National Key Research and Development Program of China (No.2022YFF0711805, 2021YFF0704204, 2022YFF0711801), and the Project of Sanya Yazhou Bay Science and Technology City (No.SCKJ-JYRC-2023-45), and the Innovation Project of the Chinese Academy of Agricultural Sciences (No.CAAS-ASTIP-2024-AII, No.CAAS-ASTIP-2023-AII), and The Special Fund of Chinese Central Government for Basic Scientific Research Operations in Commonweal Research Institutes (No.JBYW-AII-2024-05, No.JBYW-AII-2023-06, No.Y2022XK24, No.Y2022QC17, No.JBYW-AII-2022-14), and Nanfan special project, CAAS Grant Nos. YBXM2409, YBXM2410, YBXM2312, ZDXM2311, YBXM2312, YDLH01, YDLH07, YBXM10 and National Natural Science Foundation of China (No.31971792, No.32160421).

Acknowledgments

Appreciations are given to the editors and reviewers of the Journal Frontiers in Plant Science.

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

Balaska, V., Adamidou, Z., Vryzas, Z., and Gasteratos, A. (2023). Sustainable crop protection via robotics and artificial intelligence solutions. *Machines* 11, 774. doi: 10.3390/machines11080774

Cai, Y., He, X., Mo, J., Sun, Q., Yang, J., and Liu, J. (2009). Molecular research and genetic engineering of resistance to Verticillium wilt in cotton: a review. *Afr. J. Biotechnol.* 8, 7363–7372.

Chen, J., Kao, S.-H., He, H., Zhuo, W., Wen, S., Lee, C.-H., et al. (2023). "Run, don't walk: chasing higher FLOPS for faster neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. (IEEE: Vancouver, BC, Canada), 12021–12031. doi: 10.1109/CVPR52729.2023.01157

Chi, B., Zhang, D., and Dong, H. (2021). Control of cotton pests and diseases by intercropping: A review. *J. Integr. Agric.* 20, 3089–3100. doi: 10.1016/S2095-3119(20)63318-4

Cho, Y.-J. (2021). Weighted intersection over union (wIoU): A new evaluation metric for image segmentation. *arXiv e-prints arXiv:2107.09858*. doi: 10.48550/arXiv.2107.09858

Chohan, S., Perveen, R., Abid, M., Tahir, M. N., and Sajid, M. (2020). "Cotton diseases and their management," in *Cotton Production and Uses: Agronomy, Crop Protection, and Postharvest Technologies*. Eds. S. Ahmad and M. Hasanuzzaman (Springer Singapore, Singapore), 239–270.

Cui, M., Lou, Y., Ge, Y., and Wang, K. (2023). LES-YOLO: A lightweight pinecone detection algorithm based on improved YOLOv4-Tiny network. *Comput. Electron. Agric.* 205, 107613. doi: 10.1016/j.compag.2023.107613

Dai, G., Hu, L., and Fan, J. (2022). DA-actNN-YOLOV5: hybrid YOLO v5 model with data augmentation and activation of compression mechanism for potato disease identification. *Comput. Intell. Neurosci.* 2022, 6114061. doi: 10.1155/2022/6114061

Devi Priya, R., Devisurya, V., Anitha, N., Dharani,, Geetha, B., and Kirithika, R. V. (2022). "Faster R-CNN with augmentation for efficient cotton leaf disease detection," in *Hybrid Intelligent Systems*. Eds. A. Abraham, P. Siarry, V. Piuri, N. Gandhi, G. Casalino, O. Castillo and P. Hung (Cham: Springer International Publishing), 140–148. doi: 10.1007/978-3-030-96305-7_13

Gevorgyan, Z. (2022). SIoU loss: more powerful learning for bounding box regression. *arXiv e-prints arXiv:2205.12740*. doi: 10.48550/arXiv.2205.12740

Guo, Y., Lan, Y., and Chen, X. (2022). CST: Convolutional Swin Transformer for detecting the degree and types of plant diseases. *Comput. Electron. Agric.* 202, 107407. doi: 10.1016/j.compag.2022.107407

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

Huang, G., Huang, J.-Q., Chen, X.-Y., and Zhu, Y.-X. (2021). Recent advances and future perspectives in cotton research. *Annu. Rev. Plant Biol.* 72, 437–462. doi: 10.1146/annurev-arplant-080720-113241

Jia, L., Wang, T., Chen, Y., Zang, Y., Li, X., Shi, H., et al. (2023). MobileNet-CA-YOLO: an improved YOLOv7 based on the mobileNetV3 and attention mechanism for rice pests and diseases detection. *Agriculture* 13(7). doi: 10.3390/agriculture13071285

Jiang, H., Hu, F., Fu, X., Chen, C., Wang, C., Tian, L., et al. (2023). YOLOv8-Peas: a lightweight drought tolerance method for peas based on seed germination vigor. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1257947

Johnson, J., Sharma, G., Srinivasan, S., Masakapalli, S. K., Sharma, S., Sharma, J., et al. (2021). Enhanced field-based detection of potato blight in complex backgrounds using deep learning. *Plant Phenomics* 2021. doi: 10.34133/2021/9835724

Kang, C. H., and Kim, S. Y. (2023). Real-time object detection and segmentation technology: an analysis of the YOLO algorithm. *JMST Adv.* 5, 69–76. doi: 10.1007/s42791-023-00049-7

Lawrence, K., Strayer-Scherer, A., Norton, R., Hu, J., Faske, T., Hutmacher, R., et al. (2022). "Cotton disease loss estimate committee repor," in *2022 Beltwide Cotton Conferences* (National Cotton Council, San Antonio, TX).

Leng, S., Musha, Y., Yang, Y., and Feng, G. (2023). CEMLB-YOLO: efficient detection model of maize leaf blight in complex field environments. *Appl. Sci.* 13(16). doi: 10.3390/app13169285

Li, H., Li, J., Wei, H., Liu, Z., Zhan, Z., and Ren, Q. (2022). Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv e-prints* 2206, 2424. doi: 10.48550/arXiv.2206.02424

Li, J., Wen, Y., and He, L. (2023a). "SCConv: spatial and channel reconstruction convolution for feature redundancy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. (IEEE: Vancouver, BC, Canada), 6153–6162. doi: 10.1109/CVPR52729.2023.00596

Li, K., Zhu, X., Qiao, C., Zhang, L., Gao, W., and Wang, Y. (2023b). The gray mold spore detection of cucumber based on microscopic image and deep learning. *Plant Phenomics* 5, 11. doi: 10.34133/plantphenomics.0011

Liu, J., and Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods* 17, 22. doi: 10.1186/s13007-021-00722-9

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7. doi: 10.3389/fpls.2016.01419

Nawaz, H. H., Nelly Rajaofera, M. J., He, Q., Anam, U., Lin, C., and Miao, W. (2018). Evaluation of antifungal metabolites activity from bacillus licheniformis OE-04 against Colletotrichum gossypii. *Pesticide Biochem. Physiol.* 146, 33–42. doi: 10.1016/j.pestbp.2018.02.007

Pan, P., Guo, W., Zheng, X., Hu, L., Zhou, G., and Zhang, J. (2023a). Xoo-YOLO: a detection method for wild rice bacterial blight in the field from the perspective of unmanned aerial vehicles. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1256545

Pan, P., Zhang, J., Zheng, X., Zhou, G., Hu, L., Feng, Q., et al. (2023b). Research progress of deep learning in intelligent identification of disease resistance of crops and their related species. *Acta Agriculturae Zhejiangensis* 35, 1993–2012. doi: 10.3969/j.issn.1004-1524.20236105

Peyal, H. I., Pramanik, M. A. H., Nahiduzzaman, M., Goswami, P., Mahapatra, U., and Atusi, J. J. (2022). "Cotton leaf disease detection and classification using lightweight CNN architecture," in *2022 12th International Conference on Electrical and Computer Engineering (ICECE)*. (IEEE: Dhaka, Bangladesh), 413–416. doi: 10.1109/ICECE57408.2022.10088570

Priya, D., Devisurya,, Dharani,, Geetha,, and Kiruthika, (2021). Cotton leaf disease detection using Faster R-CNN with Region Proposal Network. *Int. J. Biol. Biomedicine* 6, 23–35.

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (IEEE: Vancouver, BC, Canada), 658–666. doi: 10.1109/ICECE57408.2022.10088570

Saleem, M. H., Potgieter, J., and Arif, K. M. (2021). Automation in agriculture by machine and deep learning techniques: A review of recent developments. *Precis. Agric.* 22, 2053–2091. doi: 10.1007/s11119-021-09806-x

Siliang, M., and Yong, X. (2023). MPDIoU: A loss for efficient and accurate bounding box regression. *arXiv e-prints 2307.07662*. doi: 10.48550/arXiv.2307.07662

Soudy, M., Afify, Y., and Badr, N. (2022). RepConv: A novel architecture for image scene classification on Intel scenes dataset. *Int. J. Intelligent Computing Inf. Sci.* 22, 63–73. doi: 10.21608/ijicis.2022.118834.1163

Susa, J. A. B., Nombrefia, W. C., Abustan, A. S., Macalisang, J., and Maaliw, R. R. (2022). "Deep learning technique detection for cotton and leaf classification using the YOLO algorithm," in *2022 International Conference on Smart Information Systems and Technologies (SIST)*. (IEEE: Nur-Sultan, Kazakhstan), 1–6. doi: 10.1109/SIST54437.2022.994575710.1109/SIST54437.2022.9945757

Tang, Y., Qiu, J., Zhang, Y., Wu, D., Cao, Y., Zhao, K., et al. (2023a). Optimization strategies of fruit detection to overcome the challenge of unstructured background in field orchard environment: a review. *Precis. Agric.* 24, 1183–1219. doi: 10.1007/s11119-023-10009-9

Tang, Z., He, X., Zhou, G., Chen, A., Wang, Y., Li, L., et al. (2023b). A precise image-based tomato leaf disease detection approach using PLPNet. *Plant Phenomics* 5, 42. doi: 10.34133/plantphenomics.0042

Terven, J., and Cordova-Esparza, D. (2023). A comprehensive review of YOLO: from YOLOv1 to YOLOv8 and beyond. *arXiv e-prints 2304.00501*. doi: 10.48550/arXiv.2304.00501

Thakur, P. S., Khanna, P., Sheorey, T., and Ojha, A. (2022). Trends in vision-based machine learning techniques for plant disease identification: A systematic review. *Expert Syst. Appl.* 208, 118117. doi: 10.1016/j.eswa.2022.118117

Tian, Y., Su, D., Lauria, S., and Liu, X. (2022). Recent advances on loss functions in deep learning for computer vision. *Neurocomputing* 497, 129–158. doi: 10.1016/j.neucom.2022.04.127

Toscano-Miranda, R., Toro, M., Aguilar, J., Caro, M., Marulanda, A., and Trebilcok, A. (2022). Artificial-intelligence and sensing techniques for the management of insect pests and diseases in cotton: a systematic literature review. *J. Agric. Sci.* 160, 16–31. doi: 10.1017/S002185962200017X

Townsend, T. (2020). "1B - World natural fibre production and employment," in *Handbook of Natural Fibres, 2nd ed.* Eds. R. M. Kozłowski and M. Mackiewicz-Talarczyk (Houston: Woodhead Publishing), 15–36.

Wang, H., Lin, Y., Xu, X., Chen, Z., Wu, Z. , and Tang, Y. (2022a). A study on long-close distance coordination control strategy for litchi picking. *Agronomy* 12, 1520. doi: 10.3390/agronomy12071520

Wang, H., Siddiqui, M. Q., and Memon, H. (2020). "Physical structure, properties and quality of cotton," in *Cotton Science and Processing Technology: Gene, Ginning, Garment and Green Recycling*. Eds. H. Wang and H. Memon (Springer Singapore, Singapore), 79–97.

Wang, P., Su, L., Qin, L., Hu, B., Guo, W., and Zhang, T. (2009). Identification and molecular mapping of a Fusarium wilt resistant gene in upland cotton. *Theor. Appl. Genet.* 119, 733–739. doi: 10.1007/s00122-009-1084-4

Wang, Y., Wang, Y., and Zhao, J. (2022b). MGA-YOLO: A lightweight one-stage network for apple leaf disease detection. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.927424

Wani, J. A., Sharma, S., Muzamil, M., Ahmed, S., Sharma, S., and Singh, S. (2022). Machine learning and deep learning based computational techniques in automatic agricultural diseases detection: methodologies, applications, and challenges. *Arch. Comput. Methods Eng.* 29, 641–677. doi: 10.1007/s11831-021-09588-5

Xiao, B., Nguyen, M., and Yan, W. Q. (2023). Fruit ripeness identification using YOLOv8 model. *Multimedia Tools Applications* 83 (9), 28039–28056. doi: 10.1007/s11042-023-16570-9

Xu, C., Wang, Z., Du, R., Li, Y., Li, D., Chen, Y., et al. (2023). A method for detecting uneaten feed based on improved YOLOv5. *Comput. Electron. Agric.* 212, 108101. doi: 10.1016/j.compag.2023.108101

Yao, Z., Douglas, W., O'Keeffe, S., and Villing, R. (2022). "Faster YOLO-LITE: faster object detection on robot and edge devices," in *RoboCup 2021: Robot World Cup XXIV*. Eds. R. Alami, J. Biswas, M. Cakmak and O. Obst (Cham:Springer International Publishing), 226–237.

Ye, L., Wu, F., Zou, X., and Li, J. (2023). Path planning for mobile robots in unstructured orchard environments: An improved kinematically constrained bi-directional RRT approach. *Comput. Electron. Agric.* 215, 108453. doi: 10.1016/j.compag.2023.108453

Zaidi, S.-e., Naqvi, R. Z., Asif, M., Strickler, S., Shakir, S., Shafiq, M., et al. (2020). Molecular insight into cotton leaf curl geminivirus disease resistance in cultivated cotton (Gossypium hirsutum). *Plant Biotechnol. J.* 18, 691–706. doi: 10.1111/pbi.13236

Zhang, D.-Y., Luo, H.-S., Cheng, T., Li, W.-F., Zhou, X.-G., Wei, G., et al. (2023a). Enhancing wheat Fusarium head blight detection using rotation Yolo wheat detection network and simple spatial attention network. *Comput. Electron. Agric.* 211, 107968. doi: 10.1016/j.compag.2023.107968

Zhang, J., Yang, J., Zhao, J., Zhang, X., Yi, J., and Li, Z. (2023b). Improved YOLOv5-based algorithm for cotton wilt disease identification. *Comput. Knowledge Technol.* 19, 51–53+56. doi: 10.14004/j.cnki.ckt.2023.1018

Zhang, K., Wu, Q., and Chen, Y. (2021). Detecting soybean leaf disease from synthetic image using multi-feature fusion faster R-CNN. *Comput. Electron. Agric.* 183, 106064. doi: 10.1016/j.compag.2021.106064

Zhang, N., Zhang, X., Bai, T., Shang, P., Wang, W., and Li, L. (2023c). Identification method of cotton leaf pests and diseases in natural environment based on CBAM-YOLO v7. *Trans. Chin. Soc. Agric. Machinery* 54 (S1), 239–244.

Zhang, Y., Ma, B., Hu, Y., Li, C., and Li, Y. (2022). Accurate cotton diseases and pests detection in complex background based on an improved YOLOX model. *Comput. Electron. Agric.* 203, 107484. doi: 10.1016/j.compag.2022.107484

Zhang, Y., Zhou, G., Chen, A., He, M., Li, J., and Hu, Y. (2023e). A precise apple leaf diseases detection using BCTNet under unconstrained environments. *Comput. Electron. Agric.* 212, 108132. doi: 10.1016/j.compag.2023.108132

Zhang, X., Mang, X., Du, J., Ma, L., Huang, Z., Wang, X., et al. (2023d). "Bird intrusion detection method for transmission lines based on YOLOv5-SBM," in *2023 4th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)*. (Nanjing, China: IEEE), 395–398. doi: 10.1109/ICBASE59196.2023.10303238

Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Salt Lake City, UT, USA: IEEE), 6848–6856. doi: 10.1109/CVPR.2018.00716

Zhao, S., Liu, J., and Wu, S. (2022). Multiple disease detection method for greenhouse-cultivated strawberry based on multiscale feature fusion Faster R_CNN. *Comput. Electron. Agric.* 199, 107176. doi: 10.1016/j.compag.2022.107176

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 20 June 2024
DOI 10.3389/fpls.2024.1381367

Check for updates

OPEN ACCESS

EDITED BY
Yongliang Qiao,
University of Adelaide, Australia

REVIEWED BY
Preeta Sharan,
The Oxford College of Engineering, India
Ting Yun,
Nanjing Forestry University, China

*CORRESPONDENCE
Weibin Wu
wuweibin@scau.edu.cn
Si Sun
sunsi@scau.edu.cn

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

RECEIVED 03 February 2024
ACCEPTED 29 May 2024
PUBLISHED 20 June 2024

CITATION
Zhang Z, Han C, Wang X, Li H, Li J, Zeng J, Sun S and Wu W (2024) Large field-of-view pine wilt disease tree detection based on improved YOLO v4 model with UAV images. *Front. Plant Sci.* 15:1381367. doi: 10.3389/fpls.2024.1381367

Large field-of-view pine wilt disease tree detection based on improved YOLO v4 model with UAV images

Zhenbang Zhang[1,2,3†], Chongyang Han[1†], Xinrong Wang[4], Haoxin Li[1], Jie Li[5], Jinbin Zeng[1], Si Sun[6*] and Weibin Wu[1*]

[1]College of Engineering, South China Agricultural University, Guangzhou, China, [2]Guangdong Provincial Key Laboratory of Utilization and Conservation of Food and Medicinal Resources in Northern Region, Shaoguan University, Shaoguan, China, [3]College of Intelligent Engineering, Shaoguan University, Shaoguan, China, [4]College of Plant Protection, South China Agricultural University, Guangzhou, China, [5]College of Artificial Intelligence, Nankai University, Tianjin, China, [6]College of Forestry and Landscape Architecture, South China Agricultural University, Guangzhou, China

Introduction: Pine wilt disease spreads rapidly, leading to the death of a large number of pine trees. Exploring the corresponding prevention and control measures for different stages of pine wilt disease is of great significance for its prevention and control.

Methods: To address the issue of rapid detection of pine wilt in a large field of view, we used a drone to collect multiple sets of diseased tree samples at different times of the year, which made the model trained by deep learning more generalizable. This research improved the YOLO v4(You Only Look Once version 4) network for detecting pine wilt disease, and the channel attention mechanism module was used to improve the learning ability of the neural network.

Results: The ablation experiment found that adding the attention mechanism SENet module combined with the self-designed feature enhancement module based on the feature pyramid had the best improvement effect, and the mAP of the improved model was 79.91%.

Discussion: Comparing the improved YOLO v4 model with SSD, Faster RCNN, YOLO v3, and YOLO v5, it was found that the mAP of the improved YOLO v4 model was significantly higher than the other four models, which provided an efficient solution for intelligent diagnosis of pine wood nematode disease. The improved YOLO v4 model enables precise location and identification of pine wilt trees under changing light conditions. Deployment of the model on a UAV enables large-scale detection of pine wilt disease and helps to solve the challenges of rapid detection and prevention of pine wilt disease.

KEYWORDS

pine wilt disease, UAV images, large field-of-view, deep learning, target detection

1 Introduction

Pine wilt disease (PWD) is caused by pine wood nematode (PWN), which is known for its high destructiveness (Kobayashi et al., 2003). The disease has been widely distributed in Asia, especially in China, Japan, and South Korea, where it has caused the most damage (Kikuchi et al., 2011). The spread of PWD is swift. Once a diseased tree is found, nearby pine trees may also be infected (Asai and Futai, 2011). PWNS feed on and infest pine trees, causing the trees to weaken and die (Yun et al., 2012), resulting in losses to forestry production and the ecological environment. Countries have strengthened quarantine and control measures to cope with the spread of PWD. The spread of PWD poses a threat to Asia's forestry and ecological environment (Wu et al., 2020). Therefore, monitoring PWD is of great significance for the safety of China's forest resources (Schröder et al., 2010). The application of drone remote sensing technology has dramatically improved the efficiency of forest resource surveys (Kentsch et al., 2020). Traditional monitoring techniques rely on low-level semantic features extracted from remote sensing images, making them susceptible to factors such as noise, lighting, and seasons, which limits their application in complex real-world scenarios (Park et al., 2016). Using drones to aerially photograph areas affected by PWD, the location and degree of diseased trees can be visually observed from the aerial images, and targeted measures can be taken to deal with diseased trees, reducing the workload of manual investigations. It is of great significance to use drones combined with artificial intelligence algorithms to detect pine wilt disease, which significantly improves the detection efficiency of pine wilt disease.

With the rapid development of drone monitoring technology and image processing technology, drone remote sensing monitoring methods have gradually been applied in PWD monitoring (Syifa et al., 2020; Vicente et al., 2012). When drones are used to aerially photograph areas affected by pine wilt disease, visible light cameras are carried to obtain ground images within the scope of the PWD epidemic, and the images are transmitted to the display terminal for automatic identification and positioning of diseased trees by the trained target detection algorithm (Kuroda, 2010). The use of drones for automatic monitoring of PWD can improve the efficiency of diseased tree monitoring. Compared with satellite remote sensing monitoring, drone remote sensing monitoring has a lower cost and more straightforward operation. Applying this technology in PWD detection is beneficial to the protection of pine tree resources and the stability of the ecological environment (Gao et al., 2015; Tang and Shao, 2015).

In target detection, accurate feature extraction from images is a crucial issue affecting model performance. Traditional image target detection uses machine learning algorithms to extract image features. However, because machine learning algorithms can only extract shallow feature information from images, the performance of target detection is challenging to improve (Khan et al., 2021). Machine learning algorithms use manually designed feature operators to extract feature vectors of targets in the image, and based on these feature vectors, use statistical learning methods to achieve intelligent visual detection of image targets (Tian and Daigle, 2019). These algorithms rely on colors or specific shapes whose features are not stable enough, resulting in detection mode. Thus, the adaptability and robustness of the model to the environment are not good enough (Long et al., 2015). Therefore, deep learning algorithms have emerged (Li et al., 2023), and it has been successfully applied in fields such as computer vision, speech recognition, and medical image analysis. This algorithm uses convolutional neural networks to extract image features, which can extract deep-level feature information of image targets, thereby improving the detection accuracy of diseased trees (Lifkooee et al., 2018). The theoretical system of target detection algorithms has gradually improved as research in this subject has progressed, and many distinct method frameworks have been employed in many image detection fields (Zhang and Zhang, 2019). Li proposes a multi-block SSD method based on small object detection to the railway scene of UAV surveillance (Li et al., 2020). Xu extends the Faster RCNN vehicle detection framework from low-altitude drone images captured at signalized intersections (Xu et al., 2017). The focus of the research is how to change the structure of the algorithm model and achieve a balance between detection accuracy and processing time (Hosang et al., 2016).

Under changing lighting conditions, the texture features of the image change, resulting in a decrease in detection accuracy (Barnich and Van, 2011). There are relatively few algorithms for monitoring pine wilt diseased trees in the lighting change scene, and most of the target detection algorithms for diseased trees have complex structures, low detection accuracy, and low computational efficiency (Zhang et al., 2019). Huang et al. Constructed a densely connected convolutional networks (D-CNN) sample dataset, using GF-1 and GF-2 remote sensing images of areas with PWD. Then, the "microarchitecture combined with micromodules for joint tuning and improvement" strategy was used to improve the five popular model structures (Huang et al., 2022). In 2021, a spatiotemporal change detection method to improve accurate detections in tree-scale PWD monitoring was proposed by Zhang et al., which represents the capture of spectral, temporal, and spatial features (Zhang et al., 2021).

Currently, most of the detections for pine wilt are done by biological sampling, which is time-consuming and labor-intensive. Research on the detection of pine wilt disease using unmanned aerial vehicle (UAV) has mainly focused on stable light conditions, and little attention has been paid to the detection of pine wilt disease under changing light conditions, resulting in the low detection accuracy of the existing models, as well as the inability of their improved methods to detect disease spots under changing light conditions. And there is the problem of small field of view and small number of targets. The research object of this paper is PWD tree, by increasing the flying height of UAV, increasing the field of view range of the camera, increasing the number of image targets, and based on this, a set of algorithms for detecting and recognizing the targets of diseased tree is proposed, which provides theoretical and practical support for detecting and recognizing the targets of remote sensing images by UAV.

In conclusion, this paper proposes a YOLO v4 target recognition algorithm based on the Attention Mechanism Module to establish a model for rapid localization and accurate recognition of pine nematode disease trees under dynamic light changes. Further, combining it with UAV image technology realizes rapid multi-target detection over a large field of view. This can save time in investigating pine wood nematode disease and realize prevention in advance, which is of great significance for preventing the spread of pine wood nematode disease.

2 Experimental parameters and YOLO v4 network structure

2.1 Sample collection sites and UAV images acquisition

The prominent peak of Yunji Mountain has an elevation of 1434.2 meters and is located at 24°07’ north latitude and 114°08’ east longitude (Figure 1A). It is located in the north of Guangzhou City, in the central part of Xinfeng County, 10 kilometers away from the county town. It belongs to the natural ecosystem transitional zone from the South sub-tropical zone to the Central subtropical zone, with a jurisdictional area of 2700 hectares. The panoramic image collected by the drone was taken in multiple shots and stitched together to form a complete image. The collection area includes a winding road and houses distributed along the roadside. The mountain is higher in the northeast and lower in the southwest as shown in Figure 1B.

The visible light images were acquired using the DJI Mavic 2 drone, equipped with ten sensors distributed in six directions: front, rear, left, right, up, and down. The sensor model is 1-inch Complementary Metal Oxide Semic (CMOS), and the captured image resolution is 5472×3684. The drone can reduce air resistance by 19% during high-speed flight, and its maximum flight speed can reach 72 km/h, with a flight time of up to 31 minutes, the experimental drone is shown in Figure 1C.

The illumination can affect the clarity of the drone remote sensing image collection. Due to the continuously changing natural lighting conditions over time and weather, the lighting conditions greatly affect the image quality, resulting in complex information in the collected images of diseased trees. According to the lighting conditions of the photos, they can be divided into two categories: sufficient light and insufficient light. The light intensity was measured by an illuminometer.

To balance the image quality and the diseased tree target detection network, all remote sensing images of diseased trees are uniformly resized to a resolution of 416×416 pixels. The uneven lighting caused by changes in the lighting conditions affects the quality of the images (Figure 1D). The change in the lighting environment poses a significant challenge for object detection. Compared with the photos collected under sufficient lighting conditions, whose illuminance is 10826 lux, the remote sensing images of diseased trees collected under insufficient lighting conditions contain a large amount of noise. The visibility of objects such as diseased trees, houses, and roads is poor, resulting in blurred targets and severe distortion of details (Zuky et al., 2013).

2.2 Experimental environment configuration and training parameter settings

The YOLO v4 and its improved diseased tree detection algorithm run on the Windows 10.0 system with 32 GB of memory. This experiment uses an NVIDIA GeForce RTX 3080 Ti graphics card with 12 GB of memory and an 8-core 11th Gen Intel Core i7–11700KF CPU. The central frequency of the CPU is 3.6 GHz. Adopting an object detection algorithm based on PyTorch, the code runs in Python 3.7 environment. The object detection network is built using the Python language. In addition, third-party library packages such as numpy, opencv, and panda. Pytorch are Python-based machine learning libraries that can achieve powerful GPU acceleration.

The model parameters of YOLO v4 are set as shown (Table 1). The input image size is 416×416, the optimizer uses Adam, a total of 50 epochs are trained, the threshold of the prior box is set to 0.5, and

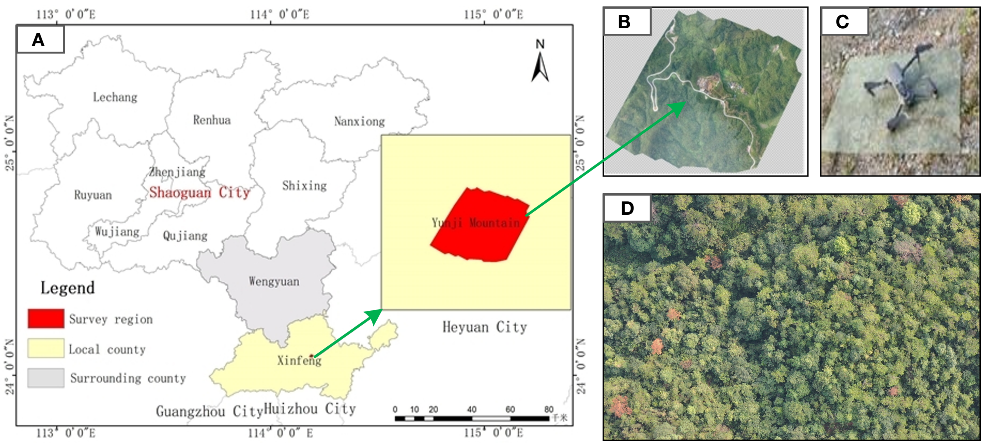


FIGURE 1
Geographical location diagram of UAV images acquisition. **(A)** Geographical location map of the research area **(B)** UAV orthophoto map **(C)** Drone appearance diagram **(D)** Single UAV aerial photo.

TABLE 1 Model training parameter settings.

Parameter Name	Freeze Training Phase	Unfreezing Training Stage
Epoch	1–25	25–50
Learning rate	0.001	0.0001
Batch size	4	4

the loss function is cross-entropy loss. The model's training process is divided into two stages: frozen training and unfrozen training (Liu et al., 2020). During the firm training process, the pretraining weights of the backbone network do not need to be trained, which can improve the training efficiency of the networks, and addicts were also used. Usually, an increase in detection accuracy leads to an increase in the complexity of the model, but due to the limitations of computer arithmetic thus leading to slow computation. Therefore, the use of higher computing power computers or multi-CPU parallel computing can improve the detection time and accuracy, but it is a challenge to balance the model size and cost control.

2.3 YOLO v4 network structure and detection process

YOLO v4 is an improvement on YOLO v3, retaining most of the structure of the YOLO v3. The improved parts of the network architecture include the input part, the leading feature extraction network, the neck network, and the head network (Bochkovskiy et al., 2020). Unlike YOLO v3, the feature extraction network of YOLO v4 is replaced by CSPDarknet53. The main feature extraction network comprises CSPDarknet53, and Cross Stage Partial (CSP) can effectively enhance the feature extraction ability of the convolutional network (Hui et al., 2021; Deng et al., 2022). The feature extraction network used by YOLO v4 is CSPDarknet, composed of the CSPX and CBM modules arranged alternately (Jiang et al., 2013). The structure of CSPX is shown in (Acharya, 2014; Fan et al., 2022).

First, visible light images of PWD trees collected by drones are annotated with the Labeling tool to save the detection box position and category information as an XML file. The training set images are rotated at different angles and input into YOLO v4 for training to increase the diversity of training samples. The trained model outputs detection boxes for the test set images (Figure 2).

In order to increase the detection accuracy of the model, this study modified the structure of the YOLO v4 model. By embedding attention mechanism and feature enhancement module in the YOLO v4 model improves the model's feature extraction ability. Determine the optimal model structure through ablation experiments.

3 Model improvement and methodology

3.1 Data enhancement and attention mechanism test

To increase the diversity of training samples, prevent over fitting during model training, and improve the accuracy of model detection. A widespread way to enhance image data is to perform geometric transformation, such as cropping, rotating, translating, and adjusting the image's brightness (Kim and Seo, 2018). This study used the rotation method to perform data augmentation on the training set samples. Five different angles, 15°, 60°, 195°, 240°, and 285°, were used to rotate the training set images, corresponding to Figures 3B–F, respectively. And the original image is showed in Fiqure 3A.

Convolutional neural networks contain the invariance property, which allows the network to preserve invariance to images under changing illuminations, sizes, and views. As a result, by rotating the acquired drone diseased tree photographs from various angles, the neural network will recognize these images as distinct (Moeskops

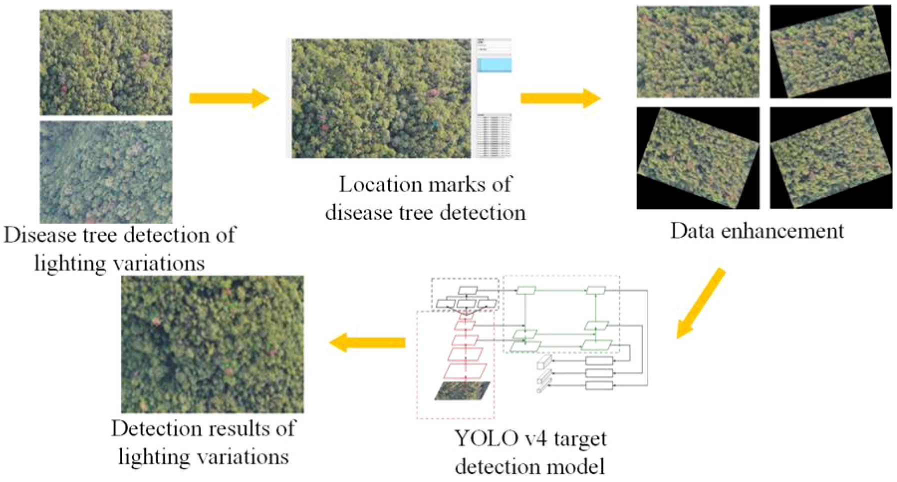

Disease tree detection of lighting variations
Location marks of disease tree detection
Data enhancement
Detection results of lighting variations
YOLO v4 target detection model

FIGURE 2
Disease tree target detection process for YOLO v4.

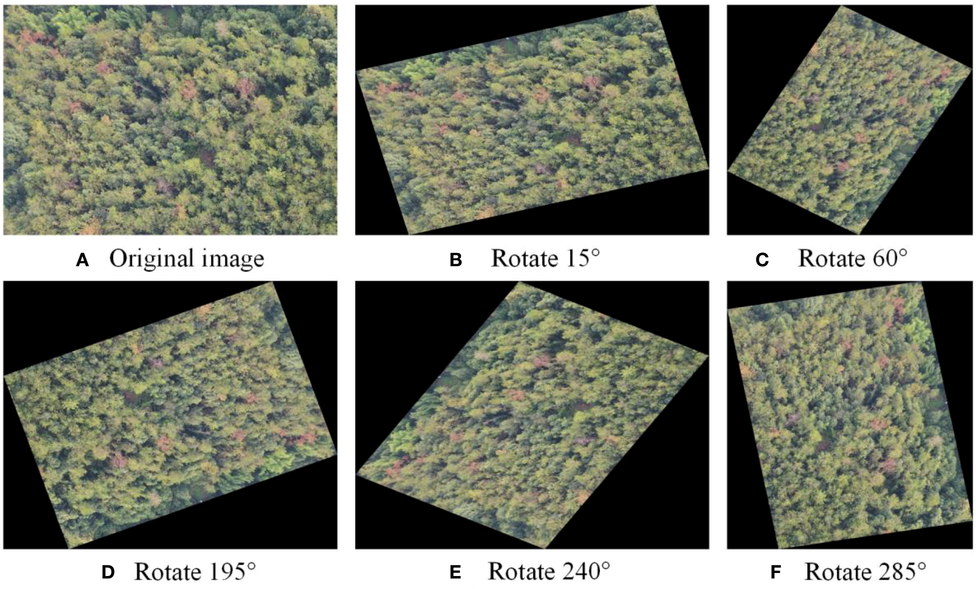


FIGURE 3
The diagram of data enhancement. **(A)** Original image **(B)** Rotate 15° **(C)** Rotate 60° **(D)** Rotate 195° **(E)** Rotate 240° **(F)** Rotate 285°.

et al., 2016). Due to the limited number of diseased tree images, a large sample set was added by augmenting the images through rotation at different angles. Five different angles were used to rotate the images, and five different images were obtained. The schematic diagram of the diseased tree images before and after sample augmentation is shown in the figure, and the number of images obtained after image transformation reached 7218, with 515 images in the test set. The above method was used to augment the sample data in the training set. The initial data in the training set was 1203 images, which was expanded six-fold. After rotating the images, the sample data set was expanded, and the expanded data was divided into a training set and a validation set. The training set contains 5052 images, the validation set contains 2166 images, and the test set contains 515 images.

The recognition results on the diseased pine tree dataset are compared (Table 2). It can be seen from the table that before data augmentation, the mean average precision (mAP) of the diseased pine tree detection was 77.45%. After data augmentation, the detection accuracy of the diseased pine tree was slightly improved, with an mAP of 77.81%, an increase of 0.36%. The accuracy increased by 0.22%, the specificity increased by 0.01, the recall increased by 2.22%, and precision decreased slightly. Overall, the detection accuracy of the diseased pine tree was improved. Data analysis shows that data augmentation can improve the detection effect of the diseased pine tree.

3.2 Attention mechanism addition position test

To determine the appropriate position for adding the attention mechanism, the detection performance of two different positions with the attention mechanism added in the YOLO v4 network structure was compared. Position 1 added the attention mechanism after the last three feature layers of the backbone feature network, before the feature pyramid network. In contract, position 2 added the attention mechanism before the three YOLO detection heads (Figure 4).

The detection accuracy of the attention mechanism at different positions is shown in Table 3. When the Squeeze-and-Excitation Networks (SENet) attention mechanism was added at position 1, the mAP of the test set was 79.29%. When the SENet attention mechanism was added at position 2, the mAP of the test set was 78.09%. The accuracy and recall in position 1 were higher than in position 2, with an increase of 0.42% and 1.76%, respectively, indicating that adding the attention mechanism at position 1 achieved higher detection accuracy and better detection performance.

Figure 5 shows the loss curves of the attention mechanism SENet at different embedding positions. The loss curves indicate that all three models can converge quickly during training. The loss in the test set decreases rapidly before 20 epochs and slows down when trained to 40 epochs. After 40 epochs, the loss value tends to

TABLE 2 Data enhancement effect.

	mAP/%	F1	Accuracy/%	Precision/%	Recall/%
Before augmentation	77.45	0.73	84.91	83.38	65.25
After augmentation	77.81	0.74	85.13	81.74	67.47

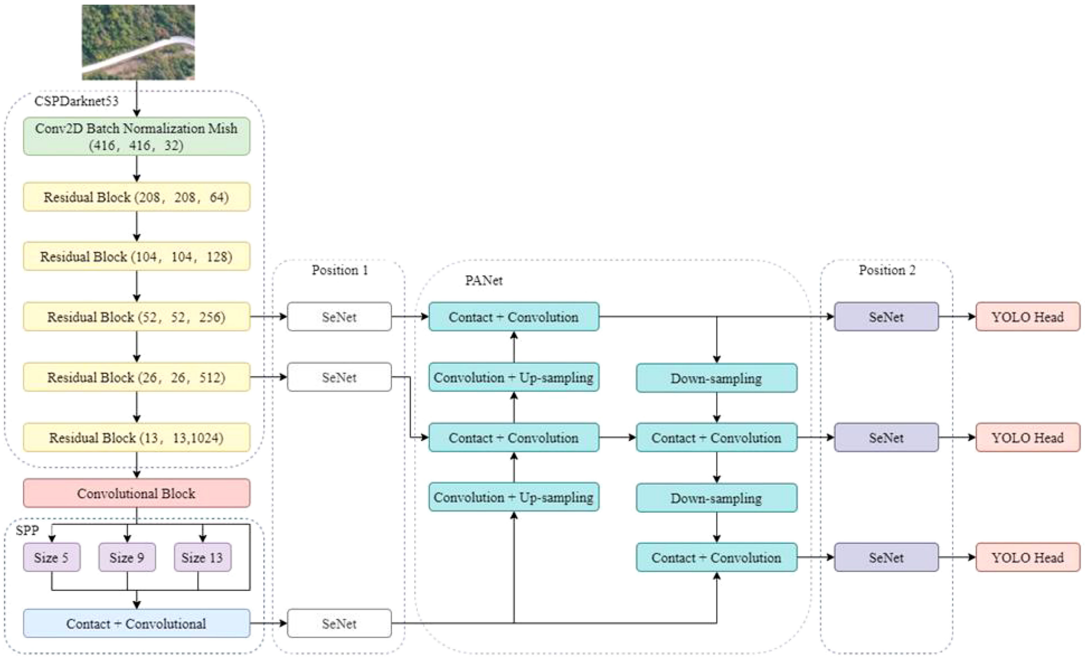


FIGURE 4
Different locations for adding attention mechanisms.

stabilize. However, the loss curve of the YOLO v4 model fluctuates more. After convergence, the model with attention mechanism SENet embedded in position 1 has a lower loss value. Therefore, the feature extraction effect of the attention mechanism SENet embedded in position 1 is better.

3.3 Attention mechanism type test

Channel attention module SENet includes squeeze, excitation, and weight calibration operations (Hu et al., 2018). The channel attention module SENet can learn feature weights based on the loss function and then re-calculate the weights for each feature channel so that the object detection model places more attention on the features, thereby improving the object detection accuracy (Figure 6).

The information propagation in the network structure follows the order of input feature map, global pooling layer, feature matrix with a size of 1×1×C, one-dimensional convolution structure with a convolution kernel size of k, and output feature map. The forward propagation process outputs channel weight parameters, which are then loaded into the input feature matrix using matrix multiplication. The core idea of efficient channel attention network (ECA-Net) is to introduce channel attention after the convolutional layer to dynamically adjust the response of different channels (Xue et al., 2022).

The convolutional block attention module (CBAM) feature module is composed of a channel attention feature module and a spatial attention feature module (Woo et al., 2018). The channel attention feature module performs global max pooling and global average pooling operations on the input feature map to obtain two feature maps, which are then input into a multi-layer perceptron network (Selvaraju et al., 2020). The multi-layer perceptron network sums the two feature maps obtained and inputs them into a sigmoid activation function to obtain the channel attention feature weights (Figure 7). Finally, the weights are multiplied by the input feature map to obtain the intermediate feature map.

To improve the accuracy of the YOLO v4 object detection model, this work introduced three attention mechanisms to the feature pyramid of the YOLO v4 model for feature extraction. Three types of attention mechanisms include SENet, ECA and CBAM (Figure 8).

The accuracy and detection speed of the model before and after improvement were tested in Table 4.

The mAP of the YOLO v4 model on the test set was 77.81%, with a recall of 65.25%, precision of 83.38%, and accuracy of 85.13%. After adding attention mechanisms, the detection accuracy of the model was improved to varying degrees. Among them, the addition of the SENet attention mechanism achieved the most significant improvement in detection accuracy, with an

TABLE 3 Evaluation indicators for detection accuracy of different addition positions in attention mechanisms.

	mAP/%	Accuracy/%	Precision/%	Recall/%	F1	FPS/(sheets/s)
Position 1	79.29	91.57	83.74	69.95	0.76	49.78
Position 2	78.09	91.15	83.84	67.19	0.75	50.24

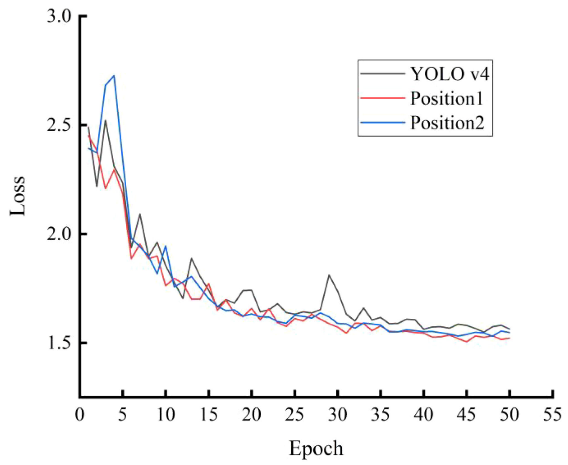


FIGURE 5
Loss curve of test set with different addition positions in attention mechanisms.

increase in mAP from 77.81% to 79.29%, an increase of 1.48% compared to the YOLO v4 model, and an increase in accuracy from 85.13% to 91.57%. FPS was used to assess the running speed of the four models. The running speed of the YOLO v4 model was 52.53 frames per second (fps), while the speed of the SENet-YOLO v4 model was slower, with an FPS of 49.78, a decrease of 2.75 fps compared to the original YOLO v4 model, indicating that the processing speed of the model decreased after adding SENet. Although the running speed of the model decreased, the added SENet showed an accuracy improvement of over 1% on the diseased pine tree dataset, indicating the effectiveness of the model improvement. Based on the evaluation of the four models' test accuracy and speed, the SENet-YOLO v4 model had the best testing performance. The accuracy of this model was the best, with an mAP of 79.29% on the test set, an increase of 1.48% compared to the YOLO v4 model. At the same time, among the four models, the CBAM-YOLO v4 model had the fastest processing speed, with an FPS of 57.32 on the test set, an increase of 0.9 fps compared to the YOLO v4 model. These show that the YOLO v4 model embedded with the SENet module can extract target features in more detail, which is beneficial for target classification. Although the detection speed decreased, the test accuracy was improved, and the model performance was optimized.

4 Model improvement and methodology

4.1 Ablation test

Three groups of ablation experiments were conducted to demonstrate the effectiveness of each improvement method used in the YOLO v4 network, including feature enhancement modules, feature fusion modules, and attention mechanisms. All parameters except for the testing module were kept consistent during the ablation experiments.

As different layers contain significantly different information, it is necessary to improve the adaptability of the feature layers to the target and the stability of the model for targets of different sizes. The working principle of this module is to perform three different operations on the input feature map (Figure 9). The second operation uses a 3x3 convolution operation, followed by the ReLU activation function, and ends with a 1x1 convolution operation. The third operation is the same as the second operation but with different padding for the 3x3 convolution. The three operations are then combined, and the enhanced feature map is output to improve the network's feature extraction ability further and acquire adequate information about the target in the feature map, acting as a feature enhancement (Liang et al., 2021).

In the YOLO v4 backbone feature extraction network, there are differences in the information contained in the feature maps of different layers (Sun et al., 2021). Deep feature maps contain rich semantic information, but small targets have less information and are usually used to detect large targets. Low-feature maps contain much detailed information but lack rich semantic information for detecting small targets. In order to better extract the feature information of diseased pine trees, a feature fusion module is designed, as shown in the Figure 9. This module adds three layers of feature maps to obtain the context information of diseased pine trees fully and then adds the outputs of three branches to achieve feature fusion (Sun et al., 2005). Three different scales of the backbone feature extraction network in the YOLO v4 model. The

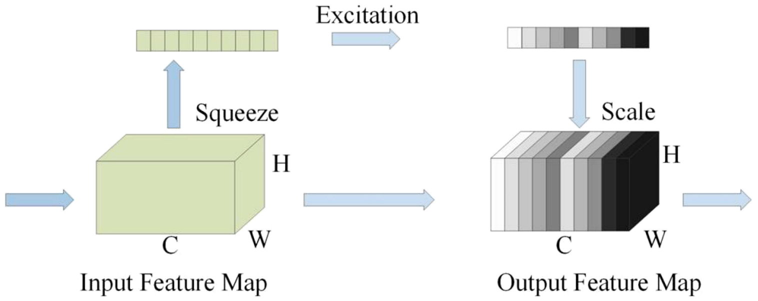


FIGURE 6
SENet channel attention mechanism.

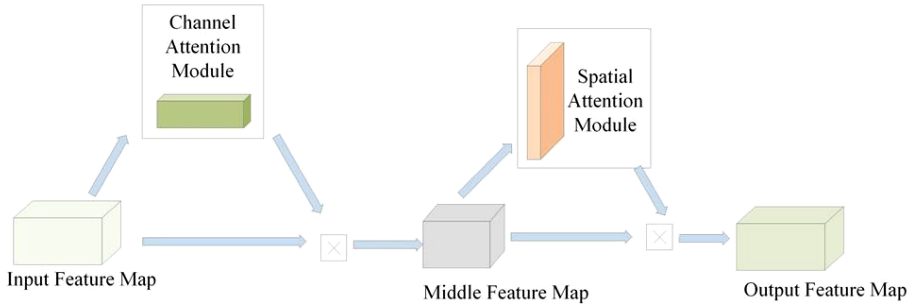


FIGURE 7
Schematic diagram of CBAM module.

working principle of this module is: three feature maps of different sizes are used as inputs for the three branches, and the input feature maps of the middle branch are enlarged to adjust the size of the feature maps, and then 3×3 to extract the features of the input feature map, and finally use the Activation function rectified linear unit (ReLU). The operation process of the input feature map for branch 3 is the same as that for branch 2. Due to the difference in size between the input feature maps of the third branch and the input feature maps of the second branch, there is a difference in magnification between the input feature maps of the third branch and the second branch. The feature maps are processed by the first branch, and the other two branches are added and fused. The fused feature map is further divided into three branches for processing, and the feature map of the first branch is processed through three steps. After the convolution operation of 3×3, use the Activation function ReLU to process, and output the feature map (Figure 10). The difference between the other two branches is that before activating the operation, the maximum pooling operation is used to adjust the size of the feature map to match the input feature map size of the corresponding branch. By fusing feature maps from adjacent layers through the feature fusion module, the semantic differences between different feature channel layers are further reduced. This module can be used to collect contextual information of different scales and improve detection accuracy (Wu et al., 2021).

The effectiveness of the target detection network improvement methods was evaluated using the mAP evaluation metric, and the impact of each module on the overall network performance was analyzed. The "√" in the table indicates that the corresponding module was added to the original YOLO v4 network, while the absence of "√" indicates that the corresponding module was not added. The specific experimental results are shown in the table. The comparison of the results of the ablation experiments is shown in Table 5.

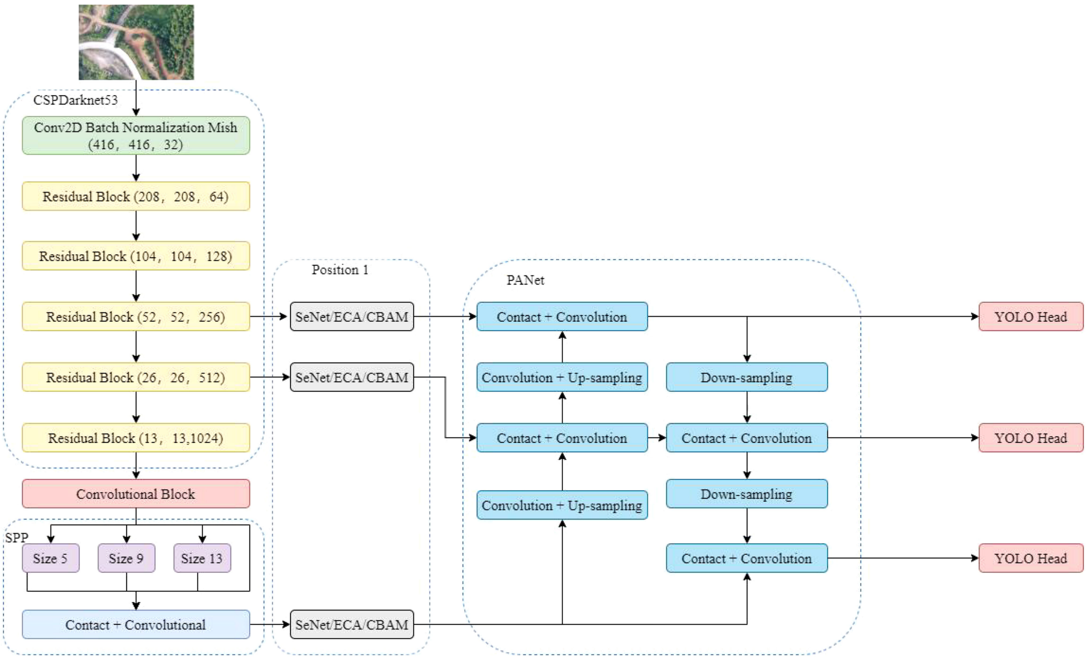


FIGURE 8
The addition positions of different attention mechanisms.

TABLE 4 Evaluation of detection accuracy for different attention mechanisms.

	mAP/%	Accuracy/%	Precision/%	Recall/%	FPS/(sheets/s)
YOLO v4	77.81	85.13	83.38	65.25	52.53
ECA-YOLO v4	79.00	91.33	83.75	68.26	51.98
SENet-YOLO v4	79.29	91.57	83.74	69.95	49.78
CBAM-YOLO v4	79.07	91.15	85.22	65.91	53.1

The study's results on the effectiveness of the feature enhancement module, feature fusion module, and attention mechanism SENet show that the mAP of the basic network on the diseased pine tree dataset is 77.81%. After adding the feature enhancement module, the mAP increased to 78.61%, resulting in a 0.8% improvement. The reason is that introducing the feature enhancement module can enhance the weight information of the target object and extract features more comprehensively and accurately. After adding the attention mechanism to the primary network, the mAP increased to 79.29%, resulting in a 1.48% improvement. As shown by the results of experiments 1 and 3, not all modules can improve the detection performance of the model. The mAP of the test set fell after adding the feature fusion module, indicating that the feature fusion module's results were unstable and unsuitable for implementation in the YOLO v4 network. The mAP climbed to 79.91% after adding the feature enhancement module and attention mechanism to the original YOLO v4 network, representing a 2.1% improvement. The combination of the feature improvement module and the attention mechanism SENet was chosen to be the best network model after screening. Thus, added the SENet attention mechanism and the feature improvement module after the last three feature layers of the YOLO v4 backbone feature network, the accuracy of YOLO v4 disease tree detection has been improved 2.1%. The improvement of detection performance is related to the feature extraction ability of the feature enhancement module. The feature enhancement module is self-designed, which can adapt to different lighting changes.

4.2 Feature visualization analysis

The Gradient-weighted Class Activation Mapping (Grad-CAM) tool was used to analyze the feature extraction process of the network, extract heat maps after embedding the improvement modules, and analyze the impact of the improvement modules on target feature extraction. The brightest point at the center is the position of the center point, and the closer the position is to the vital point of the target, the larger the activation function value (Figure 11). The darker the color of the center point, the more obvious the feature. Before embedding the improvement modules, the YOLO v4 network randomly extracted the features of diseased trees and did not pay enough attention to the features of the diseased tree location. After embedding the improvement modules, the critical feature channels accounted for a more significant proportion, the network obtained a larger receptive field, and the improved YOLO v4 network could more effectively extract the feature information of diseased trees, making it easier to distinguish the location of diseased trees from the image. The improved YOLO v4 model performs better in detecting diseased trees, not only recognizing a larger number of diseased trees, but also improving the model's ability to recognize green backgrounds as yellow diseased trees. The improved YOLO v4 model can extract more feature information about disease trees and improve the detection performance of disease trees under complex lighting conditions. In order to better achieve lightweight deployment of models, future research focuses on reducing model volume and improving detection speed while minimizing model accuracy loss.

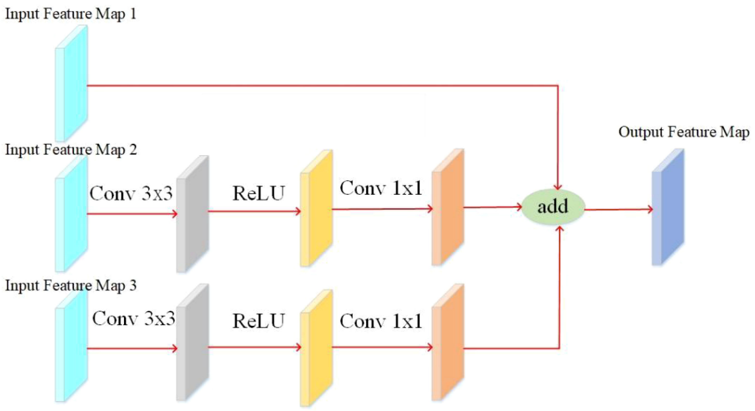


FIGURE 9
Feature enhancement module.

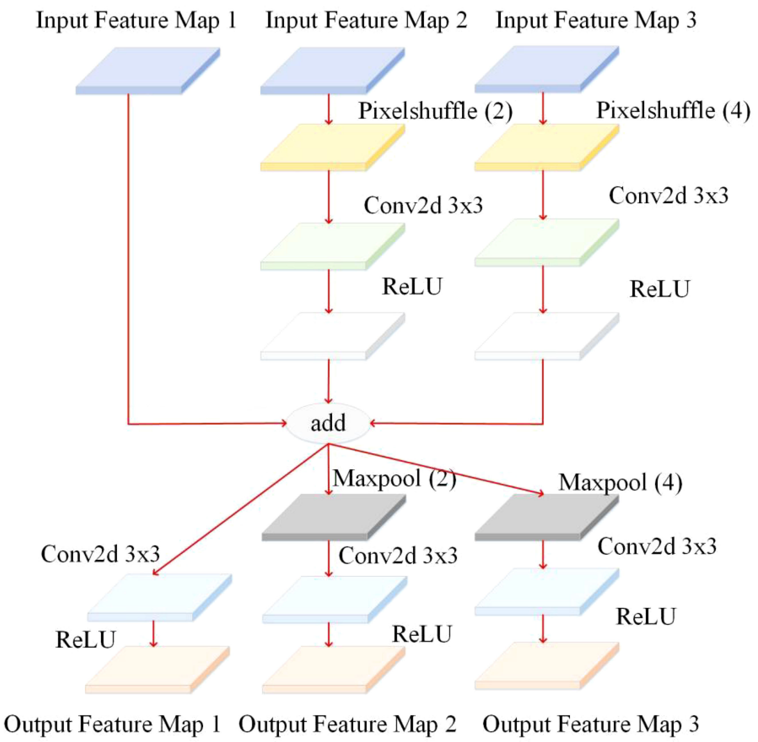


FIGURE 10
Feature fusion module.

4.3 Visualization of prediction results

The test set images were used to analyze and evaluate the results of diseased tree recognition. A total of 515 test set images were selected to evaluate the model's prediction results, and the prediction results of two models in robust light environments are shown (Figure 12).

It can be seen that after the model was improved, it could detect the specific location of the diseased tree, and the confidence values were all increased (Figure 12B). In the predicted images, there were fifteen diseased trees of different colors with strong light, and some of the diseased tree crowns had small contours and colors similar to those of surrounding trees, as well as overlapping crowns. In this complex image background, both models could identify the location of the diseased trees accurately. Among them, the YOLO v4 model identified ten diseased trees, and three were not correctly identified, with false positives (Figure 12A). After adding the channel attention mechanism SENet and feature enhancement module, the improved YOLO v4 model correctly identified thirteen diseased trees, three more than the YOLO v4 model. The reason why the YOLO v4 model failed to detect the one missed diseased tree correctly may be due to the obstruction of other healthy trees in the crown, which affected the feature extraction of the model.

TABLE 5 Comparison of ablation experiment effects.

Number	Feature Enhancement Module	Feature Fusion Module	Attention Mechanism	mAP/%
1				77.81
2	√			78.61
3		√		77.57
4			√	79.29
5		√	√	79.14
6	√	√		78.76
7	√		√	79.91
8	√	√	√	79.39

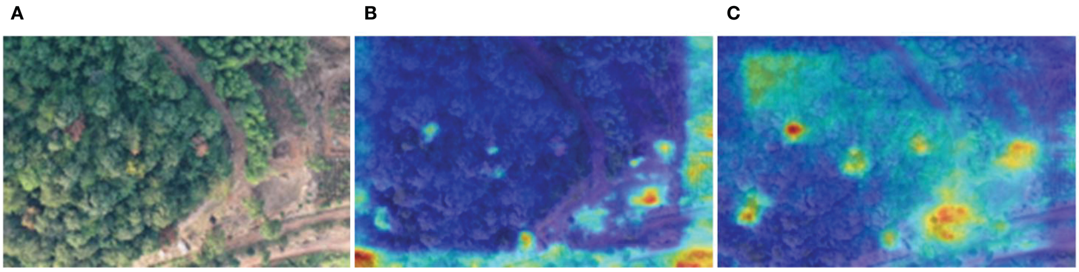


FIGURE 11
Thermal diagram before and after embedding the improved module. **(A)** Network Input Diagram. **(B)** The diagram before the improvement module is embedded. **(C)** The diagram after the improvement module is embedded.

4.4 Comparative experiments with other object detection models

To compare the comprehensive performance of the improved YOLO v4 model in this study, Single Shot Multibox Detector (SSD), Faster RCNN, YOLO v3, and YOLO v5 were compared, showing the effectiveness of the model in detecting diseased pine trees, as shown in Table 6.

The improved YOLO v4 model has the highest parameters, which are increased by 230.535 M, 228.545 M, and 194.871 M compared to SSD, Faster RCNN, and YOLO v3, respectively. This is due to the addition of the SENet module and feature enhancement module to the YOLO v4 network.

Moreover, the improved YOLO v4 model has the highest mAP, which is increased by 68.2%, 62.49%, 54.68%, and 1.22% compared to SSD, Faster RCNN, YOLO v3, and YOLO v5, respectively. The

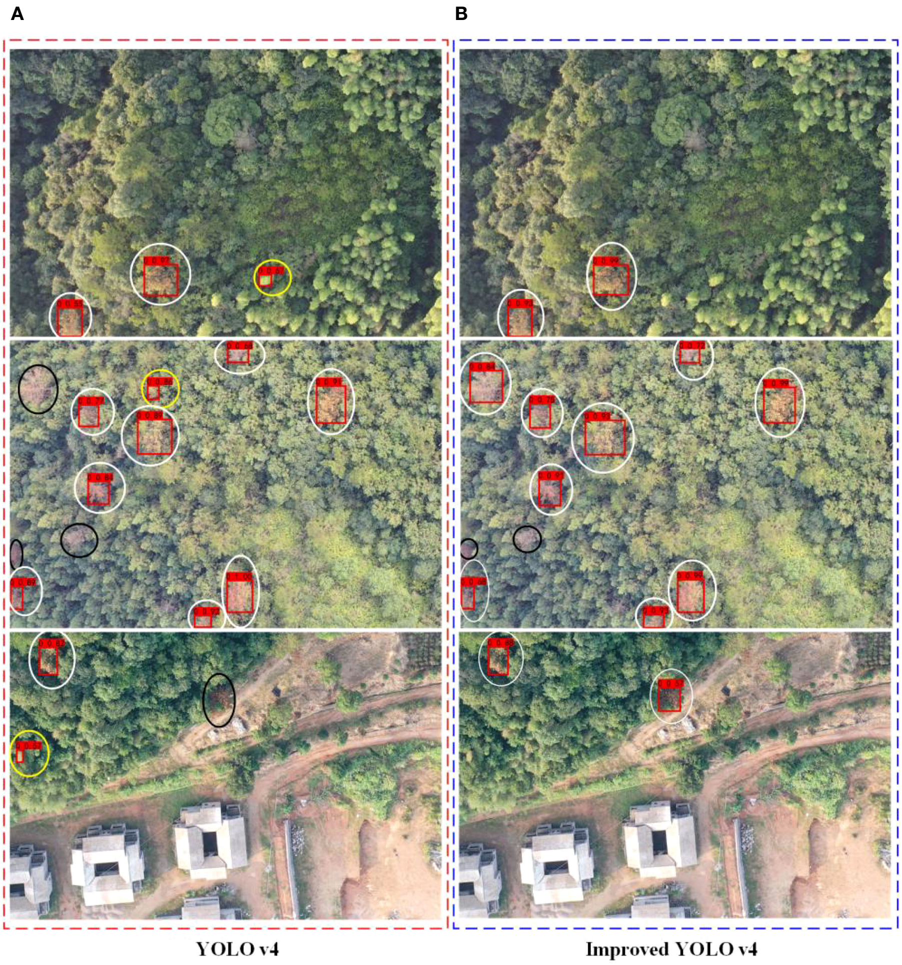


FIGURE 12
Remote sensing image recognition results under strong light environment. * white circles indicate correct detections, black circles indicate missed detections, yellow circles indicate misdetections. **(A)** YOLO v4 detection results **(B)** Improved YOLO v4 detection results.

TABLE 6 Experimental comparison results of different models.

Models	mAP/%	Recall/%	Precision/%	Params/M
SSD	11.71	10.06	64.67	26.285
Faster RCNN	17.42	25.8	21.42	28.275
YOLO v3	25.23	21.95	84.00	61.949
Improved YOLO v4	79.91	67.15	86.36	256.82
YOLO v5	78.69	69.98	81.63	7.022

model's precision is also the highest, which has increased by 21.69%, 64.94%, 2.36%, and 4.73% compared to SSD, Faster RCNN, YOLO v3, and YOLO v5, respectively. Although, the improved YOLO v4 model has the highest parameters and requires more computation, its performance is the best, as its mAP is 79.91%, the highest among the five models, indicating that the improved YOLO v4 model has higher detection accuracy. Therefore, the model improvement in this study is effective.

5 Conclusion and discussion

Since the changes in lighting conditions can lead to a decrease in image quality during unmanned aerial vehicle detection of pine wilt disease, this study used unmanned aerial vehicles to create a sample set of diseased trees at different time periods, making the deep learning model trained more generalizable and improving the performance of object recognition. The application of the YOLO v4 algorithm in the field of diseased tree object detection was studied, and the CSPDarknet53 network structure was used to complete the feature extraction process. In contrast, the feature pyramid network structure was used to enhance the feature extraction capability of the convolutional neural network. The mAP of the YOLO v4 model was 77.81%. By comparing experiments, the type of attention mechanism and its addition position in the YOLO v4 network were determined, and the detection effect was best when the attention mechanism module SENet was added before the feature pyramid network structure. The ablation experiment found that the optimal combination was the object detection model that combined the channel attention mechanism SENet and feature enhancement module. The mAP of the model was 79.91%, an increase of 2.1% after improvement, indicating that the channel attention mechanism SENet combined with feature enhancement module can effectively enhance the ability to recognize detection targets. Under the same conditions, the mAP of the improved YOLO v4 model was increased by 68.2%, 62.49%, 54.68%, and 1.22% compared to SSD, Faster RCNN, YOLO v3, and YOLO v5, respectively, indicating that the model can achieve high-precision detection of diseased trees caused by PWD under changing light conditions. In 2021, Wu estimated the power of the hyperspectral method, LiDAR and their combination to predict the infection stages of PWD using the random forest (RF) algorithm. The results showed that the combination of hyperspectral method and LiDAR had the best accuracies (Yu et al., 2021). The improved YOLO v4 model has a high recognition accuracy for diseased trees, which can achieve precise positioning and recognition of pine wilt disease trees under changing light conditions. This is critical in guiding the prevention and control of pine wilt disease.

The ablation experimental results have demonstrated the optimization effect of the improved module on the YOLOv4 detection network. Although the improved YOLOv4 algorithm performs well in the target detection task of diseased tree images captured by drones, there is still room for improvement in detection accuracy and speed. The current challenge is how to count the number of diseased trees in the image, which requires post-processing of the model but increases its complexity. Following that, there is a goal to do research on lightweight models and build software and hardware implementation of a real-time target detection system suited for drones to detect disease trees. Moreover, the system provides ideas for lychee disease detection in lychee gardens.

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

ZZ: Conceptualization, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing. CH: Project administration, Software, Visualization, Writing – original draft, Writing – review & editing. XW: Conceptualization, Investigation, Writing – original draft, Writing – review & editing. HL: Data curation, Formal analysis, Writing – original draft, Writing – review & editing. JL: Supervision, Validation, Writing – original draft, Writing – review & editing. JZ: Methodology, Resources, Visualization, Writing – original draft, Writing – review & editing. SS: Data curation, Formal analysis, Resources, Validation, Writing –

original draft, Writing – review & editing. WW: Funding acquisition, Resources, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by 2024 Rural Revitalization Strategy Special Funds Provincial Project (2023LZ04), and the Guangdong Province (Shenzhen) Digital and Intelligent Agricultural Service Industrial Park (FNXM012022020-1), Construction of Smart Agricultural Machinery and Control Technology Research and Development.

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

Acharya, A. (2014). Template matching based object detection using HOG feature pyramid. *Comput. Sci.* 11, 689–694.

Asai, E., and Futai, K. (2011). The effects of long-term exposure to simulated acid rain on the development of pine wilt disease caused by *Bursaphelenchus xylophilus*. *For. Pathol.* 31, 241–253. doi: 10.1046/j.1439-0329.2001.00245.x

Barnich, O., and Van, D. M. (2011). ViBe: A universal background subtraction algorithm for video sequences. *IEEE Trans. Image Process.* 20, 1709–1724. doi: 10.1109/TIP.2010.2101613

Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. doi: 10.48550/arXiv.2004.10934

Deng, F., Xie, Z., Mao, W., Li, B., Shan, Y., Wei, B., et al. (2022). Research on edge intelligent recognition method oriented to transmission line insulator fault detection. *Int. J. Electrical Power Energy Syst.* 139, 108054. doi: 10.1016/j.ijepes.2022.108054

Fan, S., Liang, X., Huang, W., Zhang, V. J., and Pang, Q. (2022). Real-time defects detection for apple sorting using NIR cameras with pruning-based YOLOV4 network. *Comput. Electron. Agric.* 193, 106715. doi: 10.1016/j.compag.2022.106715

Gao, R., Shi, J., Huang, R., Wang, Z., and Luo, Y. (2015). Effects of pine wilt disease invasion on soil properties and masson pine forest communities in the three gorges reservoir region, China. *Ecol. Evol.* 5, 1702–1716. doi: 10.1002/ece3.1326

Hosang, J., Benenson, R., Dollár, P., and Schiele, B. (2016). What makes for effective detection proposals? *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 814–830. doi: 10.1109/TPAMI.2015.2465908

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. (2018). "Squeeze-and-excitation networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 7132–7141 (Salt Lake City: IEEE). doi: 10.1109/TPAMI.2019.2913372

Huang, J., Lu, X., Chen, L., Sun, H., Wang, S., and Fang, G. (2022). Accurate identification of pine wood nematode disease with a deep convolution neural network. *Remote Sens.* 14, 913. doi: 10.3390/rs14040913

Hui, T., Xu, Y. L., and Jarhinbek, R. (2021). Detail texture detection based on Yolov4-tiny combined with attention mechanism and bicubic interpolation. *IET image Process.* 12), 2736–2748. doi: 10.1049/ipr2.12228

Jiang, M., Wang, Y., Xia, L., Liu, F., Jiang, S., and Huang, W. (2013). The combination of self-organizing feature maps and support vector regression for solving the inverse ECG problem. *Comput. Mathematics Appl.* 66, 1981–1990. doi: 10.1016/j.camwa.2013.09.010

Kentsch, S., Caceres, M. L. L., Serrano, D., Roure, F., and Diez, Y. (2020). Computer vision and deep learning techniques for the Analysis of drone-acquired forest images, a transfer learning study. *Remote Sens.* 12), 1287. doi: 10.3390/rs12081287

Khan, M. A., Ali, M., Shah, M., Mahmood, T., Ahmad, M., Jhanjhi, N. Z., et al. (2021). Machine learning-based detection and classification of walnut fungi diseases. *Computers Materials Continua(Tech Sci. Press)* 3), 771–785. doi: 10.32604/IASC.2021.018039

Kikuchi, T., Cotton, J. A., Dalzell, J. J., Hasegawa, K., Kanzaki, N., McVeigh, P., et al. (2011). Genomic insights into the origin of parasitism in the emerging plant pathogen *Bursaphelenchus xylophilus*. *PloS Pathog.* 7, e1002219. doi: 10.1371/journal.ppat.1002219

Kim, J., and Seo, K. (2018). Performance analysis of data augmentation for surface defects detection. *Trans. Korean Institute Electrical Engineers* 67, 669–674. doi: 10.5370/KIEE.2018.66.5.669

Kobayashi, F., Yamane, A., and Ikeda, T. (2003). The Japanese pine sawyer beetle as the vector of pine wilt disease. *Annu. Rev. Entomol.* 29, 115–135. doi: 10.1146/annurev.en.29.010184.000555

Kuroda, K. (2010). Mechanism of cavitation development in the pine wilt disease. *For. Pathol.* 21, 82–89. doi: 10.1111/j.1439-0329.1991.tb00947.x

Li, Y., Dong, H., Li, H., Zhang, X., and Zhang, B. (2020). Multi-block SSD based on small object detection for UAV railway scene surveillance. *Chin. J. Aeronautics* 33, 1747–1755. doi: 10.1016/j.cja.2020.02.024

Li, J., Li, J., Zhao, X., Su, X., and Wu, W. (2023). Lightweight detection networks for tea bud on complex agricultural environment via improved YOLO v4. *Comput. Electron. Agric.* 211), 107955. doi: 10.1016/j.compag.2023.107955

Liang, H., Yang, J., and Shao, M. (2021). FE-RetinaNet: Small target detection with parallel multi-scale feature enhancement. *Multidiscip. Digital Publishing Institute* 13), 950. doi: 10.3390/sym13060950

Lifkooee, M. Z., Soysal, M., and Sekeroglu, K. (2018). Video mining for facial action unit classification using statistical spatial–temporal feature image and LoG deep convolutional neural network. *Mach. Vis. Appl.* 30, 41–57.

Liu, Y., Zhai, W., and Zeng, K. (2020). On the study of the freeze casting process by artificial neural networks. *ACS Appl. Materials Interfaces* 12, 40465–40474. doi: 10.1021/acsami.0c09095

Long, N., Gianola, D., Weigel, K., and Avendano, S. (2015). Machine learning classification procedure for selecting SNPs in genomic selection: application to early mortality in broilers. *Developments Biologicals* 124, 377–389. doi: 10.1111/j.1439-0388.2007.00694.x

Moeskops, P., Viergever, M. A., Mendrik, A. M., de Vries, L. S., Benders, M. J. N. L., and Igum, I. (2016). Automatic segmentation of MR brain images with a convolutional neural network. *IEEE Trans. Med. Imaging* 35, 1252–1261. doi: 10.1109/TMI.2016.2548501

Park, J., Sim, W., and Lee, J. (2016). Detection of trees with pine wilt disease using object-based classification method. *J. For. Environ. Sci.* 32, 384–391. doi: 10.7747/JFES.2016.32.4.384

Schröder, T., Mcnamara, D. G., and Gaar, V. (2010). Guidance on sampling to detect pine wood nematode Bursaphelenchus xylophilus in trees, wood and insects. *Eppo Bulletin* 39, 179–188.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Devi, P., and Batra, D. (2020). Grad-CAM: visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vision* 128, 336–359. doi: 10.1007/s11263-019-01228-7

Sun, X., Liu, T., Yu, X., and Pang, B. (2021). Unmanned surface vessel visual object detection under all-weather conditions with optimized feature fusion network in YOLOv4. *J. Intelligent Robotic Syst.* 103, 55. doi: 10.1007/s10846-021-01499-8

Sun, Q., Zeng, S., Liu, Y., Hengc, P., and Xiaa, D. (2005). A new method of feature fusion and its application in image recognition. *Pattern Recognition* 38, 2437–2448. doi: 10.1016/j.patcog.2004.12.013

Syifa, M., Park, S., and Lee, C. (2020). Detection of the pine wilt disease tree candidates for drone remote sensing using artificial intelligence techniques. *Engineering* 6, 919–926. doi: 10.1016/j.eng.2020.07.001

Tang, L., and Shao, G. (2015). Drone remote sensing for forestry research and practices. *J. Forestry Res.* 26, 791–797. doi: 10.1007/s11676-015-0088-y

Tian, X., and Daigle, H. (2019). Preferential mineral-microfracture association in intact and deformed shales detected by machine learning object detection. *J. Natural Gas Sci. Eng.* 63, 27–37. doi: 10.1016/j.jngse.2019.01.003

Vicente, C., Espada, M., Vieira, P., and Mota, M. (2012). Pine wilt disease: a threat to European forestry. *Eur. J. Plant Pathol.* 133, 497–497. doi: 10.1007/s10658-012-9979-3

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). CBAM: convolutional block attention module. *Comput. Vision-ECCV* 2018, 3–19.

Wu, W., Zhang, Z., Zheng, L., Han, C., and Wang, X. (2020). Research progress on the early monitoring of pine wilt disease using hyperspectral techniques. *Sensors* 20, 3729. doi: 10.3390/s20133729

Wu, Y., Zhao, W., Zhang, R., and Jiang, F. (2021). AMR-Net: Arbitrary-oriented ship detection using attention module, multi-Scale feature fusion and rotation pseudo-label. *IEEE Access* 9), 68208–68222. doi: 10.1109/ACCESS.2021.3075857

Xu, Y., Yu, G., Wang, Y., Wu, X., and Ma, Y. (2017). Car Detection from low-altitude UAV imagery with the Faster R-CNN. *J. Adv. Transp.* 2823617. doi: 10.1155/2017/2823617

Xue, H., Sun, M., and Liang, Y. (2022). ECANet: Explicit cyclic attention-based network for video saliency prediction. *Neurocomputing* 468), 233–244. doi: 10.1016/j.neucom.2021.10.024

Yu, R., Luo, Y., Zhou, Q., Zhang, X., Wu, D., and Ren, L. (2021). A machine learning algorithm to detect pine wilt disease using UAV-based hyperspectral imagery and LiDAR data at the tree level. *Int. J. Appl. Earth Observation Geoinformation* 101, 102363. doi: 10.1016/j.jag.2021.102363

Yun, J. E., Kim, J., and Park, C. G. (2012). Rapid diagnosis of the infection of pine tree with pine wood nematode(*Bursaphelenchus xylophilus*) by use of host-tree volatiles. *J. Agric. Food Chem.* 60, 7392–7397. doi: 10.1021/jf302484m

Zhang, J. C., Huang, Y. B., Pu, R. L., Gonzalez-Moreno, P., Yuan, L., Wu, K., et al. (2019). Monitoring plant diseases and pests through remote sensing technology: a review. *Comput. Electron. Agric.* 165, 104943. doi: 10.1016/j.compag.2019.104943

Zhang, B., Ye, H., Lu, W., Huang, W., Wu, B., Hao, Z., et al. (2021). A spatiotemporal change detection method for monitoring pine wilt disease in a complex landscape using high-resolution remote sensing imagery. *Remote Sens.* 13, 2083. doi: 10.3390/rs13112083

Zhang, T., and Zhang, X. (2019). High-speed ship detection in SAR images based on a grid convolutional neural network. *Remote Sens.* 11, 1206–1229. doi: 10.3390/rs11101206

Zuky, A. A., Salman, S., and Saleh, A. A. (2013). Study the quality of image enhancement by using retinex technique which capture by different lighting (sun and tungsten). *Foundation Comput. Sci. (FCS)* 73, 31–38. doi: 10.5120/12773-9862

TYPE Original Research
PUBLISHED 06 August 2024
DOI 10.3389/fpls.2024.1420584

A lightweight dual-attention network for tomato leaf disease identification

Enxu Zhang, Ning Zhang*, Fei Li and Cheng Lv

Engineering Research Center of Hydrogen Energy Equipment& Safety Detection, Universities of Shaanxi Province, Xijing University, Xi'an, China

Check for updates

OPEN ACCESS

EDITED BY
Marcin Woźniak,
Silesian University of Technology, Poland

REVIEWED BY
Aibin Chen,
Central South University Forestry and Technology, China
Yunchao Tang,
Dongguan University of Technology, China

*CORRESPONDENCE
Ning Zhang
✉ zhangning@xijing.edu.cn

RECEIVED 20 April 2024
ACCEPTED 15 July 2024
PUBLISHED 06 August 2024

CITATION
Zhang E, Zhang N, Li F and Lv C (2024) A lightweight dual-attention network for tomato leaf disease identification.
Front. Plant Sci. 15:1420584.
doi: 10.3389/fpls.2024.1420584

Tomato disease image recognition plays a crucial role in agricultural production. Today, while machine vision methods based on deep learning have achieved some success in disease recognition, they still face several challenges. These include issues such as imbalanced datasets, unclear disease features, small inter-class differences, and large intra-class variations. To address these challenges, this paper proposes a method for classifying and recognizing tomato leaf diseases based on machine vision. First, to enhance the disease feature details in images, a piecewise linear transformation method is used for image enhancement, and oversampling is employed to expand the dataset, compensating for the imbalanced dataset. Next, this paper introduces a convolutional block with a dual attention mechanism called DAC Block, which is used to construct a lightweight model named LDAMNet. The DAC Block innovatively uses Hybrid Channel Attention (HCA) and Coordinate Attention (CSA) to process channel information and spatial information of input images respectively, enhancing the model's feature extraction capabilities. Additionally, this paper proposes a Robust Cross-Entropy (RCE) loss function that is robust to noisy labels, aimed at reducing the impact of noisy labels on the LDAMNet model during training. Experimental results show that this method achieves an average recognition accuracy of 98.71% on the tomato disease dataset, effectively retaining disease information in images and capturing disease areas. Furthermore, the method also demonstrates strong recognition capabilities on rice crop disease datasets, indicating good generalization performance and the ability to function effectively in disease recognition across different crops. The research findings of this paper provide new ideas and methods for the field of crop disease recognition. However, future research needs to further optimize the model's structure and computational efficiency, and validate its application effects in more practical scenarios.

KEYWORDS
tomato disease identification, machine vision, deep learning, lightweight models, attention mechanisms

1 Introduction

Originating from the indigenous regions of South America, the tomato is a crop with a short growth cycle, low environmental requirements, and rich nutritional value, and has been widely cultivated around the world (Mitchell et al., 2007; Bhatkar et al., 2021). In agricultural production, tomato plants are susceptible to a variety of pathogenic bacteria and environmental factors such as fungi, bacteria, and viruses, resulting in the occurrence of white spot disease, early blight, mosaic virus, leaf mold, and other diseases. These diseases are mainly manifested in the leaves and affect their function, thus affecting the yield and quality of tomatoes. Especially under conditions of frequent rainfall or high humidity, tomato plants are more likely to be infected with diseases, resulting in seedling rot and stem and fruit rot (Vos et al., 2014). However, the diversity and complexity of tomato diseases pose great challenges to control. During the occurrence of these diseases, early symptoms usually appear on tomato leaves, showing abnormal characteristics that are different from those of healthy leaves, as detailed in Supplementary Table S2. Early and accurate disease identification in agricultural production can effectively reduce the yield loss caused by diseases. However, traditional manual methods of disease identification are inefficient and often require specialized agricultural expertise, hindering widespread and accurate identification of diseases and resulting in wasted labor and medicines (Patil and Thorat, 2016). Therefore, there is an urgent need for a convenient and rapid detection method that can non-destructively identify plant developmental abnormalities at an early stage to mitigate the impact of diseases on agricultural production (Eli-Chukwu and Ogwugwam, 2019). Nowadays, with the rise of precision agriculture and smart agriculture concepts, it is important to use machine vision technology to assist agricultural production, realize the accurate identification of tomato diseases, take management measures and prevention strategies in a timely manner, and improve crop yields (Affonso et al., 2017).

Identifying plant leaf diseases falls under the field of agricultural information technology. The rapid development and advancement of machine vision technology provide new directions for crop disease identification and combined with robotics technology, can achieve more flexible agricultural production (Tang Y, et al., 2023; Ye et al., 2023). Initially, machine learning algorithms were used to extract image features and classify them. (Xie and He, 2016) used a gray-level co-occurrence matrix to extract texture features and classified them using the K-NN algorithm. (Akbarzadeh et al., 2018) employed support vector machines to efficiently distinguish weeds based on the morphological features of broadleaf and narrow plants. However, using traditional machine learning algorithms for disease identification typically relies on single global features such as color, texture, and shape. This often requires researchers to manually design image feature extraction methods based on experience, resulting in a limited ability to identify various types of diseases and insufficient recognition capability to meet the needs of large-scale agricultural disease identification (Khan et al., 2018).

With the development of deep learning, it has shown significant advantages in feature extraction and recognition tasks. Deep learning-based disease image recognition has become an important method in current research. Convolutional neural network (CNN) models, by introducing operations such as local connections and weight sharing, have made significant progress in various crop disease identification tasks and are currently considered one of the best algorithms for pattern recognition tasks (Prathibha et al., 2017). To address the data imbalance problem in cassava disease detection based on CNN models, (Gnanasekaran and Opiyo, 2020) used methods such as class weights, SMOTE, and focal loss functions to enhance the model's recognition performance on imbalanced datasets. (Wu, 2021) constructed a dual-channel convolutional neural network model by integrating ResNet50 and VGG19 network models, thereby improving the network model's ability to extract disease features and achieve high-precision recognition of maize diseases. Additionally, to address the challenge of identifying grape diseases in natural environments, (Cai et al., 2023) used an improved MSR algorithm to process images and employed a Siamese network structure to extract image features, achieving model lightweighting. (Sanida et al., 2023) improved recognition capability by combining VGG and Inception modules and using a multi-scale approach to enhance the model. (Uddin et al., 2024) integrated Inception V3 and DenseNet201 with the addition of the attention mechanism VIT to obtain the E2ETCA network model for rice disease identification. (Deng et al., 2023) used a combination of ResNest and Ghost to obtain GR-ARNet, which separately processed the depth feature information and channels of images, achieving efficient identification of banana leaf diseases. In agricultural production, to effectively prevent and control diseases, (Kamal et al., 2019) proposed a MobileNet model improved by deep separable convolution, which outperformed VGG and GoogleNet models on the 55-class PlantVillage leaf dataset. (Waheed et al., 2020) proposed an optimized DenseNet model for identifying maize leaf diseases.

In neural network models, the attention mechanism is an effective method to improve the model's recognition performance. Neural network models can use the attention mechanism to compute the weights of input images, selectively emphasizing areas of interest through feature weighting, thereby aiding feature extraction. Currently, many achievements have been made, such as SE-Net (Hu et al., 2018), ECA-Net (Wang et al., 2020), CBAM (Woo et al., 2018), and Coordinate Attention (Hou et al., 2021). Additionally, through the efforts of many researchers, the attention mechanism can be applied to plant disease detection. For example, (Zhao et al., 2022) embedded the CBAM attention mechanism into the Inception network model, thereby enhancing the network model's ability to identify diseases in maize, potatoes, and tomatoes. (Zeng and Li, 2020) proposed a self-attention convolutional neural network (SACNN) and added it to the neural network, achieving good recognition results on the MK-D2 agricultural disease dataset. (Chen et al., 2021) proposed an attention module (LSAM) for MobileNet V2, effectively enhancing the network's recognition capability for diseases. (Liao et al., 2023) optimized the network by combining ResNet-50, long short-term memory (LSTM) network, and SE-Net attention mechanism. (Tang L. et al., 2023) proposed a ternary parallel attention module based on the CBAM attention mechanism, combined with a multi-scale hybrid model composed of Inception modules and ResNext modules, achieving good results in the identification of apple leaf diseases. Additionally, some scholars have integrated deep learning network models with robotics technology. For instance, (Wang et al., 2023) integrated a visual system and a robotic intelligent control system, enabling positioning for lychee harvesting and obstacle recognition and avoidance.

In tomato disease identification, there are also numerous research achievements. (Mokhtar et al., 2015) used Gabor wavelet transform to extract image features and then used support vector machines to identify tomato leaf diseases. (Anandhakrishnan and Jaisakthi, 2022) improved the LeNet5 network model architecture, combining support vector machines (SVM) and multilayer perceptron to detect tomato diseases. (Ullah et al., 2023) used a hybrid network approach, combining EfficientNetB3 and MobileNet into the EffiMob-Net multi-scale model to detect tomato leaf diseases. (Zhou et al., 2021) introduced dense network connections into the residual network model, forming a hybrid network model that improved recognition accuracy while achieving model lightweighting. (Zaki et al., 2020) used the PCBAM attention mechanism and Dense Inception convolution blocks to optimize the MobileNet model. (Zhang et al., 2023) proposed a multi-channel automatic direction recursive attention network (M-AORANet) to address noise issues in tomato disease images, effectively achieving disease recognition, although some difficulties remained with other crop diseases. (Chen et al., 2020) combined the ResNet-50 network model with the proposed dual-channel residual attention network model (B-ARNet) to enhance the model's recognition of tomato diseases from multiple scales. (Zhao et al., 2021) optimized the ResNet50 model by using the SE-Net attention mechanism and combining it with a multi-scale feature extraction module to recognize tomato diseases.

Neural network models are effective for agricultural plant disease identification, and many new and original network structures have emerged in recent years. These network model structures can improve the recognition effect of the model by combining image enhancement algorithms, attention mechanisms and fusion methods. However, due to the uneven spatial distribution of the characteristics of agricultural diseases and the influence of different stages of their onset, the problems of small disease characteristics, large differences in similar characteristics, and small differences in heterogeneous characteristics lead to the difficulty of achieving high precision and lightweight of the model. Therefore, the purpose of this paper is to propose a high-precision detection method with limited computing resources, which can meet the accuracy requirements and be suitable for mobile deployment. The specific contributions of this article are as follows:

Image enhancement technology was used to enhance the detailed features of leaf disease images. Initially, the piecewise linear transformation method was used to remap the brightness values of the image by setting thresholds for minimum and maximum brightness, enhancing the detailed features, and helping the neural network model extract abstract features during training.

A lightweight CNN neural network model, LDAMNet, is proposed, which is mainly composed of a double attention convolution (DAC) block with mixed-channel attention (HCA) and coordinate space attention (CSA) functions. Set the number of blocks by pressing [1,1,3,1] in four different stages.

Considering the influence of noise labels in CNN model training, the Cross-Entropy loss function is improved, and the Robust Cross-Entropy Loss (RCE) loss function is derived by introducing a weighted formula with two adjustable parameters, α and β, which enhances the ability of the model to deal with label noise. In addition, by adjusting these two parameters, the loss value in model training can be flexibly adjusted.

Finally, by comparing different CNN models, different blocks, different normalization methods, as well as ablation experiments and experiments using different datasets, the effectiveness of the proposed method for identifying tomato leaf disease is proved with limited computing resources, and the recognition ability is on par with that of large-scale models, which has obvious advantages compared with existing lightweight models.

The structure of this paper is summarized as follows: the second part mainly introduces the methods used in this paper, including LDAMNet, DAC block, HCA channel attention mechanism, CSA spatial attention mechanism, and RCE loss function. Subsequently, the third part is mainly used to test the detection method proposed in this paper and evaluate the performance of the proposed identification method in all aspects through five different experiments. Finally, the fourth part mainly summarizes the work and experimental conclusions of this paper.

2 Materials and methods

2.1 Image preprocessing

2.1.1 Sample

The tomato image dataset used in this study is derived from the Plant Disease Classification Merged Dataset published on the Kaggle platform (https://www.kaggle.com/datasets/alinedobrovsky/plant-disease-classification-merged-dataset).

The dataset combines 14 existing agricultural imagery datasets covering 88 disease categories affecting 23 different crops. In this paper, the tomato leaf disease images were selected as the dataset for the study, including ten disease images at different disease stages. The image size in the dataset is 256×256, the leaf samples are shown in Figure 1, and the disease characteristics are shown in Supplementary Table S2.

2.1.2 Image processing

Training data plays a vital role in the performance of CNN models, which directly affects the training effect of model training. The process of image acquisition is usually affected by the image acquisition equipment and environment, resulting in problems such as inconsistent brightness and noise. These issues can obscure image features, hinder the model's ability to discriminate features during training, and ultimately impair its ability to recognize. In addition, due to the different number of disease samples, there are large differences in the number of images of different categories in the image dataset. This will cause the model to tend to the category with a large number of images during the training process, and the images of other categories cannot be effectively recognized, resulting in overfitting (Kong et al., 2021).

To address the aforementioned issues, this paper proposes an image processing method. In this method, images from the dataset are first decomposed into red, green, and blue color channels. Then, based on the set thresholds a and b, as well as the range 0-255, the

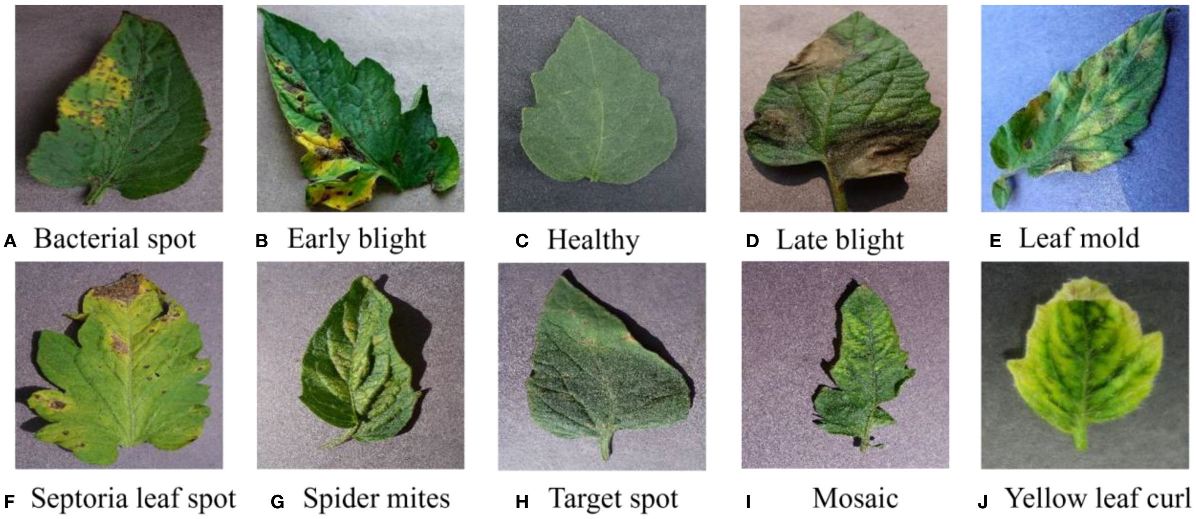


FIGURE 1
Different categories of leaf images in the tomato dataset. From **(A–J)**, it includes 10 types of disease and healthy leaf image samples.

pixels in each channel are divided into three intervals. The pixel values in different intervals are processed according to Equation 1 to obtain the processed pixel value $F(x)$. Finally, the three processed color channels are recombined to obtain enhanced disease image samples. Moreover, data imbalance is a crucial factor affecting the training effectiveness of deep learning network models. Network models tend to overly learn features from categories with more samples and struggle to effectively classify categories with fewer samples. To address this, this paper uses oversampling to balance the samples in the image dataset. The effects of image enhancement and oversampling are shown in Figure 2 The enhanced image dataset is divided into training and test sets at an 8:2 ratio, with 15975 images for training and 3994 for testing. Table 1 shows the sample distribution before and after data augmentation.

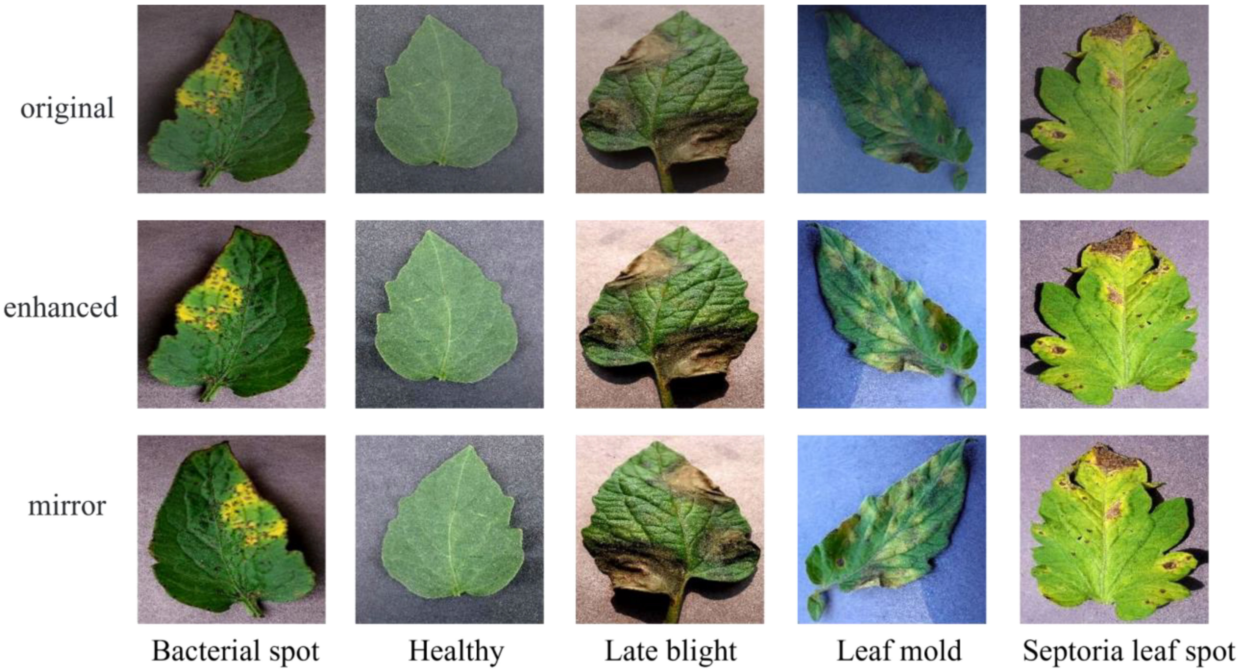


FIGURE 2
Image-enhanced sample display. Original represents the images in the original dataset, and enchanced and mirror represent enhanced and oversampled image samples, respectively.

TABLE 1 Tomato leaf disease dataset sample size.

Categories	Before	After	train	test
Bacterial spot	1612	1969	1575	394
Early blight	1000	2000	1600	400
Healthy	1251	2000	1600	400
Late blight	1209	2000	1600	400
Leaf mold	952	2000	1600	400
Septoria leaf spot	1379	2000	1600	400
Spider mites	1257	2000	1600	400
Target spot	988	2000	1600	400
Mosaic	373	2000	1600	400
Yellow leaf curl	1849	2000	1600	400

$$F(x) = \begin{cases} 0 & x < a \\ \frac{255}{(b-a)(x-a)} & a < x < b \\ 255 & x > b \end{cases} \quad (1)$$

2.2 LDAMNet model

The LDAMNet model proposed in this paper is used for tomato leaf disease identification and is composed of DN block, DAC block, D block, and Classifier. In model training, the DN block and D block are used to downsample the input image, the DAC block extracts the features of the input image, and finally, the Classifier implements the classification of the image. The DN block is composed of a convolutional layer with a convolutional kernel of 4×4 and GN, which mainly processes the input image to reduce the size of the image and extracts features in a large range by using the convolutional kernel of 4×4. The D block is composed of a convolutional layer with GN and a convolutional kernel of 2×2, which is smaller than the DN block, which makes the network model pay more attention to the local features of the image. The DAC module allocates the number of blocks in four stages according to the quantities 1, 1, 3, 1, and its structure is shown in Figure 3 and Table 2.

2.3 Dual attention convolution block

To enhance the extraction of complex image features, this paper improves the existing inverted bottleneck block and proposes a Dual Attention Convolution (DAC) block. It consists of an Inverted Bottleneck Attention (IBA) block and Coordinate Space Attention (CSA), as shown in Figure 3 The IBA block mainly comprises two pointwise convolution layers, a 3×3 depthwise convolution layer, Hybrid Channel Attention (HCA), ReLU6, and two GroupNorm layers. CSA is a lightweight coordinate space attention mechanism proposed in this paper to locate regions of interest in images. As shown in Figure 4C, the IBA module draws inspiration from the inverted bottleneck module, ConvNeXt V2 module, and Channel Attention module (CBAM), focusing primarily on processing input image channels to enrich disease feature representation.

2.3.1 Inverted bottleneck attention block

The inverted bottleneck block, first applied in MobileNet V2, serves as an optimization method for traditional convolution layers, effectively reducing the computational and parameter requirements for model training. In the inverted bottleneck block, the number of image channels and image size remain unchanged, allowing for

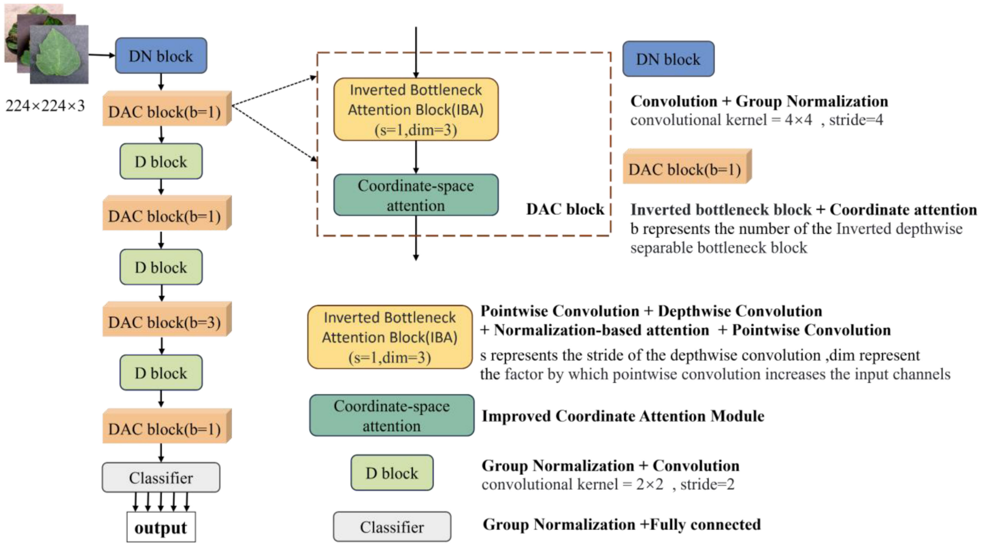

224×224×3
DN block
DAC block(b=1)
D block
DAC block(b=1)
D block
DAC block(b=3)
D block
DAC block(b=1)
Classifier
output
Inverted Bottleneck Attention Block(IBA) (s=1,dim=3)
Coordinate-space attention
DAC block
DN block
Convolution + Group Normalization
convolutional kernel = 4×4 , stride=4
DAC block(b=1)
Inverted bottleneck block + Coordinate attention
b represents the number of the Inverted depthwise separable bottleneck block
Inverted Bottleneck Attention Block(IBA) (s=1,dim=3)
Pointwise Convolution + Depthwise Convolution + Normalization-based attention + Pointwise Convolution
s represents the stride of the depthwise convolution ,dim represent the factor by which pointwise convolution increases the input channels
Coordinate-space attention
Improved Coordinate Attention Module
D block
Group Normalization + Convolution
convolutional kernel = 2×2 , stride=2
Classifier
Group Normalization +Fully connected

FIGURE 3
The structure of the LDAMNet model. b, s, and dim represent the number of IBA blocks, the stride of deep convolution, and the multiple of IBA block expansion image channels, respectively.

TABLE 2 Architectures for LDAMNet.

output size	layer name	LDAMNet

56×56	DN block	Conv(4×4, 32)
		GroupNorm(channel = 32)

	DAC block	IBA block × 1 CSA block × 1
28×28	D block	GroupNorm(channel = 32) Conv (2×2, 64)
	DAC block	IBA block × 1 CSA block × 1
14×14	D block	GroupNorm(channel = 64) Conv (2×2, 128)
	DAC block	IBA block × 3 CSA block × 1
7×7	D block	GroupNorm(channel = 128) Conv (2×2, 256)
	DAC block	IBA block × 1 CSA block × 1

1×1	Classifier	GroupNorm(channel = 256)
		Linear(256, 10)

The IBA block is shown in Figure 4B, CSA in Figure 5. The DN block and D block have strides of 4 and 2, respectively.

effective information extraction while reducing model size. The ConvNeXt V2 block improves upon the inverted bottleneck block by adding a 7×7 convolution layer before the first pointwise convolution to capture broader spatial features and mitigate the impact of complex backgrounds on model recognition performance. Additionally, it uses Global Response Normalization (GRN) instead of depthwise convolution between the two pointwise convolution layers.

This paper improves the inverted bottleneck block by placing the depthwise convolution after two pointwise convolution layers, adding an HCA module and ReLU6 activation function to enhance the model's expressiveness, and introducing GroupNorm to replace BatchNorm, reducing the model's dependence on training batch sizes. As shown in Figure 4C, after the input image passes through the first pointwise convolution and ReLU6 activation function, the input image channels are expanded according to the size of the parameter 'dim' and undergo nonlinear transformation. Then, the HCA channel attention mechanism obtains weights for the expanded channels and applies weighting. After weighing, the channels pass through a second pointwise convolution, preserving channels with rich feature expressions. Finally, a 3x3 depthwise convolution organizes the spatial information of the retained channels, facilitating subsequent CSA extraction of regions of interest in the image.

2.3.2 Hybrid channel attention

EfficientNet and ConvNeXt V2 use SE-Net and Global Response Normalization (GRN), respectively, to help models expand and integrate image dimensions and extract important image channels. This paper proposes a lightweight channel attention mechanism called Hybrid Channel Attention (HCA). This attention mechanism calculates weights for different channels in the image and applies weighting to image channels to preserve important ones during channel integration. In this paper, the HCA attention mechanism mainly consists of Nam and ECA modules. As shown in Figure 5, the input image is fed into both modules to calculate channel weights, and the resulting two channel weights are applied to the input image channels.

The Nam module uses input normalization to obtain weights for different dimensions in the image (Liu et al., 2021). This paper uses GroupNorm for calculation, as shown in Equation 2. GroupNorm is a method that groups input data based on channel dimensions and then normalize within each group. In the Nam module, mean and variance are calculated for each group and normalized to obtain W_γ.

$$GN(x) = \gamma \frac{x - \mu_x}{\sqrt{\sigma_x^2 + \varepsilon}} + \beta \tag{2}$$

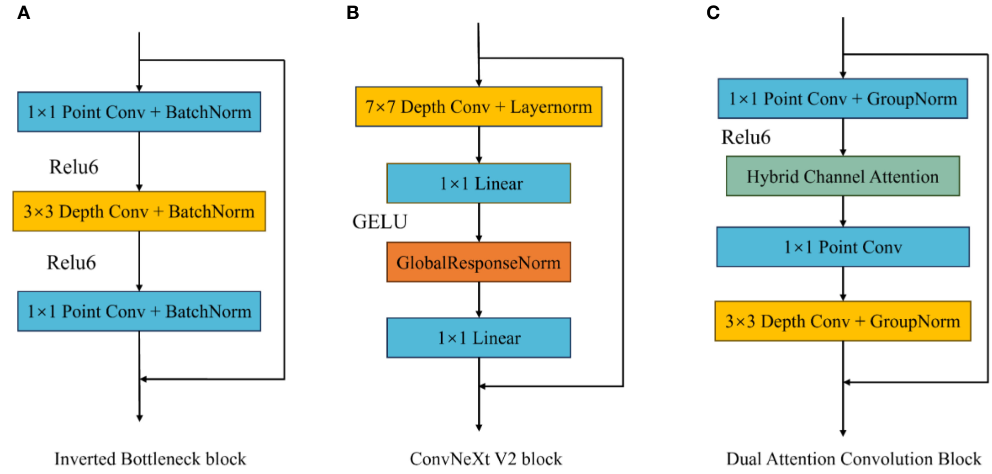

A
1×1 Point Conv + BatchNorm
Relu6
3×3 Depth Conv + BatchNorm
Relu6
1×1 Point Conv + BatchNorm
Inverted Bottleneck block
B
7×7 Depth Conv + Layernorm
1×1 Linear
GELU
GlobalResponseNorm
1×1 Linear
ConvNeXt V2 block
C
1×1 Point Conv + GroupNorm
Relu6
Hybrid Channel Attention
1×1 Point Conv
3×3 Depth Conv + GroupNorm
Dual Attention Convolution Block

FIGURE 4
Different inverted bottleneck block structures. **(A)** is the convolution block in MobileNetV2, **(B)** is the convolution block in ConvNextV2, and **(C)** is the IBA Block.

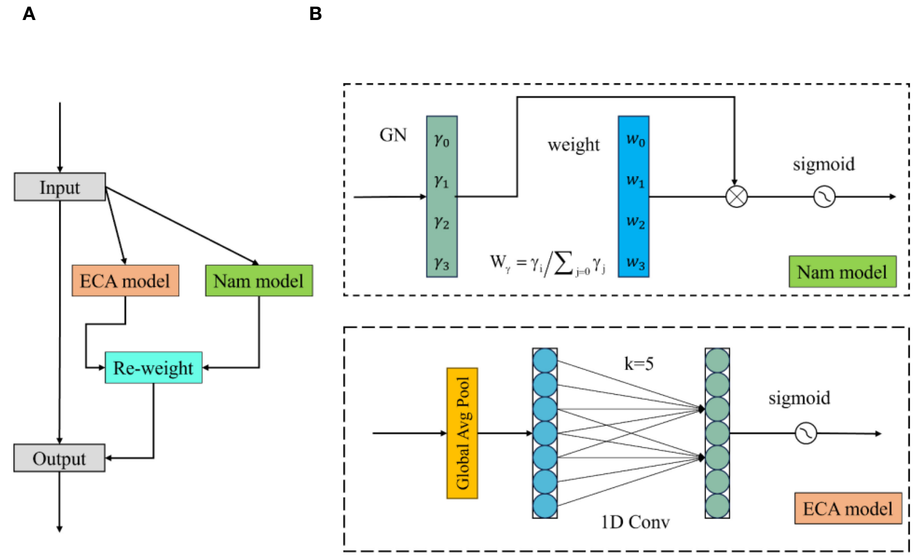


FIGURE 5
Hybrid channel attention structure. **(A)** is the HCA structure diagram and **(B)** is the NAM and ECAmodel structure diagram.

$$W_\gamma = \frac{\gamma_i}{\sum_{j=0} \gamma_j} \tag{3}$$

$$W_{nam} = sigmoid(W_g(GN(x))) \tag{4}$$

where μ_x and σ_x are the mean and variance in each specified grouping, respectively, and the γ and β are trainable affine transformations. The parameters, W_γ, are composed of the scaling factor γ of each channel and are calculated according to Equation 3. W_{nam} is the channel attention weight obtained from the Nam module, as shown in Equation 4.

The ECA module first applies global average pooling to the input image, then processes the image through a one-dimensional convolution with an adaptively adjustable kernel to generate channel weights (Wang et al., 2020). In the formula, the size of the convolution kernel k is determined by the mapping of the channel dimension. The calculation formula is as follows.

$$W_{eca} = sigmoid(C1D_k(x)) \tag{5}$$

$$k = \gamma(C) = \left| \frac{\log_2(C)}{d} + \frac{b}{d} \right|_{odd} \tag{6}$$

In Equation 5, *C1D* represents the 1-dimensional convolution processing; in Equation 6, *C* is the given channel dimension, *k* is the adaptive convolution kernel size, δ, and b are set to 2 and 1, respectively, and $|t|_{odd}$ represents the odd number closest to *t*.

After obtaining the channel attention weights of the Nam module and the ECA module, the channel attention weights *W* are obtained by combining them, and the calculation process is shown in Equation 7. Finally, the image is weighted using the resulting weight *W* input.

$$W = sigmoid(W_\gamma(GN(x))) \times sigmoid(C1D_k(x)) \tag{7}$$

2.3.3 Coordinate-space attention

In this paper, Coordinate Space Attention (CSA) is a spatial attention mechanism that mainly utilizes spatial position information to obtain weights for different regions in the image. As shown in Figure 6, the CSA attention mechanism first applies average pooling to the input image to generate horizontal and vertical feature vectors. Then, the obtained feature vectors are concatenated to form a feature map of the input image. Subsequently, dilated convolution, GroupNorm, and ReLU6 are used to enrich the feature map's expression and determine whether regions of interest exist in both directions.

The CSA module uses average pooling to obtain horizontal and vertical spatial information of the image, using two spatial pooling kernels $(H, 1)$ and $(1, W)$ to encode each channel of the input x along the horizontal and vertical coordinate directions. The height h obtained in channel c can be represented as Equation 8.

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} x_c(h,i) \tag{8}$$

Similarly, the width w obtained in the c-channel can be expressed as Equation 9.

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le i < h} x_c(h,w) \tag{9}$$

where z_c represents the encoded results of h in the horizontal direction w and vertical direction of the c-channel using average pooling, and x_c represents the eigenvalues of the c-channel in the feature map at the positions of height h and width w.

Through the calculation Formula 9, the eigenvalues along the abscissa and the ordinate can be obtained. Then, it was re-stitched to obtain a new feature image, and the feature map was processed by using a dilated convolution with a convolution kernel of 3×3. By using dilated convolutions with an expansion rate of 2, it is possible

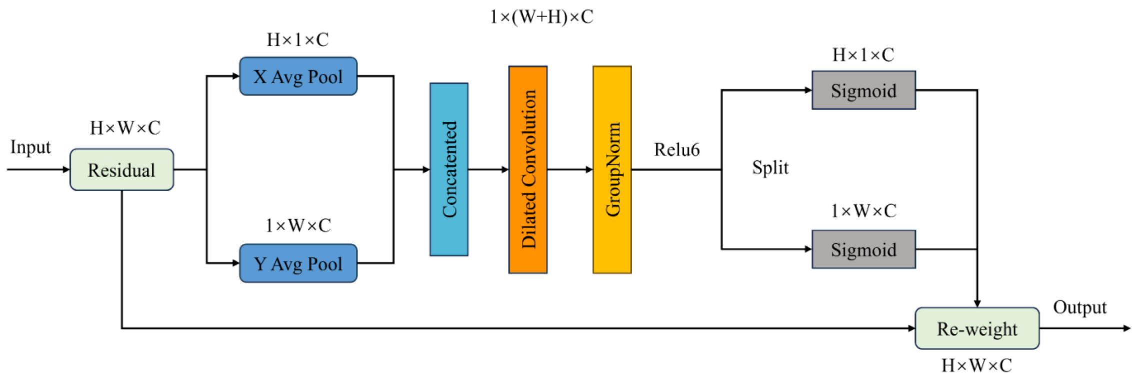


FIGURE 6
Coordinate space attention mechanism.

to increase the receptive field without increasing the computational cost of the model. The calculation is shown in Equation 10.

$$f = \delta(F^d_{3\times3}([z^h, z^w])) \tag{10}$$

where $[\cdot\ ,\ \cdot]$ Represents a concatenated operation along a spatial dimension. Where $F_{3\times3}$ is the convolutional transformation function, d is the expansion rate, δ represents the nonlinear activation function, and $f \in R^{C\times(H+W)}$ is the intermediate feature map that encodes spatial information in the horizontal and vertical directions. Then, the feature map is divided into two independent tensors along the spatial dimension to obtain $f^h \in R^{C\times H}$ and $f^w \in R^{C\times H}$. In addition, the attention weights of the tensors f^h and f^w were obtained by using the sigmoid mapping, respectively, and the calculation formula is shown in Equations 11 and 12.

$$g^h = sigmoid(F_h(f^h)) \tag{11}$$

$$g^w = sigmoid(F_w(f^w)) \tag{12}$$

Finally, the weights g^h and g^w are used to weight the input image, and the final result is shown in Equation 13.

$$y(i,j) = x_c(i,j) \times g^h_c(i,j) \times g^w_c(i,j) \tag{13}$$

where y is the final output of CSA, inspired by Coordinate attention, CSA obtains eigenvalues from the horizontal and vertical aspects of the image and processes them. It can help the model to locate the disease location of the leaf in detail. In addition, compared with Coordinate attention, CSA has a smaller number of parameters.

2.4 Robust cross-entropy loss

In CNN model training, dataset samples play a crucial role in the training of network model recognition capabilities. However, due to the constraints of environment, equipment, and other factors, the collected plant disease images may contain some data that are difficult to classify, outlier data, and mislabeled data, which will seriously affect the training effect of the model. The Cross-Entropy loss function is a loss function commonly used in classification problems to measure the difference between the model input and the actual label, and for classification problems of M categories, it can be defined as Equation 14.

$$CE(p,y) = -\sum_{c=1}^{M} y_{(o,c)} log(p_{o,c}) \tag{14}$$

where M is the total number of categories, C is the index of the category or class, O is the index of a particular sample in the dataset in the calculation, and when the loss is calculated for a particular sample, c iterates through all possible categories, from 1 to M. $y_{o,c}$ represents the true label of category C in sample O, while the model predicts the probability of category C for sample O when $p_{o,c}$. If only one category is considered per sample, the $y_{(o,c)}$ can be treated as 1, and the O index is omitted, the CE loss function can be defined as Equation 15.

$$CE(y) = -\sum_{c=1}^{M} log(p_c) \tag{15}$$

As shown in Figure 7, when the CE loss function is trained, when the accuracy is low, the loss function will give a large loss value to help the model quickly adapt to the image data. This can effectively promote the training of the network model for the dataset with correct labeling, but it will cause the network model to learn wrong data when facing the dataset with noise labels, which will not only lead to the decline of the model's recognition ability but also make the model over-adapt to the wrong labels, thereby reducing the generalization ability of the model.

To solve this problem, (Zhang and Sabuncu, 2018) proposed Generalized Cross-Entropy Loss (GCE). By introducing the parameter q, the GCE loss function can reduce the penalty by reducing the loss value when the noise is wrong, thereby increasing the tolerance of the model. When q is close to 0, the contribution of the noise label to the total loss is also limited to a small range, reducing the impact on model training. The formula is shown in Equation 16.

$$GCE(p,y) = \frac{1 - p_y^q}{q} \tag{16}$$

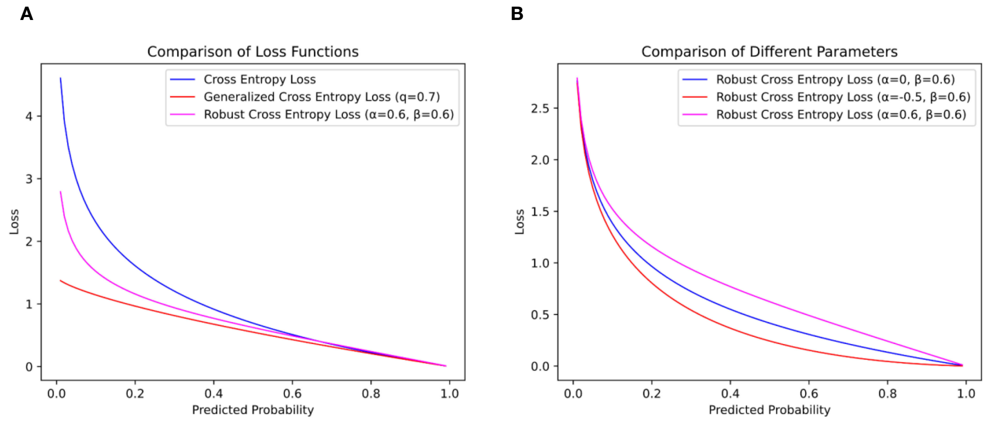


FIGURE 7
Comparison of loss function curves. **(A)** is the loss function curve of α and β 0.6 and **(B)** is the influence of different α values on the loss function curve.

Although the GCE loss function can reduce the impact of noise labels on the recognition ability of the model, it gives a small loss value when the accuracy is high, which makes the model unable to be further trained to improve the recognition ability. In this paper, to help the network model be robust to noise labels in training and improve the effect of model training, a weighted formula with adjustable parameters α and β is introduced to optimize the CE loss function, defined as Equation 17.

$$RCE(y) = \frac{1}{M}\sum_{c=1}^{M}((\alpha \times p_c + \beta) \times (-log(p_c))) \tag{17}$$

In the formula, $(\alpha \cdot p_c + \beta)$ represents the added weighting formula. In this weighting formula, α and β are two parameters. Specifically, β serves as a scaling factor that directly influences the overall loss magnitude during model training. When β is greater than 1, the RCE loss function imposes larger penalties; when β is less than 1, it imposes smaller penalties, with $\beta > 0$. On the other hand, parameter α can be used to adjust the magnitude of the loss during model training, constrained by $\alpha > -\beta$. A larger α assigns larger losses during training, while a smaller α assigns smaller losses, as depicted in Figure 7B.

In this paper, the parameters α and β of the RCE loss function are set to 0.6, as shown in Figure 7A. When α and β are 0.6, fewer loss values can be given when the model accuracy is low, and no formal distribution of data is learned, so as to reduce the penalty of the loss function on the noise label, help the model focus on the label with higher confidence, and reduce the influence of the noise label on the model training. When the progress of the model is high, a larger loss value is given, and on the basis of extracting abstract and useful information to a certain extent, the model pays more attention to the samples that are difficult to classify correctly and improves the recognition ability of the model.

2.5 Tomato disease identification model process

The overall training flow of the LDAMNet model is illustrated in Figure 8. In this study, a CNN-based deep learning method is used to construct the recognition model, which heavily relies on the training data. To address issues such as blurred disease features and insufficient image samples in the dataset, this paper first resizes the images and applies piecewise linear transformation to enhance image detail features, as shown in Figure 8. After processing the images in the dataset, they are divided into training and test sets at an 8:2 ratio. Then, to improve the model's generalization ability, normalization is applied to the images in both training and test sets to standardize pixel values across different channels. Finally, to avoid issues caused by interrupted training, this paper saves model parameter files at each training epoch to facilitate continued training. Additionally, to achieve effective disease recognition, the weights of the best-fitting model are saved during training based on set evaluation parameters for subsequent use.

3 Experimental results and discussion

3.1 Experimental design

The computer used in this paper uses the Windows 11 operating system, uses a 12th Gen Intel(R) Core(TM) i7-12700 (2.10 GHz) processor, and uses a GPU for model training and testing, and the GPU is NVIDIA GeForce RTX 3060(12G). The software environment uses Python 3.9.13, PyTorch 1.13.1, and Cuda 11.6 frameworks.

The experiment is divided into five parts. Namely, the comparative test between different network models proposed in this paper, the comparative test with the inverted bottleneck block,

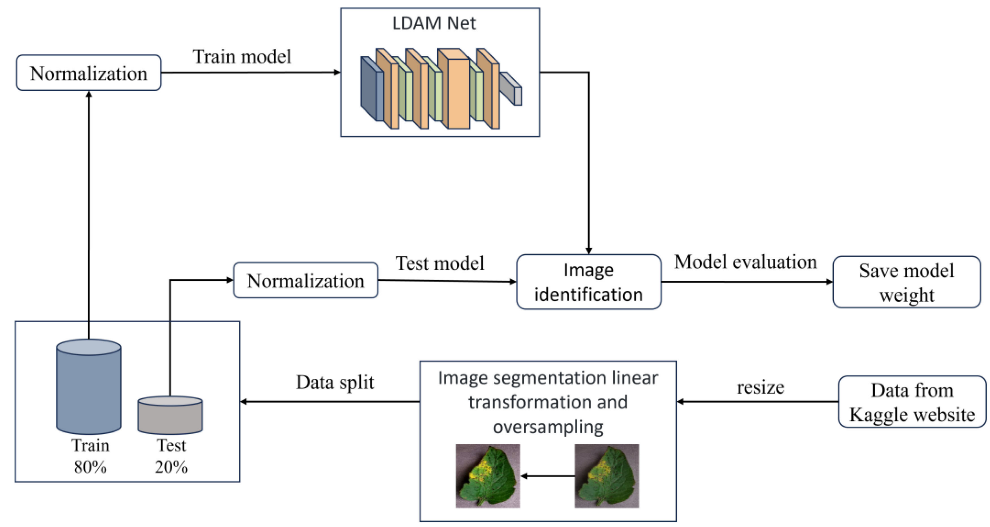


FIGURE 8
Overall flowchart of LDAMNet model training.

the comparative test using different loss functions, the ablation experiment, and the comparative test on different datasets.

In the training of the neural network model, the Adam stochastic gradient descent method was used to optimize the network model. The calculation of this algorithm is relatively simple, and it has strong adaptability to the gradient. The learning rate is set to 0.0001, the number of iterations is set to num_epochs = 100, and the number of images per batch batch_size = 16. In addition, the AutoAugment method is used to process the training set images, which can enhance the network training effect.

3.2 Evaluation indicators

In order to effectively evaluate the trained neural network model, precision, recall, accuracy, and F1 score were used to measure the performance of the neural network model in the identification of tomato leaf diseases. These parameters are calculated as shown in Equations 18–21.

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

$$Recall = \frac{TP}{TP + FN} \tag{19}$$

$$Accuray = \frac{TP + TN}{TP + FN + FP + TN} \tag{20}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{21}$$

In the formula, TP (True Positive) is the true example, which indicates the number of positive samples predicted by the model; TN (True Negative) is the true negative example, which indicates the number of negative samples predicted by the model, FP (False Positive) is a false positive example, which indicates the number of negative samples predicted by the model, and FN(False Negative) is a false negative example, which indicates the number of positive samples predicted by the model to be negative.

In this study, precision represents the proportion of samples that are correctly judged to be positive by the network model. Recall measures the proportion of positive class samples correctly identified by the network model in actual positive class samples. Precision represents the ratio of the total number of samples correctly classified by the network model to the total number of samples. The F1 value is a harmonic average of precision and recall, taking into account precision and recall, and is balanced between precision and recall.

In addition, two parameters, Flops (floating-point arithmetic) and Params (number of parameters), are introduced to evaluate the size of the network model. The larger the Flops, the more computational resources the network model needs for training and inference, and this parameter usually represents the Flops computation in a single forward propagation. Params represent the number of parameters in the model, including all weights and biases that need to be learned, and larger Params mean that the larger the network model, the more storage space is needed to hold the model weights.

3.3 Experiments of different network models

To test the performance of the LDAMNet network model, this study compares it with ConvNeXtV2 (Woo et al., 2023), Inception_Next (Yu et al., 2023), DenseNet121 (Huang et al., 2017), ResNet18 (He et al., 2016), GhostNet (Han et al., 2020), EfficientNet (Tan and Le, 2019), EfficientFormer (Li et al., 2023), MobileNet (Sandler et al., 2018), MobileVitV2 (Mehta and Rastegari, 2022), Swin Transformer V2 (Liu et al., 2022), Deit3 (Touvron et al., 2022). The models participating in the experiment were evaluated using six parameters: precision, recall, accuracy, F1 score, Flops, and Params. In the comparative experiment, to effectively detect the recognition capabilities of LDAMNet and different network models, these 11 network models were divided into three categories: large-scale CNN models, lightweight CNN models, and Vit models. The large-scale CNN models include ConvNeXt, Inception_Next, DenseNet, and ResNet; the lightweight CNN models include GhostNet, EfficientNet, and MobileNet; the Vit models include EfficientFormer, MobileVitV2, Swin Transformer V2, and Deit3. The accuracy curve comparison of these three types of models with LDAMNet is shown in Supplementary Figure S1. The network models used in this experiment are all from the Timm library, and the experimental results are shown in Table 3.

As shown in Table 3, in the comparative experiment, the average values of accuracy, precision, recall, and F1 score of the LDAMNet model are the highest among the eight network models, which are 98.71, 98.73, 98.69, and 98.71, respectively. In addition, the Flops and Params parameters of the LDAMNet model are 0.142 and 0.91, respectively, which are the smallest among the 12 models, indicating that this network model can achieve lightweight and high-precision recognition of tomato diseases.

Furthermore, the experimental results show that on the processed tomato dataset, the recognition ability of the LDAMNet model is higher than that of the other 11 network models. The recognition ability of the LDAMNet model is not significantly different from the large models in this paper; according to the evaluation parameters obtained in Table 3, the average accuracy of LDAMNet, DenseNet121, Swin Transformer, and ResNet18 can all reach more than 98%. However, the floating-point operations and parameter counts required by the LDAMNet network model are much smaller than those of the other three types of models, which proves that the LDAMNet model, as a lightweight network model, can achieve the recognition ability of large-scale CNN models or mainstream Vit models, or even slightly better.

In this experiment, the LDAMNet model was compared with three mainstream lightweight models: GhostNet, EfficientNet, and MobileNet. As shown in Supplementary Figure S1B, the recognition ability of the LDAMNet model is better than that of these three mainstream lightweight models, with the gap between the EfficientNet model and the LDAMNet model being the largest, with the four evaluation parameters being about 6.63%, 6.54%, 6.58%, and 6.56% higher, respectively.

Finally, to test the classification effects of the 12 models, a confusion matrix was used to test the models. As shown in Figure 9, BS, EB, H, LB, LM, SLS, SM, TS, M, and YLC represent ten types of leaves in the tomato image dataset used in this paper. The test dataset was obtained from the dataset in proportion, including 999 images in 10 categories. In all confusion matrix tests, only the LDAMNet proposed in this paper achieved complete recognition of the test dataset. The other 11 network models all produced a certain number of misjudgments. Among them, ConvNext, MobileNet, and MobileVit had the most misjudgments, with 8, 6, and 6, respectively. EB was the main category of incorrect recognition by the models. The main reason is the uncertain regional distribution of tomato leaf data and the similar characteristics of different types of leaf diseases. For example, both EB and BS diseases produce black or brown spots in appearance, with only slight differences in the shape and color of the spots, leading to some network models being unable to effectively extract features, causing misjudgment.

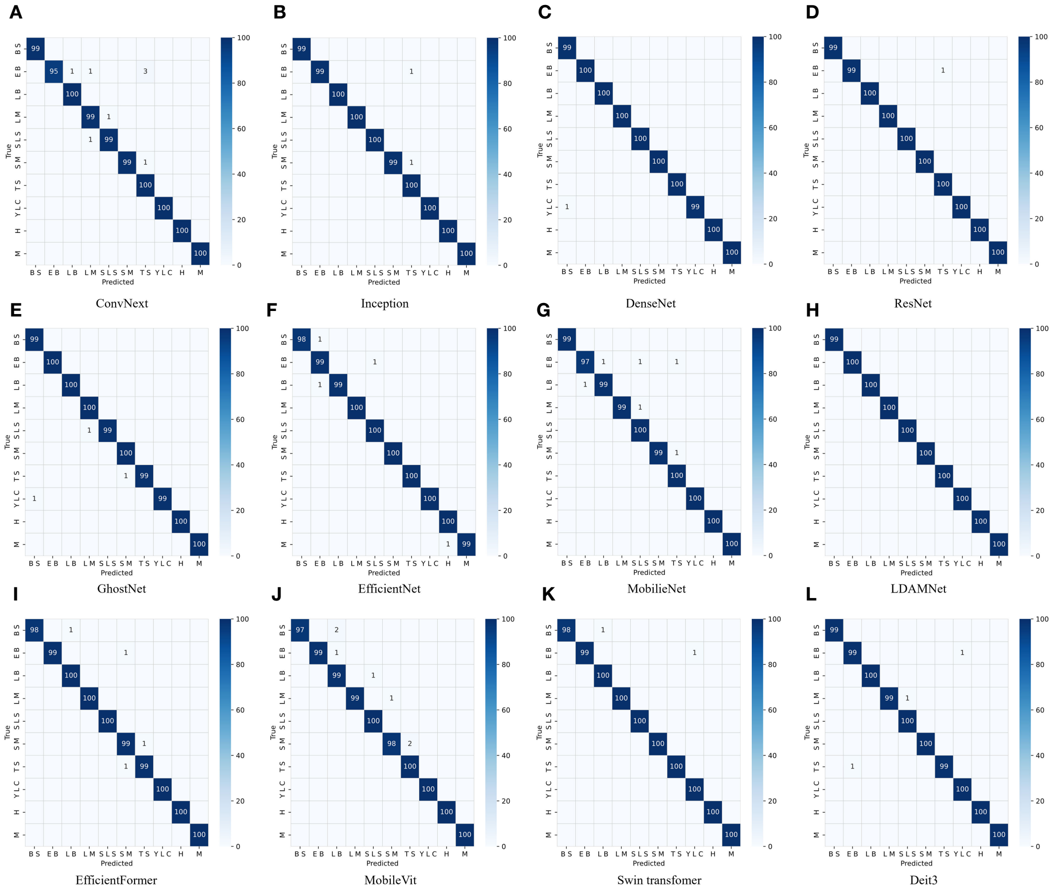
3.4 Experimental of inverted bottlenecks

In subsection 2.2.1, this paper proposes an improved inverted bottleneck block DAC block by adding a channel attention mechanism and a spatial attention mechanism, respectively. In order to verify the improvement effect, the improved inverted

TABLE 3 Comparison table of evaluation parameters obtained from training of different network models.

Model	Accuracy	Precision	Recall	F1 score	Flops(G)	Params(M)
ConvNeXt V2_T	94.87	95.02	94.83	94.92	4.45	27.79
Inception_Next_T	97.15	97.30	97.16	97.23	4.2	28.04
DenseNet121	98.64	98.69	98.67	98.68	2.83	7.89
ResNet18	98.05	98.04	97.94	97.99	1.82	11.69
GhostNet V2	94.30	94.53	94.40	94.46	0.42	11.10
EfficientNet	92.08	92.19	92.11	92.15	0.38	5.24
EfficientFormerV2	96.49	96.57	96.50	96.48	1.23	12.63
MobileNetV2	94.19	94.33	94.23	94.28	0.3	3.47
MobileVitV2	95.15	95.25	95.16	95.14	1.41	4.87
Swin TransformerV2	98.22	98.28	98.22	98.22	4.51	28.33
Deit3	97.57	97.60	97.51	97.51	4.24	21.97
LDAMNet (Proposed model)	**98.71**	**98.73**	**98.69**	**98.71**	**0.142**	**0.910**

The bold values represent the best data in the experiment, such as the best average Accuracy, the best average Precision, the best Recall, the best F1 score, the minimum Flops requirement, and the minimum Params requirement.



ConvNext Inception DenseNet ResNet GhostNet EfficientNet MobilieNet LDAMNet EfficientFormer MobileVit Swin transfomer Deit3

FIGURE 9
Confusion matrix testing for different CNN models. Models from **(A)** to **(L)** are the models for comparative experiments in Table 4, respectively.

bottleneck block (CIB block) in ConvNeXt V2 and the inverted bottleneck block (IB block) used in MobileNet V2 were used for experimental comparison in this experiment, and the three-block structures are shown in Figure 4. Table 4 shows the parameters of the LDAMNet model trained with three blocks, among which the model with DAC block has the highest evaluation parameters, which are 3.34, 3.33, 1.09 and 1.09 larger than the lowest CIB block, respectively. In addition, among the three types of block, most of the parameters required for training are made with the IB block, followed by the CIB block, and finally, the DAC block. The results show that compared with the IB block and CIB block, the DAC block can achieve higher recognition ability with fewer computing resources.

CIB block, as an improved method of IB block, although the parameters of IB block are effectively reduced by reducing the normalization method and placing the convolutional layer in front of the whole channel, it cannot effectively extract image features in the face of tomato disease image dataset due to its use of 7×7 convolution kernel. The reason is that some diseases in the tomato disease image show small local regions, and although the 7×7 convolutional kernel can obtain the association of regions in space through the receptive field method, it will also lead to inaccurate information acquisition in local regions, which leads to the inferior recognition ability of CIB block in this dataset. However, the IB block and DAC block using the 3×3 convolutional kernel can fully extract the local features of the image so that the recognition ability of the two blocks is similar.

TABLE 4 Parameters were evaluated using the IB, CIB, and DAC block.

Methods	Accuracy	Precision	Recall	F1 score	Flops(G)	Params(M)
IB block	97.57	97.64	97.60	97.62	0.288	1.836
CIB block	95.37	95.49	95.36	95.42	0.189	1.176
DAC block	**98.71**	**98.73**	**98.69**	**98.71**	**0.142**	**0.910**

The bold values represent the best data in the experiment, such as the best average Accuracy, the best average Precision, the best Recall, the best F1 score, the minimum Flops requirement, and the minimum Params requirement.

Figure 10 shows the feature map of the LDAMNet model using different blocks in four stages. In the first three stages, the CIB block and IB block can better preserve the image outline than the DAC block. However, in the fourth stage, the output feature map contains fewer abstract features than the DAC block, and even some feature maps do not contain image features. As a result, the classifier of the LDAMNet model using CIB block and IB block cannot discriminate the input graph with missing features, which affects the recognition ability of the LDAMNet model.

3.5 Experiment of normalization methods

In this paper, in order to reduce the influence of different batches in model training, the GN normalization method is used instead of the BN normalization method commonly used in convolutional neural networks. In addition, in order to verify the optimization effect of the LDAMNet network using the GN method, in this experiment, four normalization methods were used: (Wu and He, 2020), BN (Ioffe and Szegedy, 2015), IN (Ulyanov et al., 2016), and LN (Ba et al., 2016), respectively, in Batch size=8, Batch size=16, and Batch size=32 cases to train the network model.

Supplementary Figure S2 shows the transformation of the accuracy curve of the LDAMNet network model trained using four normalization methods: GN, BN, LN, and IN. Among the four normalization methods, the accuracy curves using the IN and LN normalization methods fluctuated greatly with different batches. However, the accuracy curve using the GN and BN normalization methods is more stable in the three cases. In addition, as shown in Table 5, the GN normalization method can achieve the highest accuracy in the three cases with different batch sizes, while the BN normalization method is slightly lower. Among them, the maximum accuracy difference between GN and BN is 1.05% when Batch size = 16, and the minimum accuracy difference is 0.36% when Batch size = 32.

3.6 Ablation experiments

In this section, we conducted ablation experiments, comparison experiments of different attentional mechanisms, and experiments using different loss functions for training. As shown in Supplementary Figure S3A, among the three improvements of HCA, CSA, and DAC, the DAC block has the greatest improvement in the recognition performance of the network, followed by the HCA and CSA modules. As shown in Table 6, the DAC block, which aggregates CSA and HCA in the lightweight LDAMNet model, can effectively help the network model improve the recognition accuracy of the disease, and its average accuracy can reach 98.71%. The average recognition accuracy of the HCA block and CSA block is 96.21% and 95.89% respectively, which indicates that both of them can effectively improve the recognition ability of the model in LDAMNet, with the attention effect of HCA being slightly better.

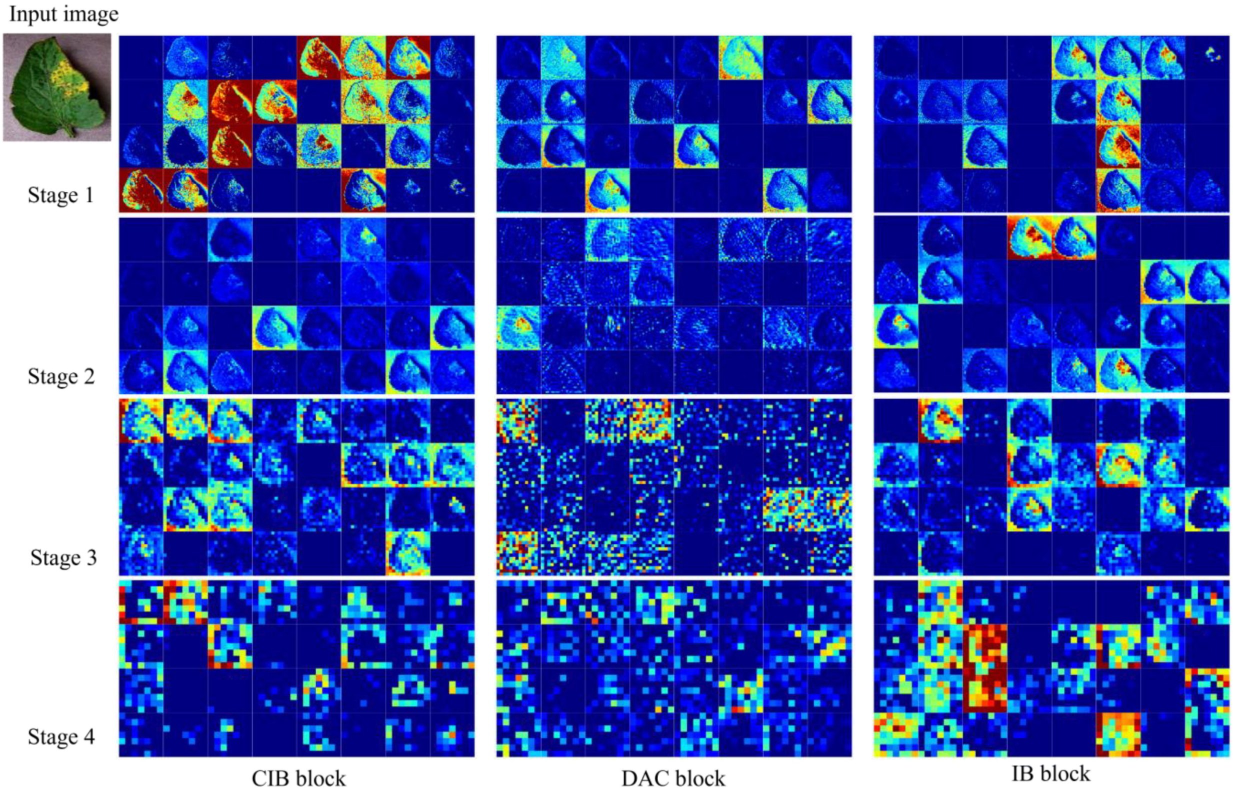


FIGURE 10
Characteristic diagram of the LDAMNet network at different stages using three blocks.

TABLE 5 Accuracy values of different normalization methods under different batch sizes.

Batch size	Methods	Accuracy	Precision	Recall	F1 score
8	LN	96.69	96.84	96.67	96.75
	IN	96.22	96.30	96.20	96.25
	BN	97.68	97.81	97.65	97.73
	GN	**98.49**	**98.64**	**98.51**	**98.57**
16	LN	96.28	96.48	96.34	96.41
	IN	96.34	96.52	96.35	96.43
	BN	97.66	97.75	97.63	97.69
	GN	**98.71**	**98.73**	**98.69**	**98.71**
32	LN	96.66	96.87	96.63	96.75
	IN	96.03	96.25	96.02	96.13
	BN	97.88	97.94	97.89	97.91
	GN	**98.24**	**98.38**	**98.25**	**98.31**

The bold values represent the best data in the experiment, such as the best average Accuracy, the best average Precision, the best Recall, the best F1 score, the minimum Flops requirement, and the minimum Params requirement.

Then, to examine the difference between the DAC block proposed in this paper and mainstream attention mechanisms, CA and CBAM were introduced for comparison experiments. Supplementary Figure S3B shows the variation of the experimental accuracy curves, in which the CA and CBAM attention mechanisms have some fluctuations in their accuracy curves during the training cycle, while the DAC accuracy curve is relatively smooth. The test data, as shown in Table 6, show that there is no significant difference among the three methods in terms of the amount of computation and the number of parameters required, while the average accuracy of the DAC block method is slightly higher than that of the two attention mechanisms, CA and CBAM. Figure 11 shows the class activation diagrams of the LDAMNet network model using different attention mechanisms, and the input images are the four leaf disease images in Figure 1. From the figure, it is clearly observed that the DAC block effectively captures the leaf disease regions at different locations, whereas HCA, CSA, CA, and CBAM attention mechanisms do not capture the regions as accurately as the DAC block.

Finally, in this section, to validate the RCE loss function proposed in this paper, the mainstream CE loss function is used for comparison. Supplementary Figure S3C shows the comparison of the accuracy curves of LDAMNet using the RCE loss function and the CE loss function, respectively. The LDAMNet model applying the RCE loss function does not have the effect of too small loss values in the pre-training period, which leads to slower convergence, and its accuracy curve is more stable in the late training period.

3.7 Cross-dataset experiments

Through the above experiments, it can be proved that the model proposed in this paper has a strong recognition ability in the tomato dataset. However, it is unknown whether the model can have the same advantages in the face of different leaf disease datasets. Therefore, in this experiment, in order to test the recognition ability of the LDAMNet network model in the face of different leaf disease images, the model was trained and tested using Rice Leaf Disease Images with complex backgrounds, and the samples of the rice dataset are shown in Supplementary Figure S4, including Bacterialbight, Blast, Brownsport, Tungro has a total of 5932 images (https://www.kaggle.com/datasets/nirmalsankalana/rice-leaf-disease-image) in four categories.

TABLE 6 Comparative experiments of different structures and training methods of network models.

Settings	Accuracy	Precision	Recall	F1 score	Flops(G)	Param(M)
Baseline	95.19	95.31	95.18	95.24	**0.1418**	**0.9057**
+HCA	96.21	96.31	96.18	96.24	0.1425	0.9057
+CSA	95.89	95.97	95.88	95.92	0.1419	0.9105
+CA	97.98	98.03	97.92	97.97	0.1422	0.9385
+CBAM	97.34	97.41	97.31	97.36	0.1424	0.9279
+DAC(CE)	98.15	98.27	98.20	98.23	0.1426	0.9105
+DAC(RCE)	**98.71**	**98.73**	**98.69**	**98.71**	0.1426	0.9105

The bold values represent the best data in the experiment, such as the best average Accuracy, the best average Precision, the best Recall, the best F1 score, the minimum Flops requirement, and the minimum Params requirement.

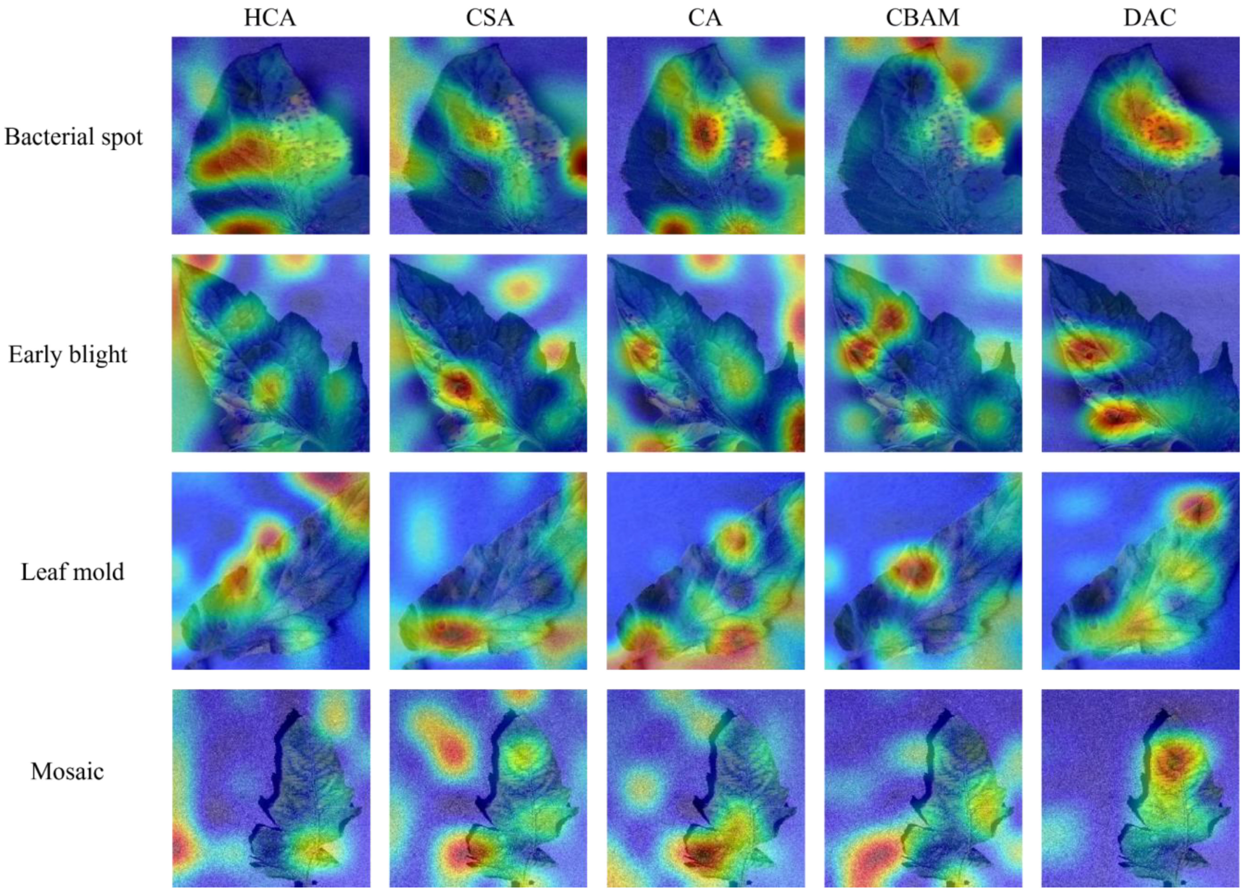


FIGURE 11
Category activation diagram using different attention mechanisms in the face of four different tomato diseases.

In order to detect the gap between the recognition ability of the LDAMNet model in this dataset and the current mainstream models, the seven models used in Part 3.3 were used in this experiment. In the experiment, set the Batch Size to 16, the training round epoch to 100, and the learning rate to 0.00001. Table 7 lists the number of datasets. In addition, ConvNeXt, Inception, DenseNet, ResNet, GhostNet, EfficientNet, and MobileNet were trained using the CE loss function, and LDAMNet was trained using the RCE loss function, and the evaluation parameters obtained from the test are shown in Table 8, and the change of the accuracy curve is shown in Figure 12.

The measured data are shown in Table 8, and the highest scores of the Accuracy, Recall, and F1 scores of the model are 98.56, 98.58, and 98.65, respectively, while the Precision parameter of the ConvNeXt model achieves the highest value of 98.71, which is slightly higher than the 98.70 of the LDAMNet model. The measured data show that LDAMNet can still maintain the same recognition ability as the existing mainstream large-scale models after replacing it with the rice dataset and can also maintain certain advantages compared with the lightweight model.

Figure 12 shows the accuracy curves of the different models in the experiment. As shown in the figure, the recognition accuracy convergence speed of the proposed model in the early stage of training is relatively slow and fluctuates to a certain extent. However, with further training of the model, the recognition accuracy of the LDAMNet model can be stabilized in a high region. The results show that the network model proposed in this paper can still maintain high recognition performance in the face of cross-dataset and has a certain generalization.

4 Conclusion

This paper addresses the issues of uneven distribution of disease features in tomato leaf images, significant differences within similar features, and small differences between dissimilar features. A high-

TABLE 7 Number of samples from the training and test sets of rice image datasets without data augmentation.

Categories	Bacterial blight	Blast	Brownsport	Tungro
Train	1268	1152	1280	1046
Test	316	288	320	262

TABLE 8 Comparison of evaluation parameters obtained by the model trained using the rice dataset.

	Model	Accuracy	Precision	Recall	F1 score	Flops(G)	Params(M)
1	Connect V2_T	98.44	**98.71**	98.46	98.58	4.45	27.79
2	Inception_Next_T	96.10	96.35	96.20	96.27	4.2	28.04
3	DenseNet121	98.52	98.51	98.51	98.51	2.83	7.89
4	ResNet18	97.16	97.28	97.19	97.24	1.82	11.69
5	GhostNet V2	97.95	97.95	97.95	97.95	0.42	11.10
6	EfficientNet	95.56	95.47	95.53	95.50	0.38	5.24
7	MobileNetV2	90.44	90.40	90.59	90.49	0.3	3.47
8	LDAMNet (Proposed model)	**98.56**	98.70	**98.58**	**98.63**	**0.142**	**0.910**

The bold values represent the best data in the experiment, such as the best average Accuracy, the best average Precision, the best Recall, the best F1 score, the minimum Flops requirement, and the minimum Params requirement.

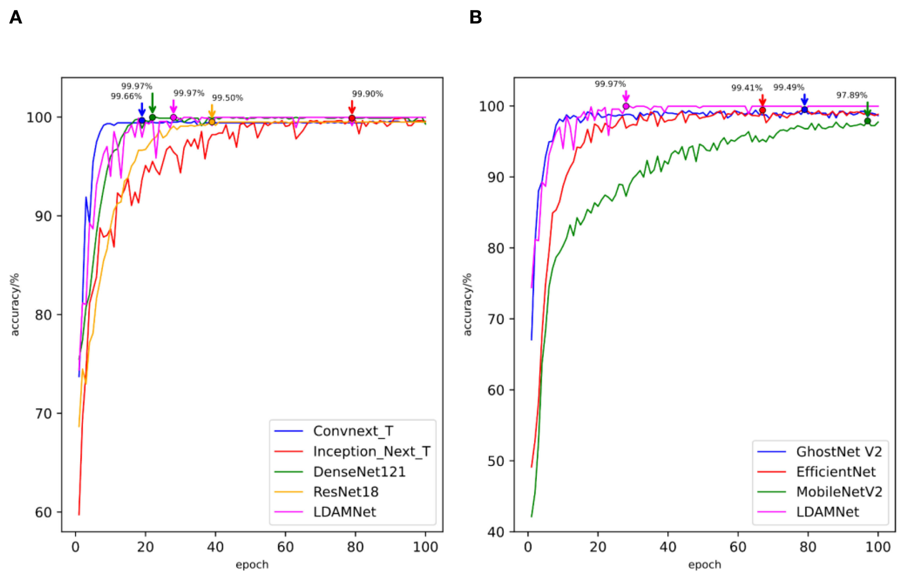


FIGURE 12
The accuracy curves of different models trained using rice datasets. **(A)** comparison of the accuracy curves of large-scale models **(B)** and comparison of the accuracy curves of lightweight models.

precision and lightweight leaf disease recognition method has been designed. First, linear transformation is used to enhance the image, augmenting the detail features of the disease and mitigating the problems of significant differences within similar features and small differences between dissimilar features. Then, DAC block, composed of HCA, CSA, and IBA blocks, is used to build a lightweight network model called LDAMNet. Additionally, the RCE loss function is employed to train the model, increasing its robustness. Comprehensive testing shows that this method can effectively identify tomato leaf diseases, offering certain advantages over mainstream large-scale and lightweight models, with maximum accuracy, precision, recall, and F1 scores reaching 99.88, 99.88, and 99.87, respectively. This confirms that LDAMNet achieves high-precision disease recognition while being a lightweight model.

Moreover, to verify the generalization of this detection method, a rice disease dataset was used for testing. Experimental results indicate that the proposed method still maintains certain advantages and can be used for cross-dataset disease recognition. Although LDAMNet achieves high-precision disease recognition, it still has potential for further exploration. Its average recognition accuracy on the rice disease dataset has not reached the optimum level. Further improvements are needed to address the issue of uneven distribution of disease features in complex backgrounds.

In summary, this paper proposes a method for detecting tomato leaf diseases and establishes a new lightweight convolutional neural network model, LDAMNet. Tests have shown that this model can effectively identify tomato leaf diseases and maintain strong recognition capability even in the complex backgrounds of the rice disease dataset. The proposed method can effectively identify agricultural leaf diseases, providing a feasible approach for early identification and reasonable treatment of agricultural diseases.

Data availability statement

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

Author contributions

EZ: Writing – original draft. NZ: Writing – review & editing. FL: Resources, Writing – original draft. CL: Visualization, Writing – original draft.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

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

Publisher's note

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

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2024.1420584/full#supplementary-material

References

Affonso, C., Rossi, A. L. D., Vieira, F. H. A., and de Carvalho, A.C.P.d. (2017). Deep learning for biological image classification. *Expert Syst. Appl.* 85, 114–122. doi: 10.1016/j.eswa.2017.05.039

Akbarzadeh, S., Paap, A., Ahderom, S., Apopei, B., and Alameh, K. (2018). Plant discrimination by Support Vector Machine classifier based on spectral reflectance. *Comput. Electron. Agric.* 148, 250–258. doi: 10.1016/j.compag.2018.03.026

Anandhakrishnan, T., and Jaisakthi, S. M. (2022). Deep Convolutional Neural Networks for image based tomato leaf disease detection. *Sustain. Chem. Pharm.* 30, 100793. doi: 10.1016/j.scp.2022.100793

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv 1607.06450.* doi: 10.48550/arXiv.1607.06450

Bhatkar, N. S., Shirkole, S. S., Mujumdar, A. S., and Thorat, B. N. (2021). Drying of tomatoes and tomato processing waste: a critical review of the quality aspects. *Dry. Technol.* 39, 1720–1744. doi: 10.1080/07373937.2021.1910832

Cai, C., Wang, Q., Cai, W., Yang, Y., Hu, Y., Li, L., et al. (2023). Identification of grape leaf diseases based on VN-BWT and Siamese DWOAM-DRNet. *Eng. Appl. Artif. Intell.* 123, 106341. doi: 10.1016/j.engappai.2023.106341

Chen, J., Zhang, D., Zeb, A., and Nanehkaran, Y. (2021). Identification of rice plant diseases using lightweight attention networks. *Expert Syst. Appl.* 169, 114514. doi: 10.1016/j.eswa.2020.114514

Chen, X., Zhou, G., Chen, A., Yi, J., Zhang, W., and Hu, Y. (2020). Identification of tomato leaf diseases based on combination of ABCK-BWTR and B-ARNet. *Comput. Electron. Agric.* 178, 105730. doi: 10.1016/j.compag.2020.105730

Deng, J.-s., Huang, W.-q., Zhou, G.-x., Hu, Y.-h., Li, L.-j., and Wang., Y.-f. (2023). Identification of banana leaf disease based on KVA and GR-ARNet1. *J. Integr. Agricult.* doi: 10.1016/j.jia.2023.11.037

Eli-Chukwu, N., and Ogwugwam, E. (2019). Applications of artificial intelligence in agriculture: A review. *Eng. Technol. Appl. Sci. Res.* 9, 4377–4383. doi: 10.48084/etasr.2756

Gnanasekaran, S., and Opiyo, G. (2020). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egypt. Inf. J.* 22.

Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., and Xu, C. (2020). "GhostNet: more features from cheap operations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1577-1586. doi: 10.1109/CVPR42600.2020

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

Hou, Q., Daquan, Z., and Feng, J. (2021). Coordinate attention for efficient mobile network design. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 13708-13717. doi: 10.1109/CVPR46437.2021.01350

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7132-7141. doi: 10.1109/CVPR.2018.00745

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)*, 2261-2269. doi: 10.1109/CVPR35066.2017

Ioffe, S., and Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. *abs/1502.03167.*

Kamal, K. C., Yin, Z., Wu, M., and Wu, Z. (2019). Depthwise separable convolution architectures for plant disease classification. *Comput. Electron. Agric.* 165, 104948. doi: 10.1016/j.compag.2019.104948

Khan, M. A., Akram, T., Sharif, M., Awais, M., Javed, K., Ali, H., et al. (2018). CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features. *Comput. Electron. Agric.* 155, 220–236. doi: 10.1016/j.compag.2018.10.013

Kong, J., Wang, H., Wang, X., Jin, X.-b., Fang, X., and Lin, S. (2021). Multi-stream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture. *Comput. Electron. Agric.* 185, 106134. doi: 10.1016/j.compag.2021.106134

Li, Y., Hu, J., Wen, Y., Evangelidis, G., Salahi, K., Wang, Y., et al. (2023). "Rethinking vision transformers for mobileNet size and speed," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 16843-16854. doi: 10.1109/ICCV51070.2023.01549

Liao, F.-b., Feng, X., Li, Z.-q., Wang, D.-y., Xu, C.-m., Chu, G., et al. (2023). A hybrid CNN-LSTM model for diagnosing rice nutrient levels at the rice panicle initiation stage. *J. Integr. Agric.* 23. doi: 10.1016/j.jia.2023.05.032

Liu, Y., Shao, Z., Teng, Y., and Hoffmann, N. (2021). NAM: normalization-based attention module. *abs/2111.12419.*

Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., et al. (2022). "Swin transformer V2: scaling up capacity and resolution," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11999-12009. doi: 10.1109/CVPR52688.2022.01170

Mehta, S., and Rastegari, M. (2022). Separable self-attention for mobile vision transformers. *ArXiv abs/2206.02680.*

Mitchell, A. E., Hong, Y.-J., Koh, E., Barrett, D. M., Bryant, D. E., Denison, R. F., et al. (2007). Ten-year comparison of the influence of organic and conventional crop management practices on the content of flavonoids in tomatoes. *J. Agric. Food Chem.* 55, 6154–6159. doi: 10.1021/jf070344+

Mokhtar, U., Ali, M. A. S., Hassenian, A. E., and Hefny, H. (2015). "Tomato leaves diseases detection approach based on Support Vector Machines," in *2015 11th International Computer Engineering Conference (ICENCO)*, 246-250. doi: 10.1109/ICENCO.2015.7416356

Patil, S. S., and Thorat, S. A. (2016). "Early detection of grapes diseases using machine learning and IoT," in *2016 Second International Conference on Cognitive Computing and Information Processing (CCIP)*, 1-5. doi: 10.1109/CCIP.2016.7802887

Prathibha, S. R., Hongal, A., and Jyothi, M. P. (2017). "IOT based monitoring system in smart agriculture," in *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, 81-84. doi: 10.1109/ICRAECT.2017.52

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*, 4510-4520. doi: 10.1109/CVPR.2018.00474

Sanida, M. V., Sanida, T., Sideris, A., and Dasygenis, M. (2023). An efficient hybrid CNN classification model for tomato crop disease. *Technologies* 11. doi: 10.3390/technologies11010010

Tang, L., Yi, J.-z., and Li., X.-y. (2023). Improved multi-scale inverse bottleneck residual network based on triplet parallel attention for apple leaf disease identification. *J. Integr. Agric.* 23.

Tan, M., and Le, Q. V. (2019). EfficientNet: rethinking model scaling for convolutional neural networks. *abs/1905.11946.*

Tang, Y., Chen, C., Leite, A. A. C., and Xiong, Y. (2023). Editorial: Precision control technology and application in agricultural pest and disease control. *Front. Plant Sci.* 14. doi: 10.3389/fpls.2023.1163839

Touvron, H., Cord, M., and J'egou, H. e. (2022). "DeiT III: revenge of the viT," in *European Conference on Computer Vision*.

Uddin, M. Z., Mahamood, M. N., Ray, A., Pramanik, M. I., Alnajjar, F., and Ahad, M. A. R. (2024). E2ETCA: End-to-end training of CNN and attention ensembles for rice disease diagnosis1. *J. Integr. Agricult*. doi: 10.1016/j.jia.2024.03.075

Ullah, Z., Alsubaie, N., Jamjoom, M., Alajmani, S. H., and Saleem, F. (2023). EffiMob-net: A deep learning-based hybrid model for detection and identification of tomato diseases using leaf images. *Agriculture* 13. doi: 10.3390/agriculture13030737

Ulyanov, D., Vedaldi, A., and Lempitsky, V. S. (2016). Instance normalization: the missing ingredient for fast stylization. *abs/1607.08022.*

Vos, C. M., Yang, Y., De Coninck, B., and Cammue, B. P. A. (2014). Fungal (-like) biocontrol organisms in tomato disease control. *Biol. Control* 74, 65–81. doi: 10.1016/j.biocontrol.2014.04.004

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassanien, A. E., and Pandey, H. M. (2020). An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. *Comput. Electron. Agric.* 175, 105456. doi: 10.1016/j.compag.2020.105456

Wang, C., Li, C., Han, Q., Wu, F., and Zou, X. (2023). A performance analysis of a litchi picking robot system for actively removing obstructions, using an artificial intelligence algorithm. *Agronomy* 13. doi: 10.3390/agronomy13112795

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q. (2020). "ECA-net: efficient channel attention for deep convolutional neural networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11531-11539. doi: 10.1109/CVPR42600.2020

Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I. S., et al. (2023). "ConvNeXt V2: co-designing and scaling convNets with masked autoencoders," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16133-16142. doi: 10.1109/CVPR52729.2023.01548

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. (2018). CBAM: convolutional block attention module. *ArXiv abs/1807.06521*. doi: 10.48550/arXiv.1807.06521

Wu, Y. (2021). Identification of maize leaf diseases based on convolutional neural network. *J. Phys.: Conf. Ser.* 1748, 032004. doi: 10.1088/1742-6596/1748/3/032004

Wu, Y., and He, K. (2020). Group normalization. *Int. J. Comput. Vision* 128, 742–755. doi: 10.1007/s11263-019-01198-w

Xie, C., and He, Y. (2016). Spectrum and image texture features analysis for early blight disease detection on eggplant leaves. *Sensors* 16, 676. doi: 10.3390/s16050676

Ye, L., Wu, F., Zou, X., and Li, J. (2023). Path planning for mobile robots in unstructured orchard environments: An improved kinematically constrained bi-directional RRT approach. *Comput. Electron. Agric.* 215, 108453. doi: 10.1016/j.compag.2023.108453

Yu, W., Zhou, P., Yan, S., and Wang, X. (2023). InceptionNeXt: when inception meets convNeXt. *abs/2303.16900.*

Zaki, S., Zulkifley, M. A., Mohd Stofa, M., Kamari, N., and Mohamed, N. (2020). Classification of tomato leaf diseases using MobileNet v2. *IAES Int. J. Artif. Intell. (IJ-AI)* 9, 290. doi: 10.11591/ijai.v9.i2

Zeng, W., and Li, M. (2020). Crop leaf disease recognition based on Self-Attention convolutional neural network. *Comput. Electron. Agric.* 172, 105341. doi: 10.1016/j.compag.2020.105341

Zhang, Y., Huang, S., Zhou, G., Hu, Y., and Li, L. (2023). Identification of tomato leaf diseases based on multi-channel automatic orientation recurrent attention network. *Comput. Electron. Agric.* 205, 107605. doi: 10.1016/j.compag.2022.107605

Zhang, Z., and Sabuncu, M. R. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. *abs/1805.07836.*

Zhao, S., Peng, Y., Liu, J., and Wu, S. (2021). Tomato leaf disease diagnosis based on improved convolution neural network by attention module. *Agriculture* 11, 651. doi: 10.3390/agriculture11070651

Zhao, Y., Sun, C., Xu, X., and Chen, J. (2022). RIC-Net: A plant disease classification model based on the fusion of Inception and residual structure and embedded attention mechanism. *Comput. Electron. Agric.* 193, 106644. doi: 10.1016/j.compag.2021.106644

Zhou, C., Zhou, S., Xing, J., and Song, J. (2021). Tomato leaf disease identification by restructured deep residual dense network. *IEEE Access* 9, 28822–28831. doi: 10.1109/Access.6287639

frontiers | Frontiers in Plant Science

TYPE Methods
PUBLISHED 13 August 2024
DOI 10.3389/fpls.2024.1411689

TP-Transfiner: high-quality segmentation network for tea pest

Ruizhao Wu[1], Feng He[1,2], Ziyang Rong[1,2], Zhixue Liang[3], Wenxing Xu[4], Fuchuan Ni[2]* and Wenyong Dong[3]*

[1]College of Informatics, Huazhong Agricultural University, Wuhan, China, [2]Engineering Research Center of Intelligent Technology for Agriculture, Ministry of Education, College of Informatics, Huazhong Agricultural University, Wuhan, China, [3]School of Computer Science, Wuhan University, Wuhan, China, [4]College of Plant Science & Technology, Huazhong Agricultural University, Wuhan, China

Detecting and controlling tea pests promptly are crucial for safeguarding tea production quality. Due to the insufficient feature extraction ability of traditional CNN-based methods, they face challenges such as inaccuracy and inefficiency of detecting pests in dense and mimicry scenarios. This study proposes an end-to-end tea pest detection and segmentation framework, TeaPest-Transfiner (TP-Transfiner), based on Mask Transfiner to address the challenge of detecting and segmenting pests in mimicry and dense scenarios. In order to improve the feature extraction inability and weak accuracy of traditional convolution modules, this study proposes three strategies. Firstly, a deformable attention block is integrated into the model, which consists of deformable convolution and self-attention using the key content only term. Secondly, the FPN architecture in the backbone network is improved with a more effective feature-aligned pyramid network (FaPN). Lastly, focal loss is employed to balance positive and negative samples during the training period, and parameters are adapted to the dataset distribution. Furthermore, to address the lack of tea pest images, a dataset called TeaPestDataset is constructed, which contains 1,752 images and 29 species of tea pests. Experimental results on the TeaPestDataset show that the proposed TP-Transfiner model achieves state-of-the-art performance compared with other models, attaining a detection precision (AP50) of 87.211% and segmentation performance of 87.381%. Notably, the model shows a significant improvement in segmentation average precision (mAP) by 9.4% and a reduction in model size by 30% compared to the state-of-the-art CNN-based model Mask R-CNN. Simultaneously, TP-Transfiner's lightweight module fusion maintains fast inference speeds and a compact model size, demonstrating practical potential for pest control in tea gardens, especially in dense and mimicry scenarios.

KEYWORDS

tea pest, instance segmentation, dense and mimicry scenarios, attention mechanism, Mask Transfiner

Check for updates

OPEN ACCESS

EDITED BY
Yongliang Qiao,
University of Adelaide, Australia

REVIEWED BY
Aibin Chen,
Central South University Forestry and Technology, China
Weibin Wu,
South China Agricultural University, China
Nitin Goyal,
Central University of Haryana, India

*CORRESPONDENCE
Fuchuan Ni
✉ fcni_cn@mail.hzau.edu.cn
Wenyong Dong
✉ dwy@whu.edu.cn

RECEIVED 03 April 2024
ACCEPTED 11 July 2024
PUBLISHED 13 August 2024

CITATION
Wu R, He F, Rong Z, Liang Z, Xu W, Ni F and Dong W (2024) TP-Transfiner: high-quality segmentation network for tea pest. *Front. Plant Sci.* 15:1411689.
doi: 10.3389/fpls.2024.1411689

1 Introduction

As a vital economic crop, tea faces annual challenges from various pests during its cultivation, significantly impacting productivity and quality. Major tea pests include *Jacobiasca formosana*, *Geisha distinctissima*, *Arctornis alba*, *Measuring worm*, *Tortricida*, *Amata germana*, and *Euricania ocellus*, among others. Throughout the evolution of some tea pests, their morphological characteristics often undergo significant changes (Ihsan-ul Haq et al., 2003), making it difficult to manually track pest dynamics. Additionally, the mimicry and dense distribution characteristics exhibited by some tea pests complicate their identification and localization. Consequently, these challenges have driven the development of artificial intelligence for pest monitoring.

Convolutional neural network (CNN) is a primary choice for image processing and is widely used in various fields of computer vision. Sharma et al. (2022); Yang et al. (2024), and Singh et al. (2022) conduct image recognition and classification tasks across different application fields by constructing CNNs with various architectures. These studies leverage the excellent feature extraction capabilities of CNN and demonstrate the superiority and robustness of their respective models through experiments.

As deep learning continues to revolutionize various domains, its application in plant monitoring has garnered significant attention, leading to innovative solutions and enhanced performance in plant disease and pest detection. Liu and Wang (2021) explore challenges in the practical application of deep learning for plant disease and pest detection. They propose potential solutions, presented research ideas to address these challenges, and offered insightful suggestions. Kaur et al. (2024) utilized the H-CSM model, which integrates support vector machine (SVM), convolutional neural network (CNN), and convolutional block attention module (CBAM) to detect and classify plant leaf diseases. Experimental results indicate a classification accuracy of 98.72%. Kang et al. (2023) introduce MCUNet, a corn leaf pest detection and segmentation model that outperforms mainstream neural networks. Furthermore, aiming to obtain a more lightweight model, Agarwal et al. (2023) propose a pest detection method utilizing the EfficientNetB3 model. Experimental results demonstrate the effectiveness in achieving high accuracy for classifying various pests in image datasets. Dai et al. (2023) introduce an improved YOLOv5m-based method for pest detection in plants. By integrating Swin-Transformer and Transformer mechanism, their approach improves the detection accuracy and efficiency. Besides this, Jiao et al. (2022); Tian et al. (2023), and Yang et al. (2023a) also utilized deep learning methods to detect and classify pests on various plants. In summary, these studies have predominantly relied on conventional detection methods for monitoring and has not performed segmentation of the detected pests or leaf diseases. By achieving high detection accuracy through the construction of pest datasets and model improvements, these studies effectively address challenges such as small targets, multiscale detection, and real-time requirements.

In contrast, the field of pest or leaf diseases monitoring in tea gardens remains relatively underexplored, with only a few studies focusing on tea pest monitoring (Wang et al., 2023; Yang et al., 2023b; Ye et al., 2024). These studies primarily concentrate on the detection of tea pests without further segmentation of individual pests. The complex distribution of pests in tea gardens, characterized by mimicry and dense populations, presents significant challenges for traditional pest detection models. As for tea pest monitoring, a previous work conducted by Zhou et al. (2021) uses automatic machine learning to classify each image in the TeaPestDataset. Xue et al. (2023); Yang et al. (2023b), and Lin et al. (2023) utilize the popular object detection model YOLO (Redmon et al., 2016) to detect tea plant diseases or pests. Hu et al. (2021) employ a discriminative pyramid network for semantic segmentation of tea geometrids in natural scenes. Experimental results demonstrate excellent performance in the semantic segmentation of tea geometrids. In contrast, this research treats each pest as an individual entity, achieving specific pest counts and improving edge processing capabilities by developing a deeper network for instance segmentation. Furthermore, this study not only accurately identifies both larva and adult tea geometrids but also encompasses the identification and processing of 27 other common pests in tea gardens. Moreover, Hu et al. (2024) employ hybrid architecture based on transformer to detect tea pests in complex backgrounds. However, previous researches on tea pest monitoring primarily focus on classification, detection, or semantic segmentation tasks, ignoring the importance of instance segmentation tasks for pest control. This study summarizes previous researches on tea pest detection and applies instance segmentation tasks to improve the effectiveness of tea pest control. Instance segmentation offers a promising solution to these issues by enabling pixel-wise parsing of pest images, thereby accurately predicting the position of each pest.

Additionally, in practical applications, traditional detection methods face significant limitations, particularly in scenarios involving target overlap and occlusion, leading to suboptimal detection performance. Moreover, precise pesticide application in tea gardens necessitates adjusting dosages based on pest size to balance effective pest control with environmental concerns. Various pests and disease pathogens exhibit different degrees of resistance to pesticides at various growth stages. Consequently, pesticides should ideally be applied during periods when pests are most susceptible. The results of segmentation tasks can provide detailed information on pest growth, development, and distribution, which is critical for precise pesticide application.

To address these limitations caused by detection models, recent studies committed to segmentation tasks have shown potential solutions. Classical two-stage segmentation models, such as Mask R-CNN (He et al., 2017), Mask Scoring R-CNN (Huang et al., 2019), HTC (Chen et al., 2019), and DCT-Mask (Shen et al., 2021) exhibit excellent segmentation performance. Besides this, one-stage models such as BCNet (Ke et al., 2021) and SOLO (Wang et al., 2021) also have superior performance and efficiency. However, these segmentation models may lack sensitivity to details and edge features, leading to unsatisfactory extraction results and aliasing. Mask Transfiner (Ke et al., 2022) incorporates Transformer architecture into the model to provide supervision and self-correction for regions erroneously predicted by Mask R-CNN. Built upon this innovative mechanism, the segmentation performance of the edge area is significantly optimized.

The attention mechanism is a crucial component in various algorithmic theories within the realm of computer vision. The integration of the attention module with the deep network enhances the network's ability to better extract target features (Xu et al., 2021)—for instance, Wang et al. (2022) demonstrate the effectiveness of the attention module combined with D2Det in pest segmentation. Yang et al. (2023b) improve the YOLOv7-tiny model by utilizing deformable convolution and attention mechanism, achieving 93.23% accuracy on their self-made tea pest segmentation dataset. Additionally, Zhang and Huang (2022) design a novel attention mechanism to overcome challenges such as scale changes, complex backgrounds, and dense distribution in light trap images. Experimental results show that the model outperforms both classic detection models and lightweight detection models.

Besides this, the deformable convolutional network (DCN) (Dai et al., 2017) enhances feature extraction accuracy by employing deformable convolution kernels. A deep convolutional network combined with a deformable convolution structure is proposed by Cao et al. (2020) to overcome geometric transformations. Experiments have demonstrated that the framework, when fused with the DCN, effectively improves the accuracy as well as inference speed of object detection. Significant improvement has been observed in the trade-off between them.

In order to effectively solve the monitoring problems of tea pests in mimicry and dense scenarios, this study proposes a framework named TeaPest-Transfiner (TP-Transfiner) for tea pest detection and segmentation tasks using an enhanced Mask Transfiner framework. The main contributions are as follows:

- Provide a dataset including 1,752 tea pest images and corresponding annotated file, which can be used in the object detection and instance segmentation tasks.
- Fuse the attention mechanism into the backbone network and improve the FPN architecture of the Mask Transfiner to get a novel pest monitoring model TP-Transfiner.
- Implement experiments and demonstrate that while maintaining lightweight, TP-Transfiner outperforms classical models for tea pest detection and segmentation tasks, particularly in dense and mimicry scenarios.

2 Materials and methods

This section summarizes the datasets used in this study and the implementation details of the proposed TP-Transfiner model. Specifically, Section 2.1 discusses the collection, annotation, and data augmentation of the TeaPestDataset. Section 2.2 details the overall process and implementation of the TP-Transfiner model. Section 2.3 presents the evaluation metrics used in the experiments.

2.1 TeaPestDataset and data augmentation

To develop widely applicable pest detection and segmentation models, a carefully selected and labeled dataset is necessary. In this study, various types of pest images in diverse scenarios are collected and manually labeled, resulting in a total of 1,752 images. The original pictures in the dataset are primarily sourced through three methods. The first method involved images provided by agriculture and forestry-related laboratories and pictures pertaining to tea pest knowledge. The second method consisted of on-site shooting in tea gardens using mobile devices. The third source was from Internet search engines. Consequently, the collected scenes are mainly categorized into indoor (laboratory or specimen) and outdoor (natural environment of the tea garden) scenes. Specifically, there are 1,492 images of outdoor scenes and 260 images of indoor scenes. These images serve as the original dataset for tasks related to the localization and segmentation of pest instances.

Figure 1 presents samples of the dataset. The first row displays the original images, and the second row shows the annotated images. The dataset includes images and annotations of tea pests in mimicry and dense scenarios, providing a foundation for the model's robust generalization performance in these complex scenes. During the dataset design process, 22 common pest species found in

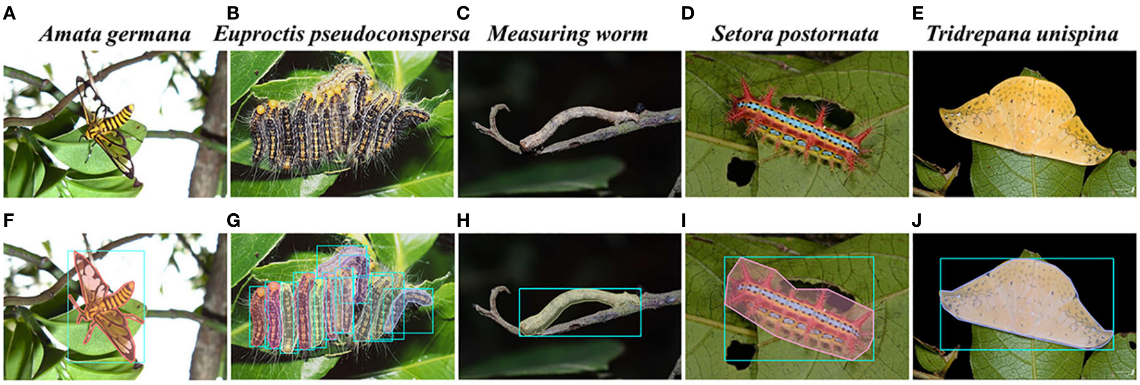


FIGURE 1
Original and annotated images of the pests. **(A–E)** Original and **(F–J)** ground truth.

tea plantations are selected. However, considering the significant morphological differences during different growth stages of some pests, the larvae and adult stages of certain pests are further subdivided. Consequently, the final dataset comprises 29 categories, with the specific quantities of images in indoor and outdoor scenes for each type of pest illustrated in Figure 2.

The initial sample size is limited, and there is inconsistency in the number of various data types. This condition may lead to model overfitting, causing a tendency to predict categories with a higher number of samples. Hence, the original dataset is augmented to achieve a more uniform distribution of each data type. In addition to rotation and cropping, random affine transformations and random color transformations (including adjustments to image brightness, contrast, saturation, and hue) are applied to enhance the model's generalization ability, as shown in Figure 3. Finally, the dataset in this study includes a total of 34,928 images across 29 categories. The process of making the dataset is to divide the 1,752 original images in a ratio of 7:2:1 and then perform data augmentation on the training set and validation set. To avoid falsely high precision, the test set remains the original images.

2.2 TeaPest-Transfiner

This study introduces an optimized framework—for instance, segmentation of tea pests based on Mask Transfiner. Primarily, it integrates the attention mechanism and DCN module into backbone network, replacing the backbone network in Mask Transfiner. Additionally, it utilizes the feature-aligned pyramid network (FaPN) (Huang et al., 2021) as a feature extraction module to segment the edge of each instance in high quality. Figure 4 depicts the network diagram of the optimized Mask Transfiner segmentation model, referred to as TP-Transfiner.

2.2.1 Backbone network

Most of the time, backbone network refers to the feature extraction network, and its function is to extract information from the image, which is then utilized by the box head and mask head. In this study, a ResNet fused with attention module and FaPN are combined as the backbone network of Mask Transfiner, which is used to extract features of pests.

2.2.1.1 ResNet

The ResNet is proposed by He et al. (2016), and it has been proven to effectively improve the accuracy and convergence of deep learning. A ResNet learns image data by its well-designed residual block (as shown in Figure 5A), which can be defined as Equations 1 and 2.

$$y = F(x, \{W_i\}) + x \tag{1}$$

$$y = F(x, \{W_i\}) + W_s x \tag{2}$$

where $F(x, \{W_i\})$ denotes the residual mapping to be learned, and x is the input vector of previous layers or image. If the dimensions of x and F are not equal, a linear projection W_s can be applied to match the dimensions, as shown in formula 2.

According to the research of He et al. (2016), the experimental results illustrate that the residual block has the ability in solving problems such as gradient vanishing and training degradation of the deep network. ResNet has outstanding feature extraction performance without increasing the model parameters and computational burden. Therefore, ResNet is chosen as the backbone network. At the same time, to balance efficiency and accuracy, ResNet-50 is chosen.

2.2.1.2 Attention mechanism

The attention mechanism in deep learning draws inspiration from the attentional processes observed in human vision. Essentially, it comprises a set of weight parameters that can

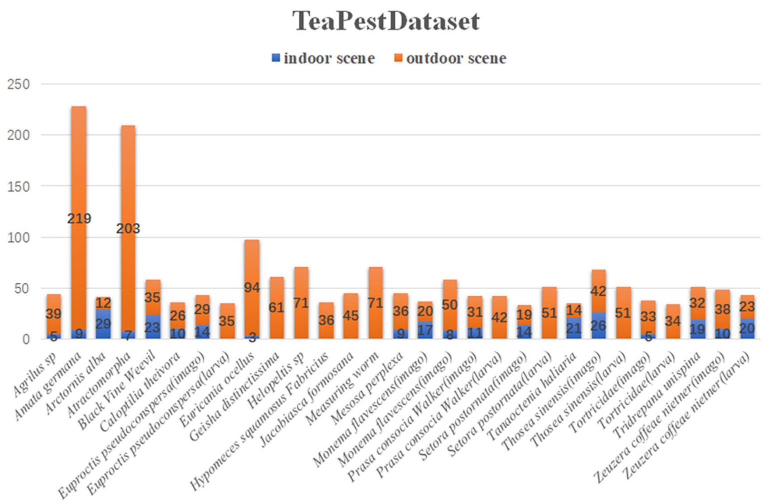


FIGURE 2
Schematic diagram of the number of types of data collection.

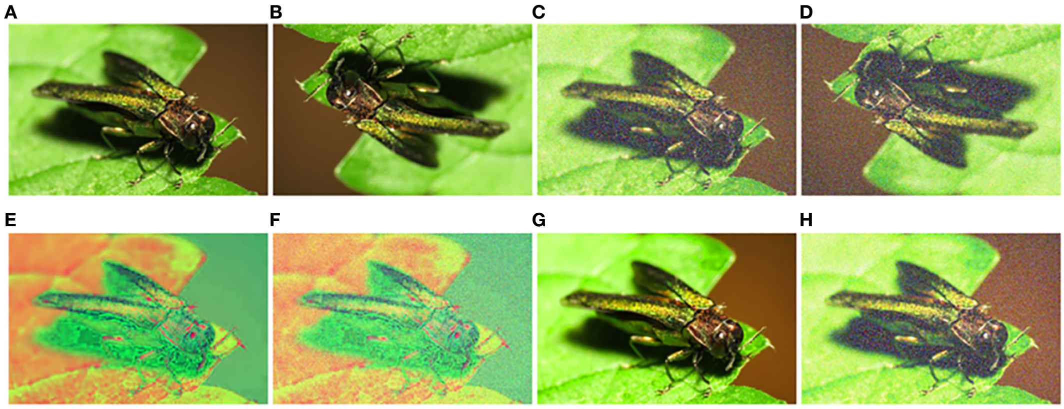


FIGURE 3
Examples of data augmentation. **(A)** Original, **(B)** flip, **(C)** noise adding, **(D)** flip and noise adding, **(E)** adjust hue 1, **(F)** adjust hue 2, **(G)** adjust saturation, and **(H)** adjust saturation and noise adding.

autonomously learn during the training period through the network. The mechanism prioritizes region of interest (RoI) in a dynamically weighted manner, simultaneously suppressing irrelevant background regions.

Dai et al. (2019) propose a solution to address the issue of context fragmentation by integrating the transformer attention module into the backbone network. Building upon this foundation, Zhu et al. (2019) conduct a comprehensive study that investigated the influence of four different factors: the query and key content, the query content and relative position, the key content only, and the relative position. Additionally, they explore the impact of incorporating deformable convolution into the network. Empirical results show that a proper combination of deformable convolution and the key content only term in transformer attention achieves the best accuracy–efficiency trade-off compared with the transformer attention module alone. Based on this conclusion, the key content self-attention module is integrated into the ResNet-50 backbone network in this study. Detailed information is indicated by Equation 3.

$$\xi = u_m^T V_m^C x_k \tag{3}$$

where u_m is a learnable vector. It captures salient key content which should be focused on the task and is irrelevant to the query. T represents the transpose of a vector, and m represents one of the

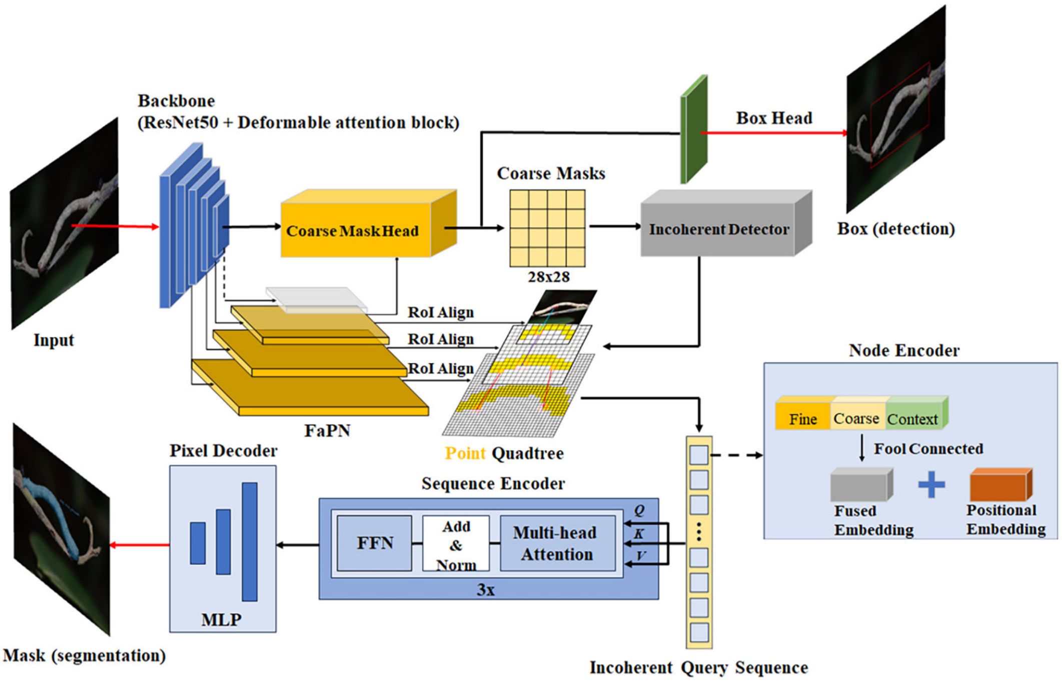


FIGURE 4
Framework of TP-Transfiner.

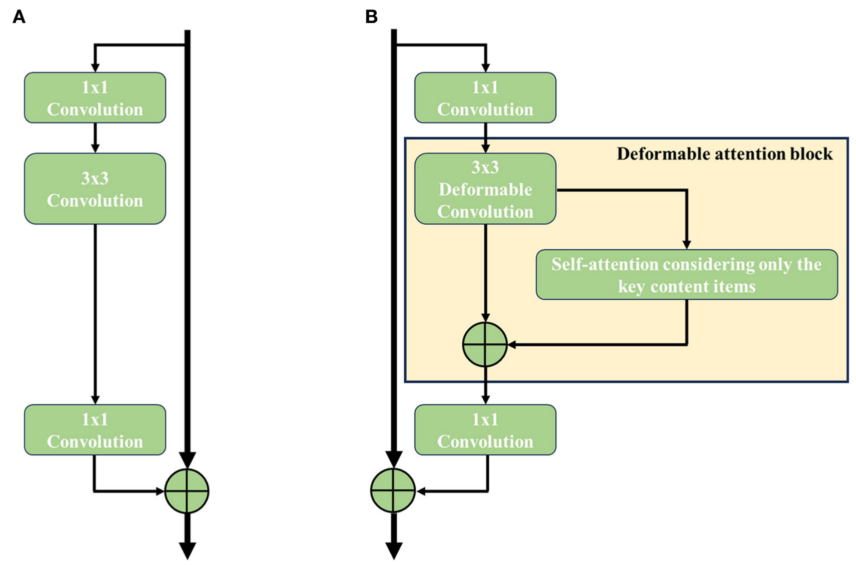


FIGURE 5
Structure comparison of residual block in ResNet-50. **(A)** Original residual block and **(B)** deformable attention block.

attention heads. V_m^C is learnable embedding matrices for the key content and x_k denotes the input.

Specifically, the 3 × 3 regular convolution in the residual block is replaced with a deformable convolution block. Subsequently, a 3 × 3 deformable convolution in the residual block is followed by the addition of a self-attention module, contributing to the deformable attention block (as shown in Figure 5B). To apply a pre-trained model without altering its original behavior, the self-attention module is inserted using a residual connection. The output of the self-attention module is then multiplied by a learnable scalar initialized to zero. The residual block after the third stage of ResNet-50 is replaced with an optimized one, and the feature map outputted by ResNet-50 serves as the input for FaPN for multi-scale feature extraction.

2.2.1.3 FaPN

Achieving accurate mimetic pest instance detection requires the availability of both high-quality spatial information for precise object detection and robust semantic information for effective classification. FaPN optimizes FPN by replacing 1 × 1 convolutions with a feature selection module (FSM) and adding a feature alignment module (FAM) during upsampling, as shown in Figure 6. Inspired by SENet (Hu et al., 2018), FSM accurately extracts crucial information about features and recalibrates them by performing channel reduction and suppressing redundant feature maps. FSM can be represented by Equation 4.

$$\widehat{C_i} = F_s(C_i + f_m(z) * C_i) \tag{4}$$

Here z signifies the data obtained through global average pooling of the input feature map C_i, while $f_m(z)$ denotes the modeling of the importance of each feature map through a process involving a 1 × 1 convolution followed by a sigmoid activation on z.

FAM refines each sampling position within the convolution kernel by employing a learnable offset, thereby aligning the upsampled feature map with a set of feature maps. The feature map $C_{i\text{-}1}$ furnishes the spatial position to determine P_i, ensuring alignment with $C_{i\text{-}1}$. FAM can be explained by Equation 5:

$$\widehat{P_i} = F_a\left(P_i, f_\circ \quad (\hat{C}_{i-1} \circ P_i)\right) \tag{5}$$

where $\circ$ signifies the channel concatenation operation, $f_\circ$ denotes the learned offset, and $F_a(\cdot)$ represents the alignment function.

2.2.2 Segmentation algorithm

To avoid a large number of edge pixels being misclassified, Mask Transfiner considers not only the high-level semantics of the image but also the large-resolution deep feature maps. With these fusion features, Mask Transfiner gains better result than the classic framework for tea pest detection and segmentation tasks in dense and mimicry scenarios. Besides this, the bounding box used for the detection task is generated by the original Faster R-CNN (Ren et al., 2016).

The mask head of Transfiner employs a quadtree structure to represent discrete points at various levels, addressing the discrete distribution characteristics of information loss areas. It utilizes a segmentation network based on Transformer to predict the label of each tree node instance in discontinuous space. As shown in Figure 4, the network comprises three modules—node encoder, sequence encoder, and pixel decoder—which work together to convert discrete nodes into unordered pixel sequences and predict instance labels for each point.

2.2.3 Loss function

Based on the structures above, the entire Mask Transfiner framework can be trained in an end-to-end manner. As shown in Equation 6, a multi-task loss function is defined as:

$$L = \lambda_1 \, L_{Detect} + \lambda_2 L_{Corase} + \lambda_3 L_{Refine} + \lambda_4 L_{Incoherent} \tag{6}$$

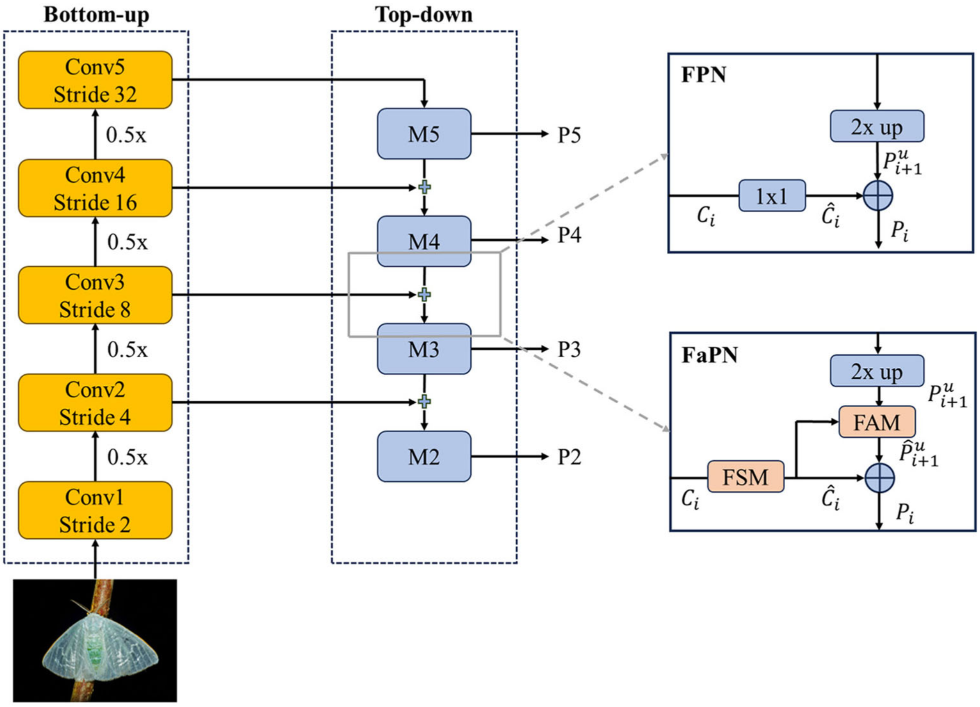


FIGURE 6
Structural comparison of FPN and FaPN.

Here L_{Refine} signifies refinement with L1 loss between predicted labels for incoherent nodes and their ground-truth labels. In TP-Transfiner, L_{Refine} is replaced with smooth L1 loss. Besides this, a binary cross-entropy loss $L_{Incoherent}$ is utilized for detecting incoherent regions. The detection loss, denoted as L_{Detect}, encompasses both localization and classification losses derived from the base detector, exemplified by Faster R-CNN. Subsequently, L_{Coarse} represents the loss attributed to the initial coarse segmentation prediction generated by Mask R-CNN. The weights $\lambda_{1,2,3,4}$ are initially given as {1.0,1.0,1.0,0.5}, respectively.

To mitigate the challenge posed by mimetic and close contact instances, focal loss (Lin et al., 2017) is introduced to L_{Coarse} during training. Focal loss is tailored to address class imbalance in object detection tasks, where background class pixels dominate. Traditional cross-entropy (CE) loss struggles with the surplus of background samples, hindering optimal learning for the minority foreground class. Similarly, the mimicry of tea pests requires TP-Transfiner model to pay more attention to instances camouflaged within the background during training.

2.2.3.1 Focal loss

Yao et al. (2022) utilize focal loss to train Mask R-CNN and Mask Scoring R-CNN for peach disease segmentation. Experimental results indicated that after parameter adjustment, focal loss not only enhances segmentation accuracy but also improves detection rate. Based on this conclusion, focal loss is introduced to L_{Coarse} to enhance the performance of TP-Transfiner, and parameters are adjusted in the same way.

Focal loss introduces a modulating factor that down-weights the contribution of well-classified examples, focusing more on the hard-to-classify samples. The key idea is to assign lower weight to easily classified examples and higher weight to misclassified or challenging examples. Equation 7 shows detailed definition for focal loss.

$$FL(p_t) = -\sum_{i=1}^{N}\alpha_i(1-p_i)^{\gamma}\log(p_i) \tag{7}$$

Here p_t represents the predicted probability of the true class, and γ is a focusing parameter initially defined. Notably, α_i represents the category weight assigned to each sample, where samples belonging to the same category share identical weights.

2.3 Evaluation metric

This study primarily focuses on object detection and instance segmentation tasks. Mean average precision (mAP) serves as a commonly used evaluation metric in object detection. Araújo et al. (2019) and Hong et al. (2020) proposed that its corresponding index is the average of the average precision rate (mAP). This metric is calculated using the values of true positive (TP) and false positive (FP) to assess the detection and segmentation results. Equations 8 and 9) can be employed for calculation. The higher the two parameters are, the better the detection and segmentation results.

$$Bbox - mAP = mean(\frac{TP}{TP+FP}) \tag{8}$$

$$Seg - mAP = \sum_{i=1}^{k} \frac{AP(i)}{C} \tag{9}$$

where TP, FP, and FN represent true positive, false positive, and false negative, respectively. AP is the average precision of pixels segmentation, and C is the number of segmentation categories. Furthermore, AP50 and AP75 in detection task represent mAP of Bbox when IoU is 0.5 and 0.75, respectively. Also, AP50 and AP75 in segmentation task represent mAP of mask when IoU is 0.5 and 0.75, respectively.

3 Results and discussion

This section summarizes all the experiments and related extended discussions conducted in this study to demonstrate the effectiveness of the TP-Transfiner model. Section 3.1 presents the hyperparameter settings and the training process of the model. Section 3.2 discusses the results of adjusting two parameters in focal loss. Section 3.3 compares the TP-Transfiner with state-of-the-art models. Section 3.4 details the ablation study of the model.

3.1 Implementation

The experiments in this paper are conducted in Linux environment of the CentOS system, utilizing Python 3.7 and the PyTorch 1.7.1 framework. Two NVIDIA Tesla V100 32 GB GPUs are employed for training. Stochastic gradient descent (SGD) with momentum is chosen as the optimization method during training, with a momentum parameter set to 0.9 and 1K constant warm-up iterations. Besides this, the initial learning rate is set to 0.01, with a weight decay factor of 0.0001. The batch size is 8, and the training process extends over 12 epochs. The learning rate is reduced to 0.1 times the original value after the 8th and 11th epochs, respectively. After each epoch, the model is validated on the validation set and the weights of the current model are saved. The Mask Transfiner encoder consists of three standard transformer layers. Each layer has four attention heads with feature dimension at 256. Furthermore, the improved Mask Transfiner is initialized using the original Mask R-CNN model pre-trained on the COCO dataset (Lin et al., 2014) to accelerate the training process. All experiments are conducted on Detectron2 (Wu et al., 2019).

3.2 Adaption of parameters

In the current study, focal loss is utilized with empirical values of γ = 2 and α = 0.25. However, it is noted that different data distributions may require different parameters. Therefore, various values of γ and α are tested to accommodate these variations. As shown in Table 1, the implementation of focal loss enhances the overall accuracy of TP-Transfiner, with BCE loss resulting in the lowest accuracy. For each γ, the optimal α is determined to fit the dataset. As α increases, the weight of difficult samples increases, but excessively large α values can decrease the accuracy of the model. Table 1 demonstrates that the experimental results align well with these observations. The table only displays detailed results when γ= 2. For γ= 1,3,4,5, only the optimal results are shown. After multiple rounds of testing, the model achieves the best result on the validation set when γ = 2 and α = 0.45. As a result, focal loss improves the overall segmentation accuracy by 2.1%.

To illustrate the optimization achieved with focal loss, the accuracy on the validation set and changes in loss during the training period are depicted. Figure 7A presents the validation mAP of bounding boxes (IoU = 0.5) from epoch 1 to epoch 12 when training the dataset with different loss functions, indicating that the validation mAP of bounding boxes is higher with focal loss compared to BCE loss. Figure 7B shows the validation mAP of segmentation (IoU = 0.5) over the same epochs when trained with different loss functions, similarly demonstrating that the mAP of segmentation is higher with focal loss. Figure 7C illustrates the

TABLE 1 Training parameter and test results based on TP-Transfiner with different loss functions.

Model	Bbox_mAP (%)	Segm_mAP (%)	Loss type		Epoch	γ	α
			L_{Coarse}	L_{Refine}			
TP-Transfiner	67.499	64.000	BCE	Smooth L1 loss	12		
TP-Transfiner	68.501	65.650	Focal	Smooth L1 loss	12	2	0.25
TP-Transfiner	67.875	65.744	Focal	Smooth L1 loss	12	2	0.35
TP-Transfiner	67.372	**66.123 (+2.1)**	Focal	Smooth L1 loss	12	2	0.45
TP-Transfiner	68.247	65.913	Focal	Smooth L1 loss	12	2	0.55
TP-Transfiner	67.359	64.851	Focal	Smooth L1 loss	12	2	0.75
TP-Transfiner	67.259	63.473	Focal	Smooth L1 loss	12	2	0.95
TP-Transfiner	67.960	65.290 (+1.3)	Focal	Smooth L1 loss	12	1	0.45
TP-Transfiner	67.834	65.237 (+1.2)	Focal	Smooth L1 loss	12	3	0.55
TP-Transfiner	67.900	65.217 (+1.2)	Focal	Smooth L1 loss	12	4	0.55
TP-Transfiner	67.791	65.010 (+1.0)	Focal	Smooth L1 loss	12	5	0.45

The bold value indicates segmentation accuracy when the model performs best. The values in brackets are the added values compared to the first row of Table 1.

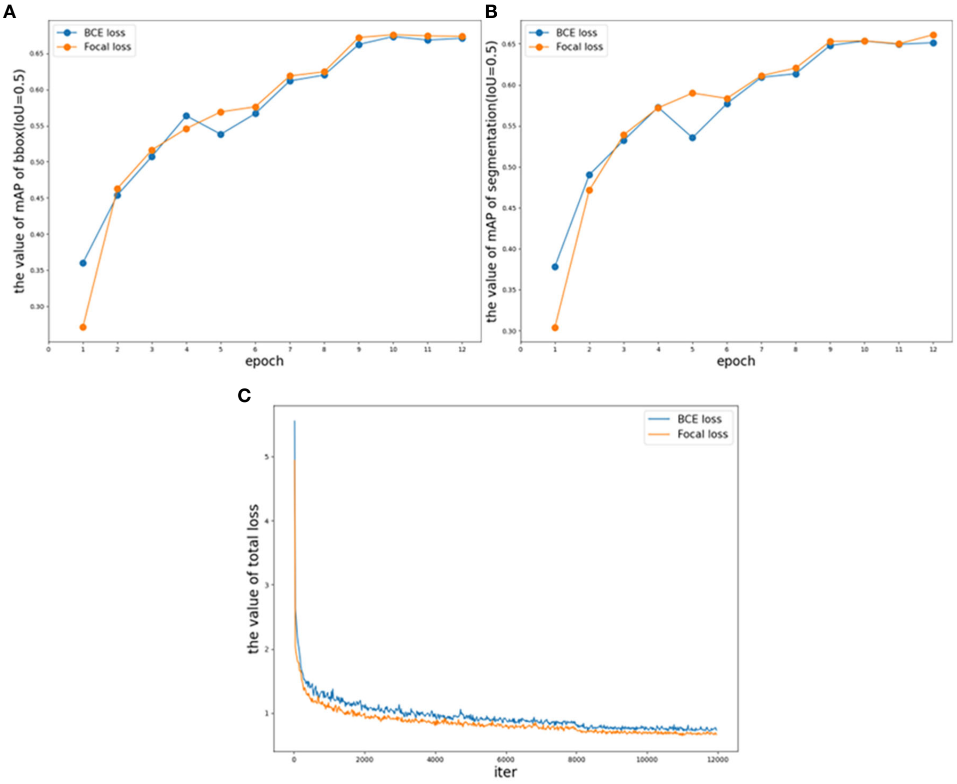


FIGURE 7
TP-Transfiner with different loss validation parameters and loss functions. **(A)** Comparison of mAP of bbox (IoU = 0.5) on different loss, **(B)** comparison of mAP of segmentation (IoU = 0.5) on different loss, and **(C)** comparison of total loss.

trend of training loss under different loss functions. It is evident that focal loss effectively reduces the loss during the training period compared to BCE loss, making the model more suitable for the distribution of the dataset. The results shown in Figure 7 and Table 1 indicate that the application of TP-Transfiner with focal loss achieves superior performance compared to BCE loss.

3.3 Comparison to state-of-the-art models

Detecting and segmenting pests in mimicry and dense scenarios poses a formidable challenge in tea production industry. The proposed TP-Transfiner model demonstrates excellent performance in addressing the detection and segmentation tasks especially in dense and mimicry scenarios. As illustrated in Figures 8A–H, conventional models such as BCNet, Mask R-CNN, Mask Scoring R-CNN, DCT-Mask, and HTC struggle to precisely segment intricate parts like antennae. Similarly, Mask Transfiner encounters difficulty in effectively capturing detailed features. In contrast, TP-Transfiner exhibits outstanding performance in accurately detecting and segmenting pests with detailed characteristics.

3.3.1 Performance in mimicry scenarios

Some tea pests like *Measuring worm* and *Mesosa perplexa* are very good at using the surrounding environment to disguise themselves. This phenomenon, called mimicry, greatly increases the difficulty of the neural network to detect tea pests. Through superior edge feature extraction ability, TP-Transfiner demonstrates excellent performance in detecting and segmenting mimetic pests. In Figures 8I–P, though various models segment the pest camouflaged in leaves, TP-Transfiner distinguishes itself by segmenting the detailed antennae and small body. Besides this, as shown in Figures 8Q–X, BCNet, Mask R-CNN, Mask Scoring R-CNN, DCT-Mask, and HTC all misidentify branches as pests. The original Mask Transfiner slightly improved the situation, while TP-Transfiner improves the segmentation of the mimetic pest very well. As a result, the proposed TP-Transfiner can effectively detect and segment the specific contours of tea pests in such scenarios.

3.3.2 Performance in dense scenarios

In the dense scenario depicted in Figure 9, the segmentation results of TP-Transfiner significantly outperforms other models. Though some models fail to segment two instances in contact (BCNet, Mask R-CNN, and Mask Scoring R-CNN) or segment overlapping objects, TP-Transfiner performs well. Additionally, TP-Transfiner demonstrates powerful ability in detail processing. Compared to other models, the mask predicted by TP-Transfiner comprehensively covers the entire detected pests, while BCNet, Mask R-CNN, Mask Scoring R-CNN, HTC, DCT-Mask, and Mask Transfiner retains a large number of unpredicted pixels belonging to pests. Overall, TP-Transfiner demonstrates superior edge feature extraction ability compared with other models,

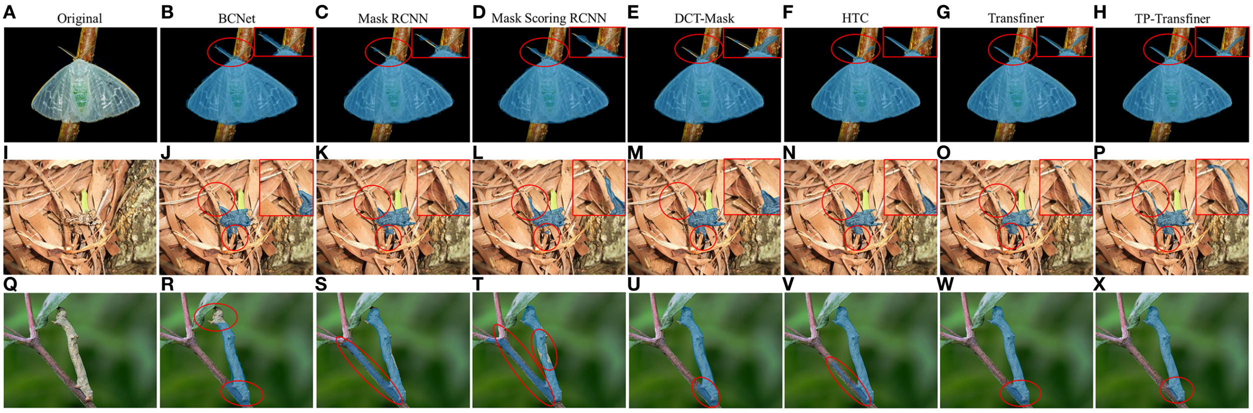


FIGURE 8
Comparison of segmentation results of different state-of-the-art models for tea pests with detailed and mimetic feature. **(A–H)** *Arctornis alba*, **(I–P)** *Mesosa perplexa*, and **(Q–X)** *Measuring worm*.

enabling accurate detection and segmentation of tea pests in dense distribution.

Besides this, this study compares the detection and segmentation accuracy of seven state-of-the-art models including BCNet, Mask R-CNN, Mask Scoring R-CNN, HTC, DCT-Mask, Mask Transfiner, and TP-Transfiner. The results are shown in Table 2. Compared with BCNet, Mask R-CNN, Mask Scoring R-CNN, DCT-Mask, and HTC, the original Transfiner has obvious advantages in instance segmentation, and BCNet and HTC have higher accuracy in object detection task. Subsequently, the study optimizes the Transfiner by integrating deformable convolution, attention mechanism, and FaPN, resulting in the TP-Transfiner. Comparative analysis reveals that TP-Transfiner outperforms other methods, achieving the highest detection accuracy (mAP) of 67.372% and segmentation accuracy (mAP) of 66.123% for object detection and instance segmentation tasks. As for the light weight of the model, TP-Transfiner has a more significant advantage than other segmentation framework (except the original Transfiner). It denotes that TP-Transfiner holds broader application prospects in tea gardens with limited hardware equipment.

3.4 Ablation study

3.4.1 Impact of deformable attention block

To evaluate the feature extraction ability of the deformable attention block on transparent wings, slender antennae, and legs of tea pests, detailed comparative experiments are conducted. Figure 10 illustrates the segmentation effects of two different modules on pests with varying characteristics. It is evident that the model integrating the deformable attention block significantly improves the detection and segmentation effect of pest antennae (A, B), transparent wings (B, C), and mimicry scenario (D). The results demonstrate the deformable attention block's exceptional feature extraction ability for pest's edges and transparent states. The impact

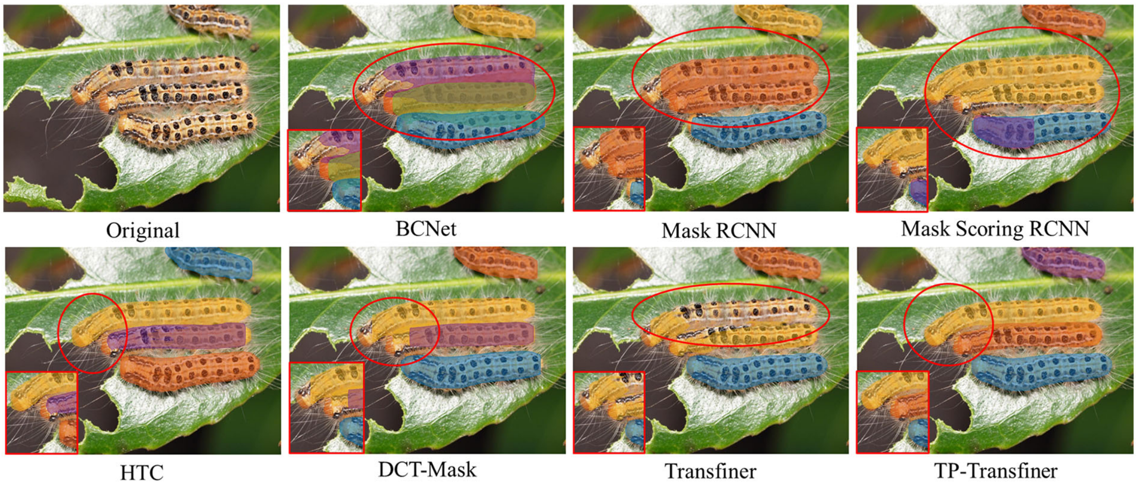


FIGURE 9
Comparison of segmentation results of different state-of-the-art models for tea pests (*Euproctis pseudoconspersa*) in dense scenarios.

TABLE 2 Comparison to different state-of-the-art models.

Model	Model size (MB)	Detection			Segmentation		
		mAP (%)	AP50 (%)	AP75 (%)	mAP (%)	AP50 (%)	AP75 (%)
BCNet (one-stage)	292.90	59.330	76.910	67.151	52.498	75.301	58.357
MS R-CNN	460.20	56.705	84.605	64.578	56.547	81.881	63.909
Mask R-CNN	335.92	56.971	84.419	66.690	56.704	83.114	63.562
HTC	590.40	61.923	83.300	69.713	57.956	80.821	65.301
DCT-Mask	736.23	57.571	83.304	66.378	58.057	82.768	66.289
Transfiner	202.35	59.886	85.067	68.738	60.687	84.871	69.149
TP-Transfiner	235.07	**67.372**	**87.211**	**76.271**	**66.123**	**87.381**	**76.002**

The six bold values are the accuracies of the best models. For the specific meaning of accuracies, refer to the table header.

of the deformable attention block on detection and segmentation accuracy will be illustrated in the next section.

3.4.2 Effect of different modules

Table 3 illustrates that the integration of various modules into the Transfiner framework yields distinct accuracy improvements, with a more pronounced enhancement observed upon combining three modules. Compared with Transfiner, the proposed TP-Transfiner model improves the object detection accuracy (mAP) by 8.6% and the segmentation accuracy (mAP) by 5%. In addition, the fusion of modules does not affect the inference speed on images.

3.4.2.1 Effect of DCN

DCN learns by updating the offset, allowing the convolution kernel to align more closely with the shape and size of the object during sampling. This approach proves to be efficient for segmenting densely distributed and mimetic tea pests. Experimental results show that employing DCN enhances the accuracy of Transfiner, either integrating self-attention and FaPN or not, in both tea pest detection and segmentation tasks. Besides this, the integration of DCN refines the edge feature extraction results to detailed areas such as the insect's antennae and legs, as shown in the feature extraction visualization outputted by the pyramid network (Figure 11).

3.4.2.2 Effect of self-attention

As an essential component of the Transformer architecture, the module aims to extract global features from input images. As shown in Table 3, it can be observed that before integrating DCN, the fusion of this attention module leads to a decrease in the model's detection and segmentation performance. However, incorporating DCN with self-attention (the deformable attention block) into the backbone results in a subtle improvement on detection and segmentation accuracy. It is noteworthy that while self-attention does not significantly improve accuracy, it enables the backbone network to focus more on detailed information such as the legs and antenna of pests, as shown in Figure 11. This mechanism has a significant impact on the TP-Transfiner's ability to segment mimetic pest with slender antennae.

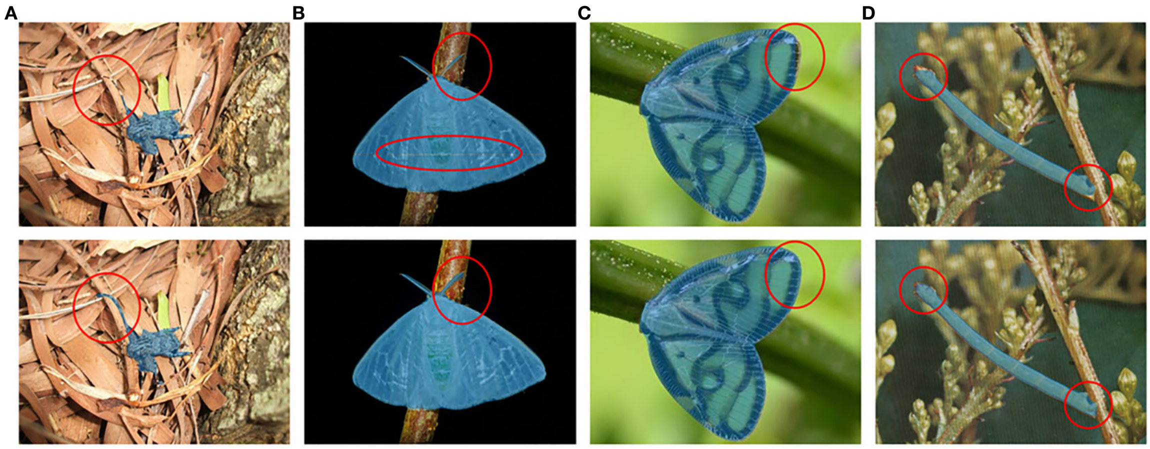


FIGURE 10
Comparison of segmentation results before (the first row) and after (the second row) integrating the deformable attention block. **(A)** *Mesosa perplexa*, **(B)** *Arctornis alba*, **(C)** *Euricania ocellus*, and **(D)** *Measuring worm*.

TABLE 3 Comparison of models after integrating different modules.

Attention	DCN	FaPN	Backbone	Runtime (FPS)	Detection			Segmentation		
					mAP (%)	AP50 (%)	AP75 (%)	mAP (%)	AP50 (%)	AP75 (%)
			ResNet-50	9.7	59.886	85.607	68.738	60.687	84.871	69.149
		√	ResNet50	7.7	64.915	86.167	74.938	63.811	85.744	73.222
√			ResNet50	9.1	61.434	85.937	71.586	61.498	84.761	70.802
√		√	ResNet50	7.3	64.416	86.237	75.089	63.539	85.989	72.271
	√		ResNet50	9.4	65.659	86.462	74.763	63.165	85.620	72.212
	√	√	ResNet50	7.2	67.413 (+7.5)	87.230	76.087	65.244 (+4.8)	86.824	74.499
√	√		ResNet50	9.3	65.278 (+5.4)	86.302	73.891	63.627 (+3.0)	85.737	73.235
√	√	√	ResNet50	7.1	**68.501** **(+8.6)**	**87.433** **(+1.8)**	**77.793** **(+9.0)**	**65.650** **(+5.0)**	**87.081** **(+2.2)**	**75.643** **(+6.5)**

All models employ focal loss during the training period with $\gamma = 0.25$ and $\alpha = 2.0$.
FPS, number of images processed per second.
The six bold values are the accuracies of the best models. For the specific meaning of accuracies, refer to the table header. The values in brackets are the added values compared to the first row of Table 3.

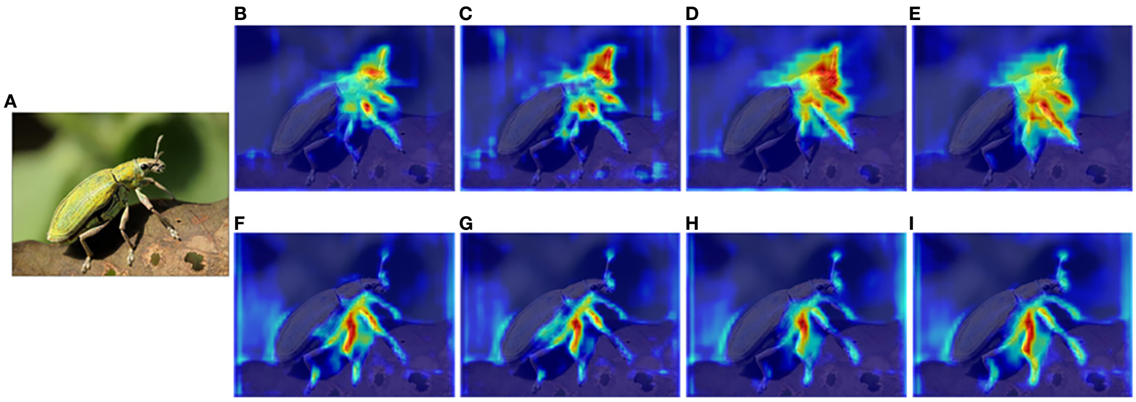


FIGURE 11
Visualization results of feature extraction after the backbone network fuses different modules. **(A)** Input, **(B)** FPN, **(C)** FPN fused with self-attention, **(D)** FPN fused with DCN, **(E)** FPN fused with DCN and self-attention (deformable attention block), **(F)** FaPN, **(G)** FaPN fused with self-attention, **(H)** FaPN fused with DCN, and **(I)** FaPN fused with DCN and self-attention (deformable attention block).

3.4.2.3 Effect of feature-aligned pyramid network

FaPN improves the feature misalignment issue of FPN, particularly around the border area. Therefore, it assists TP-Transfiner in enhancing the feature extraction ability for pest edge, leading to more accurate segmentation of pests in mimicry. A strong comparison depicted in the feature extraction visualization (Figure 11) shows that when FaPN is fused (the second line), the most attended area is distributed around the legs and antennae of the pest. As for the detection and segmentation accuracy, FaPN significantly improves model performance, regardless of whether the self-attention and DCN modules are integrated (as shown in Table 3).

4 Conclusion

To address the limitations of tea pest detection and segmentation in dense and mimicry scenarios, this study develops an end-to-end framework called TP-Transfiner. The framework integrates a deformable attention block, consisting of deformable convolution and a self-attention module, to improve pest feature extraction ability. Additionally, the FPN architecture is enhanced with a more effective FaPN to address feature misalignment issues. Focal loss is introduced during the training period, and $\gamma = 2$ and $\alpha = 0.45$ are adjusted to optimize the model's performance. Furthermore, to solve the insufficient tea pest dataset for detection and segmentation tasks, this study conducts a TeaPestDataset including 29 categories of tea pests. Experimental results on the TeaPestDataset demonstrate that TP-Transfiner has outstanding tea pest detection and segmentation performance compared with several classic models, particularly in dense and mimicry scenarios. The model achieves state-of-the-art performance in both object detection (mAP: 67.372%) and instance segmentation (mAP: 66.123%) tasks, with the same computing resource requirements as the original model while remaining lightweight. Besides this, the deformable attention block is proven to have outstanding feature extraction ability on detailed information.

However, the proposed TP-Transfiner needs to be further improved for pest detection and segmentation in occluded scenes, and it is inefficient for the accurate detection of pests in real-time applications. Therefore, future work will focus on simplifying the model's architecture. Additionally, this study plans to expand the variety and quantity of images in TeaPestDataset. These efforts aim to provide a more precise method for automating pest monitoring.

Data availability statement

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

Author contributions

RW: Writing – original draft, Conceptualization, Formal analysis, Methodology, Software, Visualization, Writing – review & editing. FH: Writing – review & editing, Supervision, Validation. ZR: Supervision, Validation, Writing – review & editing. ZL: Supervision, Validation, Writing – review & editing. WX: Supervision, Writing – review & editing, Investigation, Resources. FN: Data curation, Project administration, Resources, Writing – review & editing. WD: Supervision, Validation, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported by the Fundamental Research Funds for the Central Universities, Huazhong Agricultural University (grant number: 2662022JC005).

Conflict of interest

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

Publisher's note

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

References

Agarwal, A., Vats, S., Agarwal, R., Ratra, A., Sharma, V., and Jain, A. (2023). "Efficient netb3 for automated pest detection in agriculture," in *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*. 1408–1413 (New York, America: IEEE).

Araújo, F. H., Silva, R. R., Ushizima, D. M., Rezende, M. T., Carneiro, C. M., Bianchi, A. G. C., et al. (2019). Deep learning for cell image segmentation and ranking. *Computerized. Med. Imaging Graphics* 72, 13–21. doi: 10.1016/j.compmedimag.2019.01.003

Cao, D., Chen, Z., and Gao, L. (2020). An improved object detection algorithm based on multi-scaled and deformable convolutional neural networks. *Human-centric. Computing. Inf. Sci.* 10, 1–22. doi: 10.1186/s13673-020-00219-9

Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., et al. (2019). "Hybrid task cascade for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (New York, USA: IEEE) 4974–4983.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., et al. (2017). "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*. (New York, USA: IEEE) 764–773.

Dai, M., Dorjoy, M. M. H., Miao, H., and Zhang, S. (2023). A new pest detection method based on improved yolov5m. *Insects* 14, 54. doi: 10.3390/insects14010054

Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q., and Salakhutdinov, R. (2019). "Transformer-xl: Attentive language models beyond a fixed-length context," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. (Cambridge, Massachusetts, USA: MIT) 2978–2988.

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

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*. (New York, USA: IEEE) 770–778.

Hong, S.-J., Kim, S.-Y., Kim, E., Lee, C.-H., Lee, J.-S., Lee, D.-S., et al. (2020). Moth detection from pheromone trap images using deep learning object detectors. *Agriculture* 10, 170. doi: 10.3390/agriculture10050170

Hu, G., Li, S., Wan, M., and Bao, W. (2021). Semantic segmentation of tea geometrid in natural scene images using discriminative pyramid network. *Appl. Soft. Computing.* 113, 107984. doi: 10.1016/j.asoc.2021.107984

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (New York, USA: IEEE) 7132–7141.

Hu, X., Li, X., Huang, Z., Chen, Q., and Lin, S. (2024). Detecting tea tree pests in complex backgrounds using a hybrid architecture guided by transformers and multi-scale attention mechanism. *J. Sci. Food Agric.* 104, 3570–3584. doi: 10.1002/jsfa.13241

Huang, S., Lu, Z., Cheng, R., and He, C. (2021). "Fapn: Feature-aligned pyramid network for dense image prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*. (New York, USA: IEEE) 864–873.

Huang, Z., Huang, L., Gong, Y., Huang, C., and Wang, X. (2019). "Mask scoring r-cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (New York, USA: IEEE) 6409–6418.

Ihsan-ul Haq, M. A., Kakakhel, S., and Khokhar, M. (2003). Morphological and physiological parameters of soybean resistance to insect pests. *Asian J. Plant Sci.* 2, 202–204. doi: 10.3923/ajps.2003.202.204

Jiao, L., Xie, C., Chen, P., Du, J., Li, R., and Zhang, J. (2022). Adaptive feature fusion pyramid network for multi-classes agricultural pest detection. *Comput. Electron. Agric.* 195, 106827. doi: 10.1016/j.compag.2022.106827

Kang, C., Wang, R., Liu, Z., Jiao, L., Dong, S., Zhang, L., et al. (2023). "Mcunet: Multidimensional cognition unet for multi-class maize pest image segmentation," in *2023 2nd International Conference on Robotics, Artificial Intelligence and Intelligent Control (RAIIC)*. 340–346 (New York, USA: IEEE).

Kaur, P., Mishra, A. M., Goyal, N., Gupta, S. K., Shankar, A., and Viriyasitavat, W. (2024). A novel hybrid cnn methodology for automated leaf disease detection and classification. *Expert Syst.* 41, e13543. doi: 10.1111/exsy.13543

Ke, L., Danelljan, M., Li, X., Tai, Y.-W., Tang, C.-K., and Yu, F. (2022). "Mask transfiner for high-quality instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (New York, USA: IEEE) 4412–4421.

Ke, L., Tai, Y.-W., and Tang, C.-K. (2021). "Deep occlusion-aware instance segmentation with overlapping bilayers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (New York, USA: IEEE) 4019–4028.

Lin, J., Bai, D., Xu, R., and Lin, H. (2023). TSBA-YOLO: An improved tea diseases detection model based on attention mechanisms and feature fusion. *Forests* 14, 619. doi: 10.3390/f14030619

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

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. 740–755 (Berlin, German: Springer).

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

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*. (New York, USA: IEEE) 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2016). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 1137–1149. doi: 10.1109/TPAMI.2016.2577031

Sharma, N., Gupta, S., Mohamed, H. G., Anand, D., Mazón, J. L. V., Gupta, D., et al. (2022). Siamese convolutional neural network-based twin structure model for independent offline signature verification. *Sustainability* 14, 11484. doi: 10.3390/su141811484

Shen, X., Yang, J., Wei, C., Deng, B., Huang, J., Hua, X.-S., et al. (2021). "Dct-mask: Discrete cosine transform mask representation for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (New York, USA: IEEE) 8720–8729.

Singh, T. P., Gupta, S., Garg, M., Gupta, D., Alharbi, A., Alyami, H., et al. (2022). Visualization of customized convolutional neural network for natural language recognition. *Sensors* 22, 2881. doi: 10.3390/s22082881

Tian, Y., Wang, S., Li, E., Yang, G., Liang, Z., and Tan, M. (2023). Md-yolo: Multi-scale dense yolo for small target pest detection. *Comput. Electron. Agric.* 213, 108233. doi: 10.1016/j.compag.2023.108233

Wang, H., Li, Y., Dang, L. M., and Moon, H. (2022). An efficient attention module for instance segmentation network in pest monitoring. *Comput. Electron. Agric.* 195, 106853. doi: 10.1016/j.compag.2022.106853

Wang, X., Zhang, R., Shen, C., Kong, T., and Li, L. (2021). Solo: A simple framework for instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 8587–8601. doi: 10.1109/TPAMI.2021.3111116

Wang, Y., Xu, R., Bai, D., and Lin, H. (2023). Integrated learning-based pest and disease detection method for tea leaves. *Forests* 14, 1012. doi: 10.3390/f14051012

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. Available online at: https://github.com/facebookresearch/detectron2.

Xu, J., Lu, K., and Wang, H. (2021). Attention fusion network for multi-spectral semantic segmentation. *Pattern Recognit. Lett.* 146, 179–184. doi: 10.1016/j.patrec.2021.03.015

Xue, Z., Xu, R., Bai, D., and Lin, H. (2023). Yolo-tea: A tea disease detection model improved by yolov5. *Forests* 14, 415. doi: 10.3390/f14020415

Yang, S., Jin, Y., Lei, J., and Zhang, S. (2024). Multi-directional guidance network for fine-grained visual classification. *Visual Comput.* 40, 1–12. doi: 10.1007/s00371-023-03226-w

Yang, S., Xing, Z., Wang, H., Dong, X., Gao, X., Liu, Z., et al. (2023a). Maize-yolo: a new high-precision and real-time method for maize pest detection. *Insects* 14, 278. doi: 10.3390/insects14030278

Yang, Z., Feng, H., Ruan, Y., and Weng, X. (2023b). Tea tree pest detection algorithm based on improved yolov7-tiny. *Agriculture* 13, 1031. doi: 10.3390/agriculture13051031

Yao, N., Ni, F., Wu, M., Wang, H., Li, G., and Sung, W.-K. (2022). Deep learning-based segmentation of peach diseases using convolutional neural network. *Front. Plant Sci.* 13, 876357. doi: 10.3389/fpls.2022.876357

Ye, R., Gao, Q., Qian, Y., Sun, J., and Li, T. (2024). Improved yolov8 and sahi model for the collaborative detection of small targets at the micro scale: A case study of pest detection in tea. *Agronomy* 14, 1034. doi: 10.3390/agronomy14051034

Zhang, W., and Huang, H. (2022). Agripest-yolo: A rapid light-trap agricultural pest detection method based on deep learning. *Front. Plant Sci.* 13, 1079384. doi: 10.3389/fpls.2022.1079384

Zhou, H., Ni, F., Wang, Z., Zheng, F., and Yao, N. (2021). "Classification of tea pests based on automatic machine learning," in *Artificial Intelligence in China: Proceedings of the 2nd International Conference on Artificial Intelligence in China*. 296–306 (Berlin, German: Springer).

Zhu, X., Cheng, D., Zhang, Z., Lin, S., and Dai, J. (2019). "An empirical study of spatial attention mechanisms in deep networks," in *Proceedings of the IEEE/CVF international conference on computer vision*. (New York, USA: IEEE) 6688–6697.

TYPE Original Research
PUBLISHED 15 August 2024
DOI 10.3389/fpls.2024.1420197

Study of resistance mechanism of Alternaria blight (*Alternaria brassicicola*) by biochemical markers in Indian Mustard (*Brassica juncea* L. Czern. &Coss.)

Anurag Mishra[1,2], Nawaz Ahmad Khan[1], Ratnesh Kumar Jha[3], Tamilarasi Murugesh[2] and Ashutosh Singh[3*]

[1]Department of Plant Molecular Biology and Genetic Engineering, Acharya Narendra Deva University of Agriculture and Technology, Ayodhya, Uttar Pradesh, India, [2]Department of Agricultural Biotechnology and Molecular Biology, Dr. Rajendra Prasad Central Agricultural University, Samastipur, Bihar, India, [3]Centre for Advanced Studies on Climate Change, Dr. Rajendra Prasad Central Agricultural University, Samastipur, Bihar, India

OPEN ACCESS

EDITED BY
Chunlei Xia,
Chinese Academy of Sciences (CAS), China

REVIEWED BY
Abhijeet Shankar Kashyap,
National Bureau of Agriculturally Important Microorganisms (ICAR), India
Ni Luh Suriani,
Udayana University, Indonesia

*CORRESPONDENCE
Ashutosh Singh
singh.ashutosh026@gmail.com

RECEIVED 19 April 2024
ACCEPTED 19 July 2024
PUBLISHED 15 August 2024

CITATION
Mishra A, Khan NA, Jha RK, Murugesh T and Singh A (2024) Study of resistance mechanism of Alternaria blight (*Alternaria brassicicola*) by biochemical markers in Indian Mustard (*Brassica juncea* L. Czern. &Coss.). *Front. Plant Sci.* 15:1420197.
doi: 10.3389/fpls.2024.1420197

Indian mustard (*Brassica juncea*) is an important oilseed crop in India. Alternaria leaf spot (Alternaria blight) is incited by the fungus *Alternaria brassicicola*. It majorly affects crop production leading to a yield loss of up to 70%. To circumvent this problem, the study of the resistance mechanism and identification of biochemical markers is one of the important strategies for its management. In the present study, a total of 219 genotypes of Indian mustard with check were screened for Alternaria blight over two seasons. Based on the area under the disease progress curve (AUDPC) scores, ten consistently performing genotypes were selected for the screening of biochemical and yield attributes under artificial inoculated conditions of *Alternaria brassicicola (Berk) Sacc.* The result showed a negative correlation between disease and yield attributes. The catalase (CAT) activity was significantly increased in resistant genotypes compared to susceptible ones, indicating the crucial role of CAT in the resistance mechanism. Pathogen infection also increases the total protein content and the Alternaria-resistant genotype showed the highest total soluble protein while the susceptible genotype showed the lowest total soluble protein. The ten genotypes were categorized by SSI (stress susceptibility index) and Varuna was identified as a tolerant genotype and Giriraj as a susceptible genotype for *Alternaria brassicicola* (Berk) Sacc. Varuna and Giriraj were chosen for quantitative analysis of methionine and tryptophan amino acids from seeds using RP-HPLC (Reverse Phase-High Performance Liquid Chromatography) and there were significant differences in the levels of methionine and tryptophan between the Varuna and Giriraj genotypes. Varuna showed higher methionine and tryptophan content compared to the Giriraj genotype. Higher protein content demonstrated an increase in biotic stress-responsive amino acids, such as methionine and tryptophan, suggesting increased resistance to Alternaria diseases in these high-protein genotypes. These amino acids could be used as biochemical markers for Alternaria resistance of mustard.

KEYWORDS

mustard, Alternaria disease, biochemical, catalase, sulfur-containing amino acids

1 Introduction

Oilseed crops play a key role in the Indian agricultural economy and account for 19% of global acreage but contribute only 2.7% to global production. India is the third largest consumer and importer of edible oils and mainly relies on imports. India imports approximately $10 billion in edible oil annually (Reddy and Immanuelraj, 2017; Jha, 2019; Singh et al., 2021a; Khan et al., 2023). Mustard is commercially significant and it is most widely cultivated due to its resilient nature across diverse agroclimatic conditions (Singh et al., 2018, Singh et al., 2020a, b). Mustard productivity in India is 1.2 t/ha while in Germany productivity is 3.73t/ha (Tyagi and Singh, 2016). Indian mustard confronts a range of challenges, including biotic and abiotic stresses, that limit its productivity (Singh et al., 2021b).

Among the biotic stresses, *Alternaria brassicicola (Berk) Sacc*, (Saccardo, 1886) caused severe yield loss (up to 70%) (Gupta et al., 2020). *Alternaria brassicicola* infections cause dark brown lesions on leaves, stems, and siliquae, which, in turn, diminish photosynthetic efficiency, hasten senescence, and ultimately result in crop losses (Hansen and Earle, 1997; Nowakowska et al., 2019). This disease may cause significant loss in both temperate and tropical regions (Mathpal et al., 2016). In India, *A. brassicicola* reduces yield by up to 47% (Sharma et al., 2013). Initial symptoms of Alternaria produce a series of concentric rings (Mamgain et al., 2013). It is a necrotrophic pathogen that causes lesions in leaves, stems, and siliquae that significantly affect the quality and quantity of mustard seeds by reducing oil content, seed size, and seed color (Duczek et al., 1999). It is important to gain insight into genotypic variability in *Alternaria brassicicola* resistance among mustard crops as it could be used as a potential donor for the development of a resistant variety. Phenotyping over multiple seasons provides stable performance of genotypes (Li and Wu, 2023).

Plant cellular antioxidant enzyme activity is a biochemical response to disease stress. It is considered useful for the early detection of disease and is also associated with Alternaria resistance in mustard. Moreover, successful infections disrupt cell wall proteins and trigger an overproduction of reactive oxygen species (ROS) (Meena et al., 2016a; Narware et al., 2023). ROS plays a critical role in plant development and defense but is often linked to disease susceptibility (Bandyopadhyay et al., 1999). Excessive ROS production can lead to cell membrane damage, protein degradation, and harm to the photosynthetic machinery, causing oxidative stress in plants (Das and Roychoudhury, 2014). To counteract these effects, plants rely on antioxidants, which act as scavengers (Shereefa and Kumaraswamy, 2016). The susceptibility of plants to necrotrophic fungi, such as *Alternaria brassicicola (Berk) Sacc.*, is closely tied to the balance between ROS generation and scavenging through antioxidant defense mechanisms (Sharma et al., 2012). An imbalance in this process indicates a failure of host defense or successful infection. Peroxidase is a ROS scavenger that converts hydrogen peroxide (H_2O_2) to water and plays a critical role in preventing oxidative damage and has been implicated in various defense-related processes, including hypersensitive response, lignification, cross-linking of phenolics and glycoproteins, suberization, and phytoalexin production (Jung et al., 2004; Kashyap et al., 2023). Additionally, catalase is frequently employed by cells to rapidly break down hydrogen peroxide into less reactive gaseous oxygen and water molecules, thereby preventing cellular disintegration (Bolwell and Wojtaszek, 1997; Kannojia et al., 2017). These mechanisms represent the most common means of scavenging ROS during stress responses and play a significant role in plant resistance (Mittler, 2002; Kesharwani et al., 2023).

In this context, sulfur (S) emerges as a vital macronutrient for plants, playing a crucial role in fundamental plant processes and the regulation of various metabolic pathways (Rathore et al., 2015; Hasanuzzaman et al., 2018). Additionally, sulfur plays a critical role in shielding plants from adverse environmental conditions. Sulfur-containing amino acids and metabolites are instrumental in maintaining plant cell mechanisms, thereby enhancing their capacity to withstand Alternaria stress (Hasanuzzaman et al., 2018). Furthermore, sulfur and its derivatives, including glutathione (GSH), hydrogen sulfide (H_2S), methionine (Met), cysteine (Cys), phytochelatin (PC), ATP sulfurylase (ATPS), protein thiols, and others have been observed to fortify antioxidant defenses and mitigate the excessive production of ROS under various biotic stress conditions (Capaldi et al., 2015). Notably, plants from the Brassicaceae family, including important crops, exhibit a higher demand for sulfur compared to other plant varieties to achieve optimal growth and yield. Sulfur is primarily stored in the form of storage proteins, such as cruciferin and napin, with sulfur-rich secondary metabolite referred to as glucosinolate (GSL) (Borpatragohain et al., 2019).

With the background of the above points, this study aimed to screen large populations of Indian rapeseed mustard for Alternaria disease over two seasons to identify sources of disease resistance within rapeseed and field mustard germplasm. Further biochemical analysis and morphological markers studies were conducted to elucidate resistance mechanisms against the Brassica species.

2 Materials and methods

2.1 Material

A total of 219 genetically diverse *Brassica juncea* genotypes with a check genotype (Giriraj) were collected from the Department of Genetics and Plant Breeding, NDUAT Faizabad, Uttar Pradesh India, and the Directorate of Rapeseed-Mustard Research, Bharatpur, Rajasthan, India.

2.2 Screening of mustard genotypes for Alternaria blight

A total of 219 genotypes with susceptible check Giriraj (Dhaliwal and Singh, 2019) of *Brassica juncea* were sown for evaluation of *Alternaria brassicicola* during two crop seasons (2016-17 and 2017-18) under natural field conditions. They were grown in augmented block design. The check was sown after every ten genotypes. Each genotype was sown in two rows with a row length of 1.5 m and a spacing of 30x10 cm. All standard agronomic practices were followed to raise the crops. The genotypes were meticulously assessed and categorized based on their *Alternaria blight* disease severity scores at the reproductive stage. The disease severity scores of *Alternaria blight* are presented in Table 1 (AICRP and Proceeding, 2011).

2.2.1 Measurement of disease severity

To evaluate disease severity, five plants were randomly selected for each genotype and tagged. Infection levels and disease appearance were monitored, and severity was visually recorded for the tagged plants using a rating scale ranging from 0 to 9. It was recorded at three crop growth stages: 60 DAS, 75 DAS, and 90 DAS in both crop seasons (2016-17 and 2017-18). To quantify the disease progress, the area under the disease progress curve (AUDPC) was calculated using the midpoint rule method (Campbell and Madden, 1990). The details of the formula are as follows:

$$\text{AUDPC} = \sum_{i=1}^{n-1}[(t_{i+1} - t_i)\,(y_i + y_{i+1})/2]$$

Where,

y= Percentage of affected foliage at each reading.

t = Time in days between each reading.

n = Total number of readings.

2.2.2 Isolation of *Alternaria brassicicola* and inoculum preparation

Alternaria brassicicola was isolated from leaf samples of infected plants for artificial inoculation in the treated block. The leaf samples were sterilized in 0.5% sodium hypochlorite solution for 1-2 minutes. They were subsequently rinsed thoroughly with distilled water and placed on potato dextrose agar (PDA) culture media (G-Biosciences Geno Technology USA). The cultures were incubated at a temperature of 27°C and kept in 12-hour cycles of light and darkness for 6-10 days for fungal growth. *Alternaria brassicicola* was confirmed by microscopic examination. The morphological characteristics of *Alternaria brassicicola (Berk) Sacc.* such as conidia shape, structure, and size were observed under the microscope (Meena et al., 2016b). A *Alternaria brassicicola* colony was introduced into potato dextrose broth (PDB) and incubated for 10 days. During the incubation periods, the temperature was maintained at 27°C and followed by 12 hours of daylight and 12 hours of darkness. Further, the concentration of the inoculum was carefully determined using a hemocytometer (Avni Scientific Co.) with a size of 30 x 70 mm and 4 mm thickness and adjusted to a level of 5 x 10$^{\wedge 4}$/mL (Akhtar et al., 2007).

TABLE 1 Details of the rating scale and AUDPC (area under the disease progress curve) range used for rating Alternaria blight in rapeseed mustard.

S. No.	Rating Scale (0-9)	Description of scale	AUDPC range	Host reaction
1	0	No visible symptoms	0	Near Immune (I)
2	1	< 5% leaf area covered	≤ 50	Resistant (R)
3	3	5-10% leaf or pod area covered with small pinhead spots on the leaves and superficial pinhead spots on pods	51 - 100	Moderate Resistant (MR)
4	5	11-25% leaf or pod area covered with small spots on leaf and superficial pinhead spots on pods	101-250	Moderately Susceptible (MS)
5	7	26-50% leaf or pod area covered with bigger spots with the initiation of coalesces on leaves and deep lesions on pods	251-500	Susceptible (S)
6	9	> 50% leaf or pod area covered with bigger commonly coalescing spots on leaves and deep lesions on pods	≥ 500	Highly Susceptible (HS)

2.2.3 Evaluation of selected genotypes for morphological and yield traits

A total of 10 sets of consistently performing genotypes were identified after a 2-year field screening which included moderate resistant (MR) (2), moderate susceptible (MS) (1), susceptible (S) (3) and highly susceptible (HS) (4). . These 10 genotypes were sown during the 2018-19 rabi season at the Agricultural Research Farm, NDUAT, Faizabad, India. The experiment was conducted in two blocks in natural field conditions with three replications and a spacing of 30 x 10 cm. Isolated spores of *Alternaria brassicicola* were sprayed (75 DAS) in one block and it was considered a treated block (Chakrabarty et al., 2018). The others that had not been sprayed with spores of *Alternaria brassicicola* were considered as control. In the control block, 0.02% mancozeb fungicide solutions (non-inoculated) were sprayed to control Alternaria blight.

2.2.4 Data collection of genotypes for morphological and yield attributes

Five plants were randomly tagged from each genotype in each replication to record the observations. The following data were recorded from three replications.

Plant height (PH): At the time of maturity, plant height was measured from the ground to the main shoot tip.

Number of pods per plant (PB): The total number of pods counted from each plant and considered as the number of pods per plant.

Secondary branch (SB): The number of secondary branches per plant was assessed by enumerating branches arising from primary branches.

Mean raceme length (SMR) (cm): The length of the main raceme was measured from the joint at the apex of the primary branch to the top of the plant, and the average was taken to represent the overall length.

Siliqua length (SL) (cm): It was measured by randomly selected siliquae from base to tip in centimeters.

Seeds per siliqua (SPS): The number of seeds per siliquae was calculated by counting from five randomly chosen siliquae in each of the five tagged plants and the average value taken.

Siliquae per plant (SPP): the mean of the total number of siliquae counted from the main raceme.

Test weight (TW) (g): 1,000 sun-dried seeds from each selected plant were counted and weighed in grams using an electronic balance.

Seed yield per plant (YPP): The seed yield per plant (g) was recorded by weighing the total seeds obtained after threshing each plant separately.

Area under the disease progress curve (AUDPC): The calculation of the area under the disease progress curve (AUDPC) was based on severity scores during the reproductive stage.

2.2.5 Biochemical analysis

2.2.5.1 Estimation of total soluble proteins

Samples were collected from all 10 genotypes in both the non-inoculated and inoculated conditions 7 days after fungal inoculation on the leaves (82 DAS). A total of nine plants from each genotype were selected. The leaf samples were promptly frozen and stored at a temperature of -80°C. The total soluble protein (TSP) of the leaves was quantified by using methods of Bradford, 1976; Sambrook and Russell, 2001; and Kruger, 2009.

2.2.5.2 Peroxidase activity

Peroxidase activity was estimated from control and treated plants. A fresh leaf sample (200g) was homogenized in 10 ml of phosphate buffer (pH 6.0) and then centrifuged at 10,000 rpm for 30 min at room temperature. Afterward, 2 ml of the enzyme extract was mixed in a test tube containing 2 ml of phosphate buffer (pH 6.0) with 1 ml of pyrogallol and 0.2 ml of H_2O_2. The mixture was incubated at 37°C, shaken, and placed in water for 10 minutes to allow purpurogallin formation. The color intensity was measured at 430 nm using a spectrophotometer to assess the enzyme's activity (McCune and Galston, 1959).

2.2.5.3 Catalase activity

The catalase activity was assessed by Dhindsa et al. (1981) method. The reaction mixture, with a final volume of 1 ml, was prepared and contained 50 mM sodium phosphate buffer (pH 7.0) with 50 µl of enzyme extract. To determine the enzyme activity, 35 µl of H_2O_2 was added at every 5-second interval over a duration of 1 minute. The rate of decreasing absorbance at 240 nm was measured. The catalase activity was quantified by using an extinction coefficient of 39.4 M^{-1} cm^{-1}.

2.2.6 Stress Susceptibility Index (SSI)

The SSI was calculated for all morphological, physiological, and biochemical traits to categorize the genotypes, and it was calculated by using the following formula:

$$SSI = (1 - Y/Yp)/(1 - X/Xp)$$

Where:

Y represents the mean performance of a mustard genotype in an inoculated condition.

Yp represents the mean performance of a mustard genotype in a non-inoculated condition.

X represents the mean performance of all mustard genotypes in an inoculated condition.

Xp represents the mean performance of all mustard genotypes in a non-inoculated condition.

For each trait, genotypes with SSI values below 0.5 were categorized as resistant, those with values between 0.5 and 1 were considered moderately resistant, and genotypes showed values ≥1 were considered as susceptible genotypes (Singh et al., 2024).

2.2.7 Quantitative estimation of methionine and tryptophan amino acids by RP-HPLC

2.2.7.1 Extraction of amino acids from *B. juncea*

The following procedure was used for methionine and tryptophan amino acids estimation.

2.2.7.2 Preparation of extract

The SSI was used to identify two contrasting genotypes (Varuna and Giriraj) for Alternaria blight. These two contrasting genotypes were used for methionine and tryptophan amino acid profiling. The seeds from the genotypes were first stored in a deep freezer overnight. After 24 h, the seeds were finely ground into a powder using a mortar and pestle, and it was kept for vacuum drying.

The vacuum-dried samples were subjected to vapor-phase hydrolysis, using 200 µL of constantly boiling 6N HCl and 40 µL of phenol. Subsequently, the samples were oven-dried at a temperature range of 112-116°C for a duration of 20-24 hours to eliminate any excess HCl. Afterward, the tubes containing the samples were passed through a 90-min vacuum treatment. The seed samples were reconstituted by adding 500 µL of 20 mM boiling HCl.

2.2.7.3 Derivatization of amino acids

Amino acids were derivatized (FMOC–AA) at room temperature using a Precolumn derivatization technique. Initially, a 300 µL aliquot of the mustard seed extract (or a standard amino acid solution) was combined with 600 µL of a 200 mM borate buffer (pH 10.0). Afterward, 600 µL of 15 mM FMOC chloride (in acetonitrile) was added to the mustard seed extract, initiating the derivatization process. The reaction was stopped after 5 min by introducing 600 µL of 300 mM ADAM (a mixture of water and acetonitrile in a 1:1 ratio) and it formed the FMOC–ADAM complex during a 1 min reaction. The sample was filtered and subjected to analysis using RP-HPLC. The entire procedure was completed within 6 minutes. The detection of amino acids was carried out using automated derivatization with FMOC–AA and online analysis was conducted using RP-HPLC with ultraviolet –visible (UV–Vis) detection (Henderson et al., 2003).

Filtered samples were analyzed using a HiQ Sil C18-HS column (4.6 mm × 250 mm × 5µm) and a Systronics high-performance liquid chromatography system equipped with a UV–Vis detector and autosampler. The analysis was monitored at a wavelength of 263 nm, and the column temperature was maintained at 25°C. The mobile phase consisted of an isocratic mixture prepared from 50 mM acetate buffer and acetonitrile (in a ratio of 70:30). The flow rate was set at 0.750 mL/min, and the injection volume was 20 µL. The total runtime for a single sample analysis was 25 min. Quantification of the various compounds was based on peak areas and expressed as equivalents of representative standard compounds. The results were expressed in grams per 100 g of fresh weight.

2.2.8 Statistical analysis

Microsoft Excel 2016 Analysis Tools were utilized for a two-way ANOVA. R Studio was employed to explore Pearson's correlation coefficient, conduct principal component analysis (PCA), and use Duncan's multiple range test (DMRT) to categorize genotypes into different groups.

3 Results

3.1 Screening for Alternaria blight resistance: a comparative analysis of 2016-17 and 2017-18

The result of 2 years (2016-17 and 2017-18) of rigorous screening processes revealed varying degrees of resistance among different genotypes to Alternaria blight. Based on the AUDPC scores, six genotypes were categorized as moderately resistant (MR), 116 genotypes were moderately susceptible (MS), 56 were susceptible (S) genotypes, and 141 genotypes were highly susceptible (HS) in the year 2016-2017. Similarly, in 2017-18, the genotypes were classified: 7 genotypes were MR, 14 were MS, 63 were S, and 135 were HS. It is important to note that none of the genotypes showed resistance in both years (Supplementary Table 1; Figures 1A, B).

A total of 10 contrasting genotypes were selected from the 219 genotypes for morphological, physiological, and biochemical analysis. The selection criteria were based on the consistent performance and host reaction categories observed across the 2-

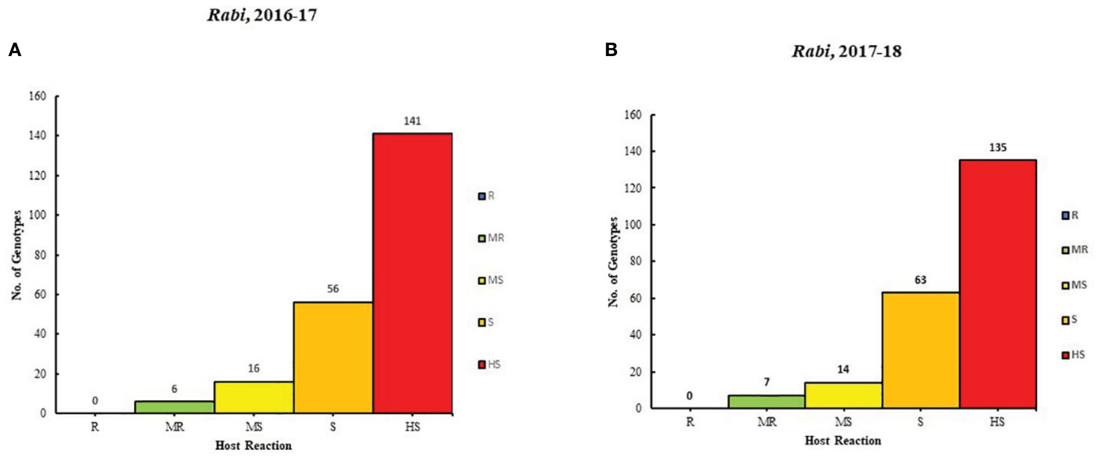


FIGURE 1
Genotype distribution in five host reaction categories to Alternaria blight during the rabi season for **(A)** 2016-17 and **(B)** 2017-18 under natural conditions.

year screening (2016-17 and 2017-2018) (Figure 2). Genotypes Varuna, RGN-13, and Pusa Mustard-25 were selected from the MR category; LET-18 and Pusa Mustard-26 from the MS category; EJ-17 and RGN-48 from the S category; and Giriraj, Anuradha, and RGN-13 from the HS category.

3.2 Performance analysis of genotypes for morphological and yield aspects

All 10 genotypes were inoculated against spore suspensions of isolated *A. brassicicola* (Supplementary Figure 1), and different genotypes showed varying responses. In control plots (mancozeb spray) genotypes in the rabi season of 2018-19 showed distinct AUDPC scores compared to the results from a 2-year screening under natural conditions. Analysis of variance showed that treatments had significant effects on all morphological and yield attributes (Table 2). However, genotypes Varuna, Kranti, PM 25, Anuradha, EJ-17, RGN 13, and RGN 48 showed no significant difference in AUDPC scores between inoculated and non-inoculated conditions. Notably, Varuna and Kranti consistently had lower AUDPC scores in both conditions. On the other hand, Giriraj, PM 26, and LET 18 had AUDPC scores ranging from 251 to 500 in non-inoculated conditions, but these scores increased beyond 500 after they were exposed to infection in inoculated conditions. The Varuna genotype showed the highest YPP (18.020 g) in non-inoculated conditions, followed by Kranti (15.480 g). Both Varuna and Kranti experienced reductions in terms of percent change over inoculated conditions of approximately 28% and 36%, respectively. Meanwhile, Giriraj showed YPP (13.110 g) in non-inoculated conditions and showed the highest reduction of 43% after inoculation against *A. brassicicola* (Supplementary Table 2).

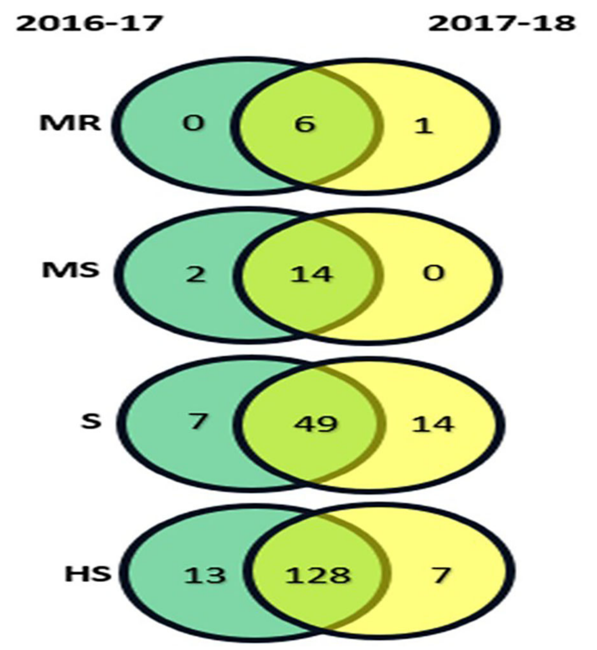


FIGURE 2
Host reaction categories for 219 mustard genotypes under natural field conditions during the rabi seasons of 2016-17 and 2017-18, with categories represented as : R, Resistant; MR, Moderately Resistant; MS, Moderately Susceptible; S, Susceptible; HS, Highly Susceptible.

3.3 Exploring trait associations in inoculated and non-inoculated conditions

For both the inoculated and non-inoculated conditions, area graphs were strategically positioned along the diagonal, and box and whisker plots were situated on the right side of the visual canvas to depict a unified trend. The area graph exhibited a leftward inclination, and the box and whisker plot showed a decrease for all traits following *A. brassicicola* inoculation. It contributed to a cohesive narrative. Notably, the AUDPC, treated as a singular entity, showed a distinct trajectory. In the case of the AUDPC, the area graph demonstrated a rightward inclination, and the box and whisker plot exhibited an increase under inoculated conditions compared to non-inoculated conditions (Figure 3).

A correlation coefficient (r) analysis of the traits under both inoculated and non-inoculated conditions was also conducted. The AUDPC demonstrated a robust, negative correlation with MRL, SPS, SPP, and SMR. In contrast, yield had a strong, positive association with these same traits. Notably, the correlation between yield and the AUDPC showed a significant negative correlation (-0.823). It showed an inverse relationship between yield and AUDPC (Figure 3).

TABLE 2 Combined analysis of variance for yield traits of 10 mustard genotypes under inoculated and non-inoculated conditions.

Source of variation	Df	PH	PB	SB	MRL	SMR	SL	SPS	SPP	TW	YPP	AUDPC
Genotypes (G)	9	3897.28***	6.15***	54.46***	828.67***	309.78***	3.95***	19.85***	4224.99***	0.58***	45.70***	130714.68***
Treatment (E)	1	2969.89***	7.95***	29.95***	580.51***	476.27***	5.35***	52.21***	4493.50***	6.09***	219.42***	73290.2***
Interaction (G X E)	9	9.16^{ns}	0.20*	0.36^{ns}	6.93^{ns}	2.86^{ns}	0.08*	0.65^{ns}	39.10^{ns}	0.06^{ns}	2.45***	3335.48***
Error	38	75.86	0.08	0.23	9.15	4.54	0.04	0.42	87.72	0.05	0.28	657.92
Total	59											

PH, Plant Height; PB, primary branch number; SB, secondary branch number; MRL, main raceme length; SMR, siliqua on main raceme number; SL, siliqua length; SPS, seeds per siliqua number; SPP, siliqua per plant number; TW, test weight; YPP, seed yield per plant; AUDPC, area under disease progress curve.
(* significant at P<0.05, and *** significant at P<0.001).

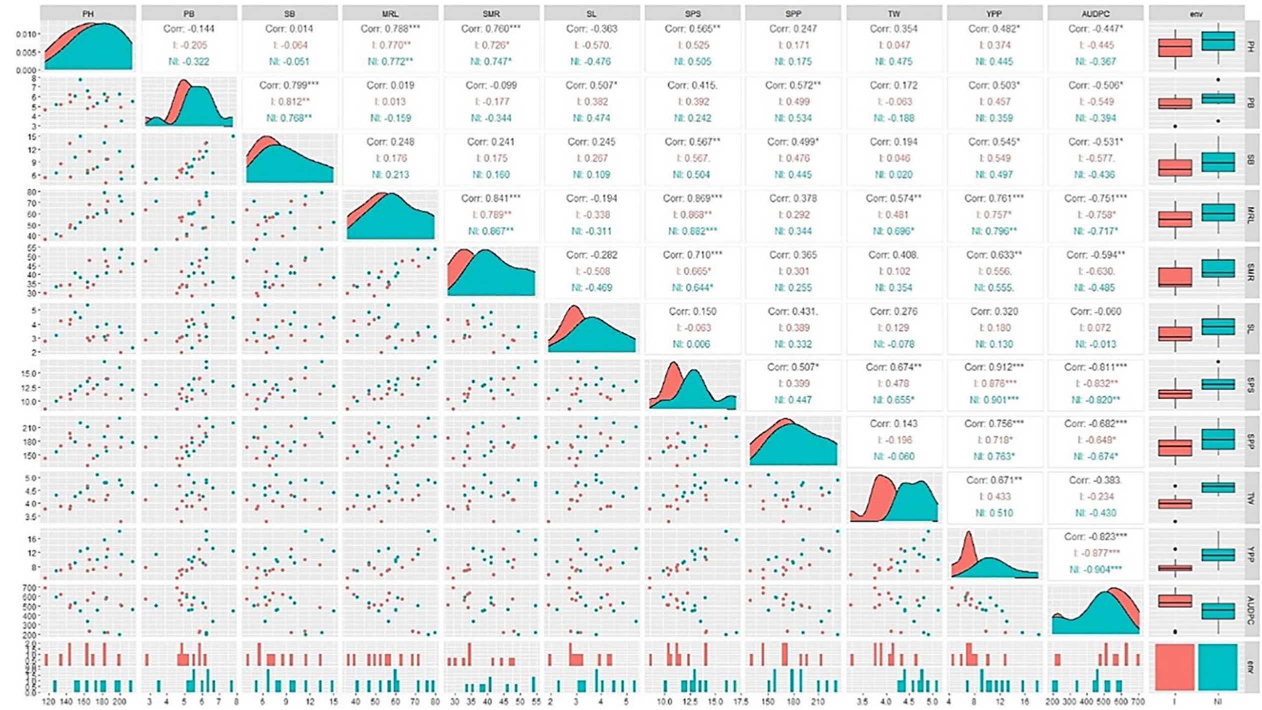


FIGURE 3
Correlation matrix, scatter plot, and data distribution for yield traits in two conditions, with diagonals indicating the distribution of each parameter and the lower triangular matrix indicating scatter plot. Correlation values and their statistical significance are denoted by asterisks and positioned above the diagonal. Red and navy-blue colors represent correlations within the non-inoculated (NI) and inoculated (I) conditions, respectively. Significance levels are as follows: *** for $p \leq 0.001$, ** for $p \leq 0.01$, and * for $p \leq 0.05$. PH, Plant Height; PB, Primary Branch Number; SB, Secondary Branch Number; MRL, Main Raceme Length; SMR, Siliqua on Main Raceme Number; SL, Siliqua Length; SPS, Seeds per Siliqua Number; SPP, Siliqua per Plant Number; TW, Test Weight; YPP, Seed Yield per Plant; AUDPC, Area Under Disease Progress Curve; Trt, Treatment; NI, Non-Inoculated; I, Inoculated.

3.4 Principal component analysis

The results of the PCA revealed the presence of three principal components (PCs) with eigenvalues exceeding 1 under non-inoculated conditions (5.29, 2.80, and 1.32). The first and second PCs individually explained 48.90% and 26.60% of the phenotypic variance, and the cumulative value was 75.50%. The prominent contributing parameters to these two PCs included MRL, YPP, SPS, PB, AUDPC, SMR, and PH (Figure 4A). Conversely, under inoculated conditions, the first three principal components exhibited eigenvalues exceeding or equal to 1 (5.38, 2.93, and 1.00). The first and second PCs independently elucidated approximately 48.10% and 25.50% of the phenotypic variation, collectively amounting to 73.60%. Noteworthy contributors to PC1 and PC2 included MRL, YPP, AUDPC, SPS, SMR, PB, PH, SB, SL, and SPP (Figure 4B). It is pertinent to note that under non-inoculated conditions, the AUDPC contributed 9.99% to the overall variability, while its contribution increased to 11.09% under inoculated conditions.

3.5 Exploring molecular and biochemical insights

The catalase activity was examined for all 10 genotypes and Varuna showed the highest activity (value), while Giriraj exhibited the lowest value under both non-inoculated and inoculated conditions. Specifically, Varuna showed a 13.00% increase in catalase activity under inoculated conditions as compared to the non-inoculated conditions, while Giriraj showed a 27.00% decrease in catalase activity (Figure 5A).

Furthermore, Varuna showed significantly higher peroxidase activity than LET-18 in both non-inoculated and inoculated conditions among the tested genotypes, while LET-18 exhibited considerably lower activity. Remarkably, the inoculated Varuna genotype showed a 10.00% increase in peroxidase activity in contrast to the non-inoculated conditions, whereas Anuradha showed a substantial decrease in peroxidase activity (38.00%) (Figure 5B).

The TSP content of all 10 genotypes was also estimated. Varuna showed the highest TSP content in both non-inoculated and inoculated environments, while Giriraj showed the lowest values. Comparatively, the Varuna genotype TSP content experienced a 5% increase under the non-inoculated compared to the inoculated condition, whereas Giriraj showed a reduction of 23.00% in TSP under the inoculated condition (Figure 5C).

3.6 Genotype grouping based on SSI

The genotypes were categorized into either “moderately resistant” or “susceptible” groups based on their performance across various morpho-physio-biochemical traits, including PH,

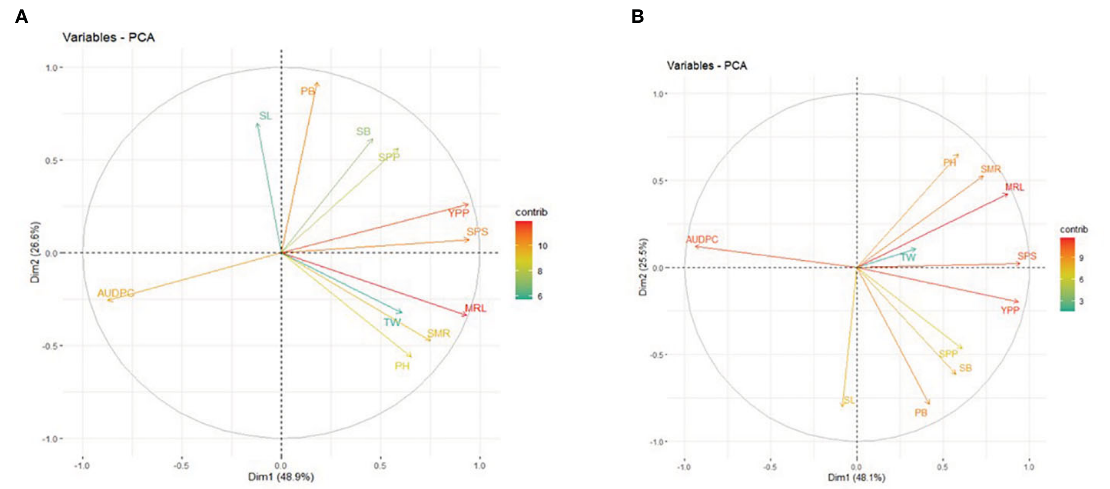


FIGURE 4
Principal Component Analysis (PCA) Biplot of PC1 and PC2: Contributions of Morphological Traits in **(A)** Non-Inoculated **(B)** Inoculated Conditions. PH, Plant Height; PB, Primary Branch Number; SB, Secondary Branch Number; MRL, Main Raceme Length; SMR, Siliqua on Main Raceme Number; SL, Siliqua Length; SPS, Seeds per Siliqua Number; SPP, Siliqua per Plant Number; TW, Test Weight; YPP, Seed Yield per Plant; AUDPC, Area Under Disease Progress Curve. The contribution to phenotypic variation is represented by the color and lengths of the vector.

SB, MRL, SMR, SL, SPS, SPP, TW, YPP, AUDPC, and TSP (Figure 6). Specifically, Genotype Kranti falls into the "resistance" category for traits such as PB, AUDPC, POD, and CAT. Genotype Varuna was categorized as resistant for POD and CAT traits, while PM 25 for AUDPC score. For the remaining traits, these genotypes were categorized as either "moderately resistant" or "susceptible." Genotype Varuna was predominantly classified as either "resistant" or "moderately resistant" across most traits except SB and SL. In contrast, Genotype Giriraj was primarily labeled as "susceptible" for most of the traits, except for SB. Therefore, for amino acid profiling, Varuna was regarded as a "resistant" genotype, while Giriraj was considered "susceptible.

3.7 Amino acid profiling of contrasting mustard genotypes

We also quantified two essential amino acids, methionine and tryptophan, in two contrasting genotypes of mustard (Varuna and Giriraj) seeds. The concentrations of these amino acids were

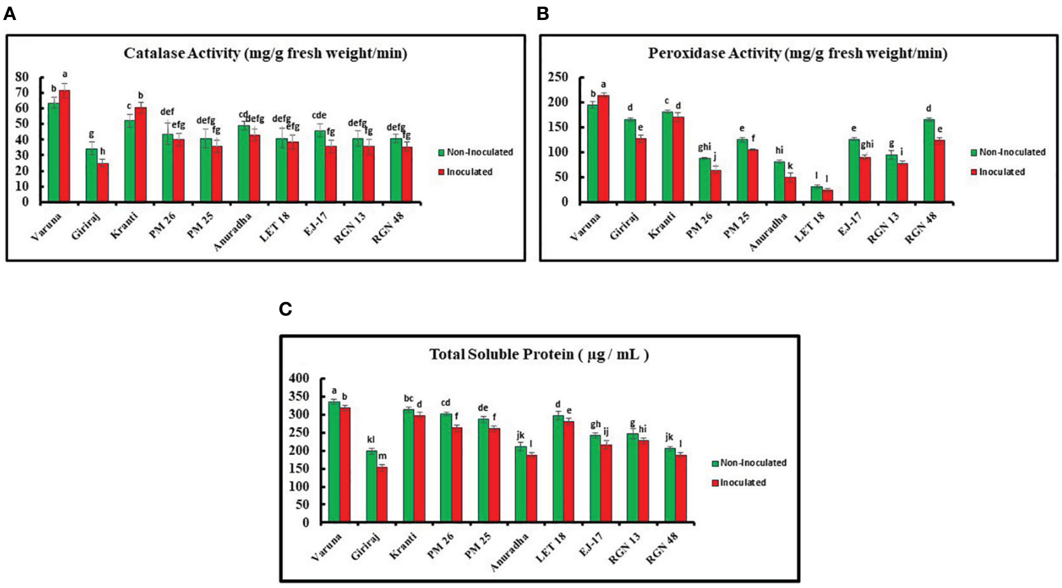


FIGURE 5
(A) Catalase **(B)** Peroxidase Activities **(C)** Total Soluble Protein Content in Mustard Genotypes under Non-Inoculated and Inoculated Conditions. Bars labeled with the same letter indicate no statistically significant difference at a significance level of P<0.05, as determined by the Duncan Multiple Range Test. The average of three determinations is represented by the bars, and error bars indicate the standard error (SE).

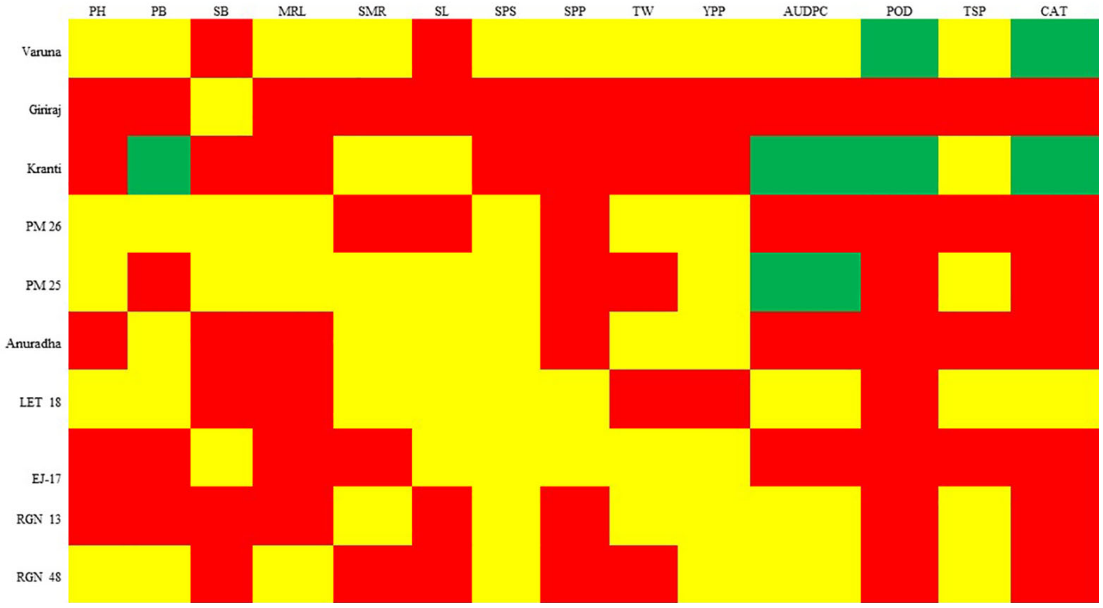


FIGURE 6
SSI grouping of genotypes based on morpho-physio-biochemical traits. The color green corresponds to an SSI value below 0.5, while the color yellow signifies an SSI value ranging from 0.5 to 1. In contrast, the color red denotes an SSI value exceeding 1. These color codes categorize the genotypes as “resistant,” “moderately resistant,” and “susceptible,” respectively. PH, Plant Height; PB, Primary Branch Number; SB, Secondary Branch Number; MRL, Main Raceme Length; SMR, Siliqua on Main Raceme Number; SL, Siliqua Length; SPS, Seeds per Siliqua Number; SPP, Siliqua per Plant Number; TW, Test Weight; YPP, Seed Yield per Plant; AUDPC, Area Under Disease Progress Curve; POD, Peroxidase Activity; TSP, Total Soluble Protein; CAT, Catalase Activity.

examined under both non-inoculated and inoculated conditions (Figure 7). The results revealed that both the Varuna and Giriraj genotypes experienced a decrease in the levels of methionine and tryptophan in the inoculated conditions in comparison to the non-inoculated conditions. Specifically, Varuna displayed an 11% reduction in methionine content, while Giriraj exhibited a more substantial 22% decrease. For tryptophan content, Varuna showed a 12% decrease, while Giriraj demonstrated a notable 39% reduction. Therefore, the level of reduction in amino acids was less for the Alternaria blight-resistant genotype.

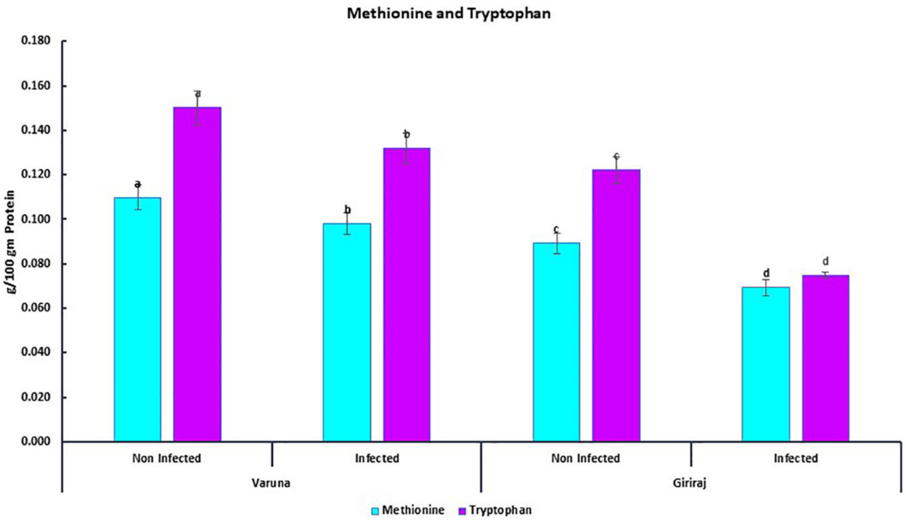


FIGURE 7
The concentration of methionine and tryptophan in Varuna and Giriraj under non-inoculated and inoculated conditions analyzed using RP-HPLC. Bars labeled with the same letter indicate no statistically significant difference at a significance level of P<0.05, as determined by the Duncan Multiple Range Test. The average of three determinations is represented by the bars, and error bars indicate the standard error (SE).

4 Discussion

Biotic stress factors have been observed to impede plant growth and induce unfavorable alterations at both the cellular and molecular levels (Møller et al., 2007). Despite extensive research endeavors, there has been no substantial discovery of materials that confer a high degree of resistance against *A. brassicicola* (Nowicki et al., 2012; Meena et al., 2020; Singh et al., 2021a). While wild Brassica species have demonstrated considerable resistance (Dharmendra et al., 2014), their compatibility with susceptible cultivated varieties remains limited (Nowakowska et al., 2019). Only very limited resistance has been identified in cultivated Brassica species. Consequently, the identification of resistant genotypes within cultivated species assumes pivotal importance for the success of breeding programs in this context.

To facilitate this objective, effective assessment tools for evaluating pathogen resistance are of paramount significance, and it is prudent to subject germplasm to examination during genuine epidemic occurrences. Building on this perspective, previous studies conducted by Summuna et al. (2012), and Singh et al. (2021a) categorized plant varieties into five distinct classes using a modified rating scale introduced by AICRP-RM-2011 (All India Coordinated Research Project - Rapeseed Mustard). Likewise, in our research efforts, we conducted an evaluation of Indian mustard genotypes for their resistance to Alternaria blight over the course of two consecutive growing seasons, specifically in 2016–17 and 2017–18. Within our investigation, we observed that none of the genotypes exhibited resistance to *Alternaria brassicicola*. Instead, they were categorized into four alternative classes based on their reactions to the prevailing field conditions: moderately resistant, moderately sensitive, susceptible, and highly sensitive.

For the performance of rapeseed-mustard genotypes under controlled conditions concerning *Alternaria brassicicola* infection, we selectively identified genotypes from each host reaction group that consistently showed stability over a two-year duration and implemented artificial inoculation during the reproductive phase. A similar study was conducted by Munir et al. (2020), although their focus was on different Brassicaceae species, such as *B. rapa* and *B. napus*. These investigations delved into various plant characteristics, offering valuable insights into plant-pathogen interactions (Saed-Moucheshi et al., 2013).

In the present study, the correlation plot revealed that the AUDPC was inversely correlated with MRL, SPS, SPP, and SMR, while crop yield demonstrated a positive correlation with these growth attributes. This underscores the crucial role of these growth parameters in enhancing plant yield, particularly in the context of disease resistance, and a positive association between growth parameters such as PH, PB, and MRL and plant yield was identified (Poli et al., 2018; Singh et al., 2021a). While siliqua length exhibited a strong negative correlation with the AUDPC, it did not demonstrate significant associations with yield per plant. These results imply that conventional proxies such as siliqua length may not serve as reliable indicators for breeding and selection purposes due to their inverse relationships with disease severity (Naznin et al., 2015; Ali et al., 2018). In contrast, yield displayed a positive correlation with seeds per pod and thousand seed weight. Interestingly, dark leaf spots caused an adverse effect on the formation of healthy seeds within infected Brassica pods. Consequently, severely disease-spotted pods prematurely dried, contracted, and broke open, leading to the premature shedding of shrunken seeds, resulting in yield loss (Allen et al., 1971; Munir et al., 2020). These findings show the importance of prioritizing strategies aimed at enhancing plant survival and facilitating the development of pods with healthier and more abundant seeds. This suggests that it may be more beneficial to focus efforts on Brassica breeding strategies that prioritize these factors rather than solely emphasizing siliqua length (Bennett et al., 2017).

Plants have developed various defense mechanisms to combat invading pathogens. These mechanisms encompass processes such as callose deposition, lignin formation, the production of phytoalexins, generation of reactive oxygen species, induction of pathogenesis-related (PR) proteins, and the presence of enzymes such as peroxidase and catalase (Torres et al., 2006; Almagro et al., 2009; Doughari, 2015; Yadav et al., 2020). Moreover, catalase and peroxidase play pivotal roles in managing excessive H_2O_2 production, which is integral to the plant defense response (Hameed et al., 2008, 2009). Some evidence supports the protective function of POD activity in the context of disease resistance against Alternaria (Tyagi et al., 1998; Hameed et al., 2010). In our investigation, peroxidase activity was elevated after pathogen inoculation Alternaria in Varuna and Kranti. Notably, resistant genotypes (Varuna) displayed higher POD activity under inoculated over non-inoculated conditions. It was observed that increased POD activity in mustard genotypes confronting Alternaria blight (Pandey et al., 2018). Our study showed an increase in CAT activity in both Varuna and Kranti. This surge in CAT activity during the disease period implies the scavanging of excess H_2O_2 quickly generated within the plants (Mittler, 2002). Significantly, our results highlight that CAT activity was more pronounced in resistant genotypes compared to susceptible ones, emphasizing the crucial role of CAT in the resistance mechanism (Debona et al., 2012; Meena et al., 2016a).

Our research findings showed a significant association between the progression of the disease and the levels of TSP in mustard genotypes. It was proposed that the proliferation of pathogens triggers the synthesis of various enzymatic proteins and can alter the nutritional composition of the substrate, ultimately leading to an increase in its protein content (Onifade and Jeff-Agboola, 2003). Amino acids play a pivotal role as substrates in host-pathogen interactions (Titarenko et al., 1993), potentially influencing metabolic processes related to disease resistance and exerting fungistatic effects (Misra et al., 2008; Mathpal et al., 2016). These insights shed light on the intricate relationship between disease progression, protein levels, and the role of amino acids in the interactions between plants and pathogens (Moormann et al., 2022).

It is well understood that under stressful conditions, plants engage in photorespiration as a protective mechanism. This process aids in removing light-induced harmful molecules and maintaining the redox balance. It has been hypothesized that photorespiration in plants contributes to the synthesis of sulfur-containing amino acids, including cysteine, and methionine (Kalwan et al., 2023). In mustard crops, growth regulators derived from methionine and tryptophan operate both independently and collaboratively, contributing to plant resistance to biotic stress (Zemanová et al., 2014). Meanwhile, within the Brassicaceae plant family, secondary metabolites known as glucosinolates, which originate from the sulfur-containing amino acid methionine (aliphatic glucosinolates) and tryptophan (indole glucosinolates), play a vital role in enhancing plant immunity and serve as an inducible defense mechanism against pathogens (Choudhury et al., 2022). Notably, our study revealed significant differences in the levels of methionine and tryptophan between the Varuna and Giriraj genotypes with Varuna showing higher levels of these amino acids. These differences suggest a potential role for methionine and tryptophan in enhancing resistance in mustard seeds. These findings raise questions about the relationship between seed protein content and resistance in various contexts. Interestingly, genotypes with higher protein content demonstrated a notable increase in biotic stress-responsive amino acids, such as methionine and tryptophan, suggesting more resistance to diseases in these high-protein genotypes.

5 Conclusion

Comprehensive screening of mustard genotypes suggested that gene pools had moderately resistant genotypes. Further, investigation suggested that the Varuna genotype showed the highest resistance compared to the rest of the genotypes against Alternaria blight infection while Giriraj showed the least resistance. Catalase activity increased after infection with the pathogen and the tolerant genotype showed more catalase activity. The observed differences in amino acid content between these genotypes may be linked to the presence of specific resistance genes. Further amino acid profiling is required in more genotypes to confirm its relation with biotic stress tolerance. Moderately tolerant genotypes Varuna and Kranti can be utilized for future experiments and can serve as tolerant material for Alternaria disease. However, further experiments are necessary to precisely identify the resistance genes responsible for modulating tryptophan and methionine content, evaluate their expression, determine their cellular localization, and assess their impact on amino acid levels and protein content in mustard seeds. Identifying these resistance genes would significantly aid in categorizing genotypes based on the relationship between the expression levels of these specific resistance genes and amino acid (methionine and tryptophan) content.

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

AM: Formal analysis, Investigation, Methodology, Writing – original draft. NK: Funding acquisition, Resources, Supervision, Visualization, Writing – review & editing. RJ: Funding acquisition, Writing – review & editing. TM: Formal analysis, Writing – review & editing. AS: Conceptualization, Formal analysis, Validation, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

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

Publisher's note

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

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2024.1420197/full#supplementary-material

References

AICRP, R., and Proceeding, M. (2007). „Revised rating scale of major diseases of rapeseed-mustard." in *Proceedings of 18th annual group meeting of AICRP rapeseed-mustard.* (Khanpur campus, AAU, Guwahati (Assam)) (Vol. 1).

Akhtar, K. P., Saleem, M. Y., Asghar, M., Jamil, F. F., and Haq, M. A. (2007). Evaluation of tomato genotypes against Alternaria leaf blight disease. *Pakistan J. Phytopathol.* 19, 15–18.

Ali, N., Khan, N. U., Farhatullah, R. M., Bibi, Z., Gul, S., Ali, S., et al. (2018). Genetic diversity in indigenous landraces of Brassica napus based on morphological and biochemical characteristics using multivariate techniques. *Int. J. Agric. Biol.* 20, 277–287. doi: 10.17957/ijab/15.0488

Allen, E. J., Morgan, D. G., and Ridgman, W. J. (1971). A physiological analysis of the growth of oilseed rape. *J. Agric. Sci.* 77, 339–341. doi: 10.1017/S0021859600024515

Almagro, L., Gómez Ros, L. V., Belchi-Navarro, S., Bru, R., Ros Barceló, A., Pedreño, M. A., et al. (2009). Class III peroxidases in plant defence reactions. *J. Exp. Bot.* 60, 377–390. doi: 10.1093/jxb/ern277

Bandyopadhyay, U., Das, D., and Banerjee, R. K. (1999). Reactive oxygen species: oxidative damage and pathogenesis. *Curr. Sci.*, 658–666.

Bennett, E. J., Brignell, C. J., Carion, P. W., Cook, S. M., Eastmond, P. J., Teakle, G. R., et al. (2017). Development of a statistical crop model to explain the relationship between seed yield and phenotypic diversity within the Brassica napus genepool. *Agronomy*. 7, 31. doi: 10.3390/AGRONOMY7020031

Bolwell, G. P., and Wojtaszek, P. (1997). Mechanisms for the generation of reactive oxygen species in plant defence–a broad perspective. *Physiol. Mol. Plant Pathol.* 51, 347–366. doi: 10.1006/pmpp.1997.0129

Borpatragohain, P., Rose, T. J., Liu, L., Barkla, B.J., Raymond, C. A., and King, G. J. (2019). Remobilization and fate of sulphur in mustard. *Ann. Bot.* 124, 471–480. doi: 10.1093/aob/mcz101

Bradford, M. M. (1976). A rapid and sensitive method for the quantitation of microgram quantities of protein utilizing the principle of protein-dye binding. *Anal. Biochem*. doi: 10.1016/0003-2697(76)90527-3

Campbell, C. L., and Madden, L. V. (1990). *Introduction to plant disease epidemiology* (North Carolina State University, Raleigh, NC, USA: John Wiley & Sons). doi: 10.1017/S0007485300051890

Capaldi, F. R., Gratão, P. L., Reis, A. R., Lima, L. W., and Azevedo, R. A. (2015). Sulfur metabolism and stress defense responses in plants. *Trop. Plant Biol*. 8, 60–73. doi: 10.1007/s12042-015-9152-1

Chakrabarty, R., Kalita, H., and Zaman, A. S. N. (2018). Screening of Indian mustard (Brassica juncea) genotypes for resistance or tolerance against Alternaria blight under natural and artificially inoculated conditions. *J. Crop Weed* 14, 183–187.

Choudhury, S., Rao, M., Kashyap, A., Ahmaed, S., Prasad, L., Singh, N., et al. (2022). Jasmonate mediated inducible accumulation of indole glucosinolates confers resistance against Alternaria blight disease in cruciferous wild species Diplotaxis erucoides. *Physiol. Mol. Plant Pathol.* 122, 101904. doi: 10.1016/j.pmpp.2022.101904

Das, K., and Roychoudhury, A. (2014). Reactive oxygen species (ROS) and response of antioxidants as ROS-scavengers during environmental stress in plants. *Front. Environ. Sci.* 2, 53. doi: 10.3389/fenvs.2014.00053

Debona, D., Rodrigues, F. Á., Rios, J. A., and Nascimento, K. J. T. (2012). Biochemical changes in the leaves of wheat plants infected by Pyricularia oryzae. *Phytopathology*. 102, 1121–1129. doi: 10.1094/PHYTO-06-12-0125-R

Dhaliwal, R. S., and Singh, B. (2019). Pathogenicity test of Alternaria brassicae (Berk.) Sacc. using artificial inoculation methods on common varieties of rapeseed-mustard in Punjab region. *J. Oilseed Brassica* 10, 21–26. doi: 10.13140/RG.2.2.29037.23523

Dharmendra, K., Neelam, M., Yashwant, K. B., Kumar, A., Kumar, K., Srivastava, K., et al. (2014). Alternaria blight of oilseed Brassicas: A comprehensive review. *Afr. J. Microbiol. Res.* 8, 2816–2829. doi: 10.5897/AJMR2013.6434

Dhindsa, R. S., Plumb-Dhindsa, P., and Thorpe, T. A. (1981). Leaf senescence: correlated with increased levels of membrane permeability and lipid peroxidation, and decreased levels of superoxide dismutase and catalase. *J. Exp. Bot*. 32, 93–101. doi: 10.1093/JXB%2F32.1.93

Doughari, J. (2015). An overview of plant immunity. *J. Plant Pathol. Microbiol*. 6, 10–4172. doi: 10.4172/2157-7471.1000322

Duczek, L. J., Seidle, E., Reed, S. L., Sutherland, K. A., Rude, S. V., and Rimmer, S. R. (1999). Effect of swathing on alternaria black spot in Brassica rapa canola in Saskatchewan. *Can. J. Plant Sci.* 79, 299–302. doi: 10.4141/P98-063

Gupta, S., Didwania, N., and Singh, D. (2020). Biological control of mustard blight caused by Alternaria brassicae using plant growth promoting bacteria. *Curr. Plant Biol.* 23, 100166. doi: 10.1016/j.cpb.2020.100166

Hameed, A., Akhtar, K. P., Saleem, M. Y., and Asghar, M. (2010). Correlative evidence for peroxidase involvement in disease resistance against Alternaria leaf blight of tomato. *Acta Physiol. Plant*. 32, 1171–1176. doi: 10.1007/s11738-010-0512-z

Hameed, A., Iqbal, N., and Malik, S. A. (2009). Mannose-induced modulations in antioxidants, protease activity, lipid peroxidation, and total phenolics in etiolated wheat leaves. *J. Plant Growth Regul*. 28, 58–65. doi: 10.1007/s00344-008-9076-1

Hameed, A., Naseer, S., Iqbal, T., Syed, H., and Haq, M. A. (2008). Effects of NaCl salinity on seedling growth, senescence, catalase and protease activities in two wheat genotypes differing in salt tolerance. *Pak. J. Bot.* 40, 1043–1051.

Hansen, L. N., and Earle, E. D. (1997). Somatic hybrids between Brassica oleracea L. and Sinapis alba L. with resistance to Alternaria brassicae (Berk.) Sacc. *Theor. Appl. Genet.* 94, 1078–1085. doi: 10.1007/s001220050518

Hasanuzzaman, M., Bhuyan, M. H. M. B., Mahmud, J. A., Nahar, K., Mohsin, S. M., Parvin, K., et al. (2018). Interaction of sulfur with phytohormones and signaling molecules in conferring abiotic stress tolerance to plants. *Plant Signaling Behav*. 13, e1477905. doi: 10.1080/15592324.2018.1477905

Henderson, J. P., Byun, J., Takeshita, J., and Heinecke, J. W. (2003). Phagocytes produce 5-chlorouracil and 5-bromouracil, two mutagenic products of myeloperoxidase, in human inflammatory tissue. *J. Biol. Chem.* 278, 23522–23528. doi: 10.1074/jbc.M303928200

Jha, G. K. (2019). Oilseeds sector in India: A trade policy perspective. *Indian J. Agric. Sci.* 89, 73–78 doi: 10.56093/ijas.v89i1.86161

Jung, W. J., Jin, Y. L., Kim, Y. C., Kim, K. Y., Park, R. D., and Kim, T. H. (2004). Inoculation of *Paenibacillus illinoisensis* alleviates root mortality, activates of lignification-related enzymes, and induction of the isozymes in pepper plants infected by *Phytophthora capsici*. *Biol. Control*. 30, 645–652. doi: 10.1016/J.BIOCONTROL.2004.03.006

Kalwan, G., Priyadarshini, P., Kumar, K., Yadava, Y. K., Yadav, S., Kohli, D., et al. (2023). Genome wide identification and characterization of the amino acid transporter (AAT) genes regulating seed protein content in chickpea (Cicer arietinum L.). *Int. J. Biol. Macromol.* 252, 126324. doi: 10.1016/j.ijbiomac.2023.126324

Kannojia, P., Sharma, P. K., Kashyap, A. K., Manzar, N., Singh, U. B., Chaudhary, K., et al. (2017). "Microbe-mediated biotic stress management in plants." in *Plant-Microbe Interactions in Agro-Ecological Perspectives: Volume 2: Microbial Interactions and Agro-Ecological Impacts*, 627–648. doi: 10.1007/978-981-10-6593-4_26

Kashyap, A. S., Manzar, N., Meshram, S., and Sharma, P. K. (2023). Screening microbial inoculants and their interventions for cross-kingdom management of wilt disease of solanaceous crops-a step toward sustainable agriculture. *Front. Microbiol.* 14, 1174532. doi: 10.3389/fmicb.2023.1174532

Kesharwani, A. K., Singh, D., Kulshreshtha, A., Kashyap, A. S., Avasthi, A. S., and Geat, N. (2023). Black rot disease incited by Indian race 1 of *Xanthomonas campestris* pv. campestris in Brassica juncea 'Pusa Bold'in India. *Plant Dis.* 107, 212. doi: 10.1094/PDIS-04-22-0738-PDN

Khan, N. A., Dubey, V. K., Mishra, A., Murugesh, T., Dwivedi, D. K., Singh, H. K., et al. (2023). Screening of Indian mustard (Brassica juncea Linn. Czern & Coss) germplasm against mustard aphid, Lipaphis erysimi pseudobrassicae (Kaltenbach 1843). doi: 10.21203/rs.3.rs-3372239/v1

Kruger, N. J. (2009). "The Bradford method for protein quantitation," in *The protein protocols handbook*. (Humana Press Springer), 17–24.

Li, Z., and Wu, W. (2023). Genotype recommendations for high performance and stability based on multiple traits selection across a multi-environment in rapeseed. *Eur. J. Agron*. 145, 126787. doi: 10.1016/j.eja.2023.126787

Mamgain, A., Roychowdhury, R., and Tah, J. (2013). Alternaria pathogenicity and its strategic controls. *Res. J. Biol.* 1, 1–9.

Mathpal, P., Punetha, H., Tewari, A. K., and Agrawal, S. (2016). Biochemical defense mechanism in rapeseed- mustard genotypes against Alternaria blight disease. *J. Oilseed Brassica* 1, 87–94.

McCune, D. C., and Galston, A. W. (1959). Inverse Effects of Gibberellin on Peroxidase Activity and Growth in Dwarf Strains of Peas and Corn. *Plant Physiol.* 34, 416. doi: 10.1104/pp.34.4.416

Meena, M., Zehra, A., Dubey, M. K., Aamir, M., Gupta, V. K., Upadhyay, R. S., et al. (2016a). Comparative evaluation of biochemical changes in tomato (Lycopersicon esculentum Mill.) infected by Alternaria alternata and its toxic metabolites (TeA, AOH, and AME). *Front. Plant Sci.* 7, 1408. doi: 10.3389/fpls.2016.01408

Meena, P. D., Awasthi, R. P., Chattopadhyay, C., Kolte, S. J., and Kumar, A. (2016b). Alternaria blight: a chronic disease in rapeseed-mustard. *J. Oilseed Brassica* 1, 1–11.

Meena, P. D., Sujith Kumar, M. S., Meena, H. S., Jambhulkar, S., Gohartaj, Pathak, D., et al. (2020). Confirmation of induced tolerance to Alternaria blight disease in Indian mustard (Brassica juncea L.). *Appl. Biochem. Biotechnol*. 192, 965–978. doi: 10.1007/s12010-020-03362-2

Misra, R. S., Sharma, K., Mishra, A. K., and Sriram, S. (2008). Biochemical alterations induced in Taro in response to Phytophthora colocasiae infection. *Adv. Nat. Appl. Sci.* 2, 112–121.

Mittler, R. (2002). Oxidative stress, antioxidants and stress tolerance. *Trends Plant Sci.* 7, 405–410. doi: 10.1016/S1360-1385%2802%2902312-9

Møller, I. M., Jensen, P. E., and Hansson, A. (2007). Oxidative modifications to cellular components in plants. *Annu. Rev. Plant Biol.* 58, 459–481. doi: 10.1146/ANNUREV.ARPLANT.58.032806.103946

Moormann, J., Heinemann, B., and Hildebrandt, T. M. (2022). News about amino acid metabolism in plant–microbe interactions. doi: 10.1016/j.tibs.2022.07.0018

Munir, S., Shahzad, A. N., and Qureshi, M. K. (2020). Acuities into tolerance mechanisms via different bioassay during Brassicaceae-Alternaria brassicicola interaction and its impact on yield. *PloS One*. 15, e0242545. doi: 10.1371/journal.pone.0242545

Narware, J., Singh, S. P., Manzar, N., and Kashyap, A. S. (2023). Biogenic synthesis, characterization, and evaluation of synthesized nanoparticles against the pathogenic fungus Alternaria solani. *Front. Microbiol.* 14, 1159251. doi: 10.3389/fmicb.2023.1159251

Naznin, S., Kawochar, M. A., Sultana, S., and Bhuiyan, M. S. R. (2015). Genetic variability, character association and path analysis in Brassica rapa L. Genotypes. *Bangladesh J. Agric. Res*, 305–323. doi: 10.3329/BJAR.V40I2.24570

Nowakowska, M., Wrzesińska, M., Kamiński, P., Szczechura, W., Lichocka, M., Tartanus, M., et al. (2019). Alternaria brassicicola–Brassicaceae pathosystem: insights into the infection process and resistance mechanisms under optimized artificial bio-assay. *Eur. J. Plant Pathol*153, 131–151. doi: 10.1007/s10658-018-1548-y

Nowicki, M., Nowakowska, M., Niezgoda, A., and Kozik, E. (2012). Alternaria black spot of crucifers: symptoms, importance of disease, and perspectives of resistance breeding. *Vege. Crops Res. Bull* 76, 5–19. doi: 10.2478/V10032-012-0001-6

Onifade, A. K., and Jeff-Agboola, Y. A. (2003). Effect of fungal infection on proximate nutrient composition of coconut (Cocos nucifera Linn) fruit. *J. Food Agric. Environ.* 1, 141–142.

Pandey, V., Tewari, A. K., and Saxena, D. (2018). Activities of defensive antioxidant enzymes and biochemical compounds induced by bioagents in Indian mustard against Alternaria blight. *Proc. Natl. Acad. Sci. India Sect. B: Biol. Sci*. 88, 1507–1516. doi: 10.1007/s40011-017-0888-2

Poli, Y., Nallamothu, V., Balakrishnan, D., Ramesh, P., Desiraju, S., Mangrauthia, S. K., et al. (2018). Increased catalase activity and maintenance of photosystem II distinguishes high-yield mutants from low-yield mutants of rice var. Nagina22 under low-phosphorus stress. *Front. Plant Sci*. 9, 1543. doi: 10.3389/fpls.2018.01543

Rathore, S. S., Shekhawat, K., Kandpal, B. K., Premi, O. P., Singh, S. P., Chand, G., et al. (2015). Annals of Plant and Soil Research 17 (1): 1-12 (2015) Sulphur management for increased productivity of Indian mustard: a review. *Ann. Plant Soil Res.* 17, 1–12.

Reddy, V. K., and Immanuelraj, K. T. (2017). Area, production, yield trends and pattern of oilseeds growth in India. *Econ. Affairs*. 62, 327. doi: 10.5958/0976-4666.2017.00016.x

Saccardo, P. A. (1886). Sylloge fungorum ominum hucusque cognitorum. *Pavia* 4, 1–807.

Saed-Moucheshi, A., Fasihfar, E., Hasheminasab, H., Rahmani, A., and Ahmadi, A. (2013). A review on applied multivariate statistical techniques in agriculture and plant science. *Int. J. Agron. Plant Product.* 4, 127–141.

Sambrook, J., and Russell, D. W. (2001). *Molecular cloning: A laboratory manual* (New York: Cold Spring Harbor Laboratory Press).

Sharma, P., Jha, A. B., Dubey, R. S., and Pessarakli, M. (2012). Reactive oxygen species, oxidative damage, and antioxidative defense mechanism in plants under stressful conditions. *J. Bot*. 2012, 217037. doi: 10.1155/2012%2F217037

Sharma, M., Deep, S., Bhati, D. S., Chowdappa, P., Selvamani, P., and Sharma, P. (2013). Mophological, cultural, pathogenic and molecular studies of Alternaria brassicae infecting cauliflower and mustard in India. *Afr. J. Microbiol. Res.* 7, 3351–3363. doi: 10.5897/AJMR12.593

Shereefa, L. A. H., and Kumaraswamy, M. (2016). Reactive oxygen species and ascorbate–glutathione interplay in signaling and stress responses in Sesamum orientale L. against Alternaria sesami (Kawamura) Mohanty and Behera. *J. Saudi*. 15, 48–56.

Singh, M., Avtar, R., Lakra, N., Hooda, E., Singh, V. K., and Bishnoi, M. (2021a). Genetic and proteomic basis of sclerotinia stem rot resistance in Indian mustard [Brassica juncea (L.) czern & coss.]. *Genes*. 12, 1784. doi: 10.3390/genes12111784

Singh, M., Avtar, R., Pal, A., Punia, R., Singh, V. K., Bishnoi, M., et al. (2020b). Genotype-specific antioxidant responses and assessment of resistance against Sclerotinia sclerotiorum causing Sclerotinia rot in Indian mustard. *Pathogens*. 9, 892. doi: 10.3390/pathogens9110892

Singh, B. K., Choudhary, S. B., Yadav, S., Malhotra, E. V., Rani, R., Ambawat, S., et al. (2018). Genetic structure identification and assessment of interrelationships between Brassica and allied genera using newly developed genic-SSRs of Indian Mustard (Brassica juncea L.). *Ind. Crops Prod*. 113, 111–120. doi: 10.1016/J.INDCROP.2018.01.023

Singh, R., Dwivedi, D. K., Mishra, A., and Singh, A. (2024). Development of sheath blight resistant genotype of rice by mutation breeding and gene expression profiling after inoculation of Rhizoctonia solani. *Cereal Res. Commun*., 1–12. doi: 10.1007/s42976-024-00533-3

Singh, L., Sharma, D., Parmar, N., Singh, K. H., Jain, R., Rai, P. K., et al. (2020a). Genetic diversity studies in Indian mustard (Brassica juncea L. Czern & Coss) using molecular markers. *Brassica Improve.: Molecul. Genet. Genom. Perspect*. doi: 10.1007/978-3-030-34694-2_11

Singh, B., Sran, A. S., and Sohi, G. S. (2021b). Innovative strategies to develop abiotic and biotic stress tolerance in Mustard (Brassicaceae). *Brassica Breed. Biotechnol*., 41–48. doi: 10.5772/intechopen.95973

Summuna, B., Gupta, S., Gupta, M., Singh, R., and Razdan, V. K. (2012). Prevalence of Alternaria blight of mustard and sources of its resistance in Jammu Division of Jammu & Kashmir. *Indian Phytopath* 65, 406–408.

Titarenko, E., Hargreaves, J., Keon, J., and Gurr, S. J. (1993). Defence-related gene expression in barley coleoptile cells following infection by Septoria nodorum. *Mech. Plant Defense Responses*., 308–311. doi: 10.1007/978-94-011-1737-1_75

Torres, M. A., Jones, J. D., and Dangl, J. L. (2006). Reactive oxygen species signaling in response to pathogens. *Plant Physiol*. doi: 10.1104/pp.106.079467

Tyagi, M., Kayastha, A. M., and Sinha, B. (1998). The role of phenolics and peroxidase in resistance to Alternaria triticina in bread wheat (Triticum aestivum L.). *J. Agron. Crop Sci*. 181, 29–34. doi: 10.1111/J.1439-037X.1998.TB00394.X

Tyagi, P. K., and Singh, R. (2016). *Rapeseed and Mustard: Breeding objectives of Rapeseed and Mustard* (Saarbrucken, Germany: LAP LAMBERT Academic Publication).

Yadav, P., Mir, Z. A., Ali, S., Papolu, P. K., and Grover, A. (2020). A combined transcriptional, biochemical and histopathological study unravels the complexity of Alternaria resistance and susceptibility in Brassica coenospecies. *Fungal Biol*. 124, 44–53. doi: 10.1016/j.funbio.2019.11.002

Zemanová, V., Pavlík, M., and Pavlíková, D. (2014). The significance of methionine, histidine and tryptophan in plant responses and adaptation to cadmium stress. *Plant Soil Environ*., doi: 10.17221/544%2F2014-PSE

TYPE Original Research
PUBLISHED 27 September 2024
DOI 10.3389/fpls.2024.1418201

OPEN ACCESS

EDITED BY
Yongliang Qiao,
University of Adelaide, Australia

REVIEWED BY
Xiujun Zhang,
Chinese Academy of Sciences (CAS), China
Ning Yang,
Jiangsu University, China
Hariharan Shanmugasundaram,
Vardhaman College of Engineering, India

*CORRESPONDENCE
Jianwen Guo
guojw@dgut.edu.cn

RECEIVED 16 April 2024
ACCEPTED 04 September 2024
PUBLISHED 27 September 2024

CITATION
Chen J, Guo J, Zhang H, Liang Z and Wang S (2024) Weakly supervised localization model for plant disease based on Siamese networks. *Front. Plant Sci.* 15:1418201.
doi: 10.3389/fpls.2024.1418201

Weakly supervised localization model for plant disease based on Siamese networks

Jiyang Chen, Jianwen Guo*, Hewei Zhang,
Zhixiang Liang and Shuai Wang

Dongguan University of Technology, Dongguan, China

Problems: Plant diseases significantly impact crop growth and yield. The variability and unpredictability of symptoms postinfection increase the complexity of image-based disease detection methods, leading to a higher false alarm rate.

Aim: To address this challenge, we have developed an efficient, weakly supervised agricultural disease localization model using Siamese neural networks.

Methods: This model innovatively employs a Siamese network structure with a weight-sharing mechanism to effectively capture the visual differences in plants affected by diseases. Combined with our proprietary Agricultural Disease Precise Localization Class Activation Mapping algorithm (ADPL-CAM), the model can accurately identify areas affected by diseases, achieving effective localization of plant diseases.

Results and conclusion: The results showed that ADPL-CAM performed the best on all network architectures. On ResNet50, ADPL-CAM's top-1 accuracy was 3.96% higher than GradCAM and 2.77% higher than SmoothCAM; the average Intersection over Union (IoU) is 27.09% higher than GradCAM and 19.63% higher than SmoothCAM. Under the SPDNet architecture, ADPL-CAM achieves a top-1 accuracy of 54.29% and an average IoU of 67.5%, outperforming other CAM methods in all metrics. It can accurately and promptly identify and locate diseased leaves in crops.

KEYWORDS
plant disease, deep learning, Siamese networks, weakly supervised localization, class activation mapping

1 Introduction

Disease detection in agriculture plays a crucial role in ensuring crop health and maximizing yields. Traditionally, manual inspection and experience-based judgment have been used to identify diseases, but these methods often lack efficiency and accuracy, particularly for minor or inconspicuous ailments. With the advancement of machine vision and deep learning models, particularly Convolutional Neural Networks (CNNs) (LeCun et al., 2015), significant progress has been made in computer vision techniques for agricultural disease detection (Ferentinos, 2018). Utilizing these cutting-edge technologies for disease image classification has greatly improved the accuracy and robustness of detection.

However, current deep learning vision detection models still face challenges when dealing with the diversity and randomness of plant diseases. For example, diversity can lead to poor adaptability of traditional algorithms at different scales, resulting in missed or false detections. Diseases might be difficult to detect due to variations in the size, shape, or color of plant leaves or due to environmental factors such as lighting and occlusion. Traditional CNN architectures often perform poorly in addressing these issues (Fuentes et al., 2017) as they are designed with fixed scales and field-of-view sizes, which do not adapt well to varying sizes of disease features, especially in large-scale agricultural fields. Moreover, conventional disease detection methods require extensive annotation of datasets, which increases training costs and limits application scenarios. In contrast, weakly supervised learning can effectively detect using existing image category labels, significantly reducing the reliance on detailed annotations. Current weak supervision localization techniques primarily rely on multiple instance learning (Carbonneau et al., 2018) and Class Activation Mapping methods (Zhou et al., 2016), which train networks using image-level labels but often focus only on local features, making it difficult to cover the entire target and handle multiple instances of the same category.

To address these challenges, we propose an innovative detection model based on Siamese neural networks and weak supervision localization techniques, transforming the disease detection problem into a task of visual difference identification. By integrating multiscale features and implementing a refined weighting strategy, we have enhanced the accuracy and efficiency of disease identification. We use the ADPL-Class Activation Map (CAM) technique to generate heatmaps for precise disease localization and employ Non-Maximum Suppression (NMS) technology to handle multiple case issues, effectively improving the model's performance in complex environments.

The latter part of this article will detail the relevant research work, foundational knowledge of Siamese networks and Class Activation Mapping techniques, describe our model architecture and experimental design, and demonstrate the effectiveness of our model through experimental results. We will discuss these results, emphasizing their significance in the field of intelligent agricultural disease detection, and outline future research directions.

2 Related work

2.1 Advances in plant disease detection research using deep learning

Early-stage plant diseases refer to diseases or diseases that occur in the early stages of plant growth, usually in the early stages after infection. Their symptoms may not be easily observed or recognized but may have potential impacts on the health and growth status of plants. The automatic recognition of early-stage plant disease images has traditionally relied on conventional machine learning techniques such as K-Nearest Neighbors (KNN) (Kumar et al., 2020), Support Vector Machines (SVM) (Rumpf et al., 2010), and Deep Forest methods (Zhou and Feng, 2017). However, with the advent of deep learning models, intelligent diagnostic methods based on these technologies have become the mainstream approach for image recognition (Sankaran et al., 2010) and have been increasingly applied to crops like corn, wheat, citrus, and potatoes (Ferentinos, 2018). For instance, (Mohanty et al., 2016) have demonstrated the accuracy and robustness of deep learning in classifying a vast array of plant disease images using CNNs. Similarly, the deep learning models developed by (Sladojevic et al., 2016) and the PlantXViT model introduced by Poornima et al (Poornima and Pushpalatha, 2021), which combines CNNs with Vision Transformers, have achieved notable success in plant disease recognition.

To address the shortage of datasets, researchers have explored small sample learning: (Li et al., 2023) investigated the potential of Diffusion Models (DDPM), Swin-Transformer models, and transfer learning for diagnosing citrus diseases with limited datasets. (Lee et al., 2018) designed two new data generation methods based on plant canopy simulation and Generative Adversarial Networks (GANs), which successfully handled the challenging task of segmenting apple scab disease in apple tree canopy images, showing promising results on small datasets. In terms of transfer learning, (Atila et al., 2020) proposed an efficient network of deep learning models for classifying plant leaf diseases, trained using the transfer learning approach on the EfficientNet architecture and other deep learning models. (Zj et al., 2019) enhanced the VGG16 model with multitask learning concepts and then applied transfer learning with pretrained models from ImageNet, effectively recognizing diseases in rice and wheat leaves, and providing a reliable method for identifying multiple plant leaf diseases. (Chen et al., 2018) explored deep convolutional neural network transfer learning to identify plant leaf diseases, considering using pretrained models from large-scale datasets and then transferring them to specific tasks.

Deep learning models still face challenges in handling the multiscale and randomness aspects of diseases. Diseases may appear on plants in various sizes, shapes, and colors, making it difficult for traditional algorithms to adapt to different scales and potentially leading to missed or false detections. Additionally, the same disease might appear differently on various plants and be influenced by environmental factors such as lighting and occlusion, increasing the likelihood of false positives. This presents significant challenges for disease detection, especially in large-scale agricultural environments. To overcome these issues, new solutions are being explored: (Singh et al., 2018) used Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997) to detect moisture stress in chickpea bud images, showcasing the potential of LSTM networks in multiscale diagnosis. (Mahlein, 2016) discussed methods for plant leaf disease detection using imaging sensors, highlighting the randomness in disease manifestation and proposing solutions. (Sumaya and Uddin, 2021) emphasized the importance of using deep learning for multiscale diagnosis and made progress in diagnosing various plant leaf diseases.

We believe that traditional CNN architectures, designed with predetermined image scales (He et al., 2016) and fixed receptive fields, struggle to adapt to disease spots of varying sizes (Liu et al., 2016). Additionally, these networks are not well-suited for spatial transformations such as rotation and scaling (Jaderberg et al., 2015), which can vary significantly across different plants, resulting in poor performance in such tasks. Moreover, these networks may lose crucial detailed information necessary for identification while extracting high-level semantic information (Zeiler and Fergus, 2014).

2.2 Siamese network

The Siamese network, as illustrated in Figure 1, is a specialized neural network architecture designed for image comparison and verification tasks. This architecture is characterized by its two parallel branches, mirroring each other and sharing identical parameters, much like the interconnected nature of Siamese twins—hence the name. The primary benefit of this shared-parameter design is that it ensures both branches carry out the same transformations. Consequently, each input image is transformed into a feature vector, enabling a direct and equitable comparison.

To enhance the network's ability to accurately measure image similarity, loss functions such as ContrastiveLoss (Hadsell et al., 2006) and TripletLoss (Schroff et al., 2015) are employed during the network's training phase. These functions are crucial for the fine-tuning of network parameters, directly impacting the precision of similarity measurements.

Upon inputting two distinct photographs, the network analyzes each one independently. Each branch meticulously deconstructs its image into a detailed array of features—lines, edges, textures, and patterns—that define the image's unique identity. These features are then transcribed into vectors, comprehensive numerical sequences that represent the images' visual characteristics. Thanks to the Siamese configuration, this feature extraction process is consistently executed across both branches, laying the foundation for a balanced comparison.

Siamese networks have proven their effectiveness in a spectrum of applications. For instance, the DeepFace system (Taigman et al., 2014) harnesses a Siamese network for facial recognition, demonstrating its prowess in complex identification tasks. Similarly, the SiamFC tracker (Bertinetto et al., 2016) showcases the power of Siamese networks in real-time object tracking in video streams. Beyond these, the architecture has shown exceptional performance in recognizing Chinese handwritten characters (Zhang et al., 2017) and evaluating semantic similarity in natural language processing (Mueller and Thyagarajan, 2016).

In our research, we leverage the Siamese network's dual-branch feature extraction capability by inputting image pairs that exhibit similarity. This approach allows us to produce highly accurate feature maps that are essential for precisely pinpointing object locations within images. By training the network with pairs of known similar images, we enhance its proficiency in detecting fine distinctions and shared characteristics between images, which is critical for tasks that demand exact localization.

2.3 Class activation map

The CAM is a technique used in imaging to interpret and visualize the decision-making process of CNNs. It is based on a

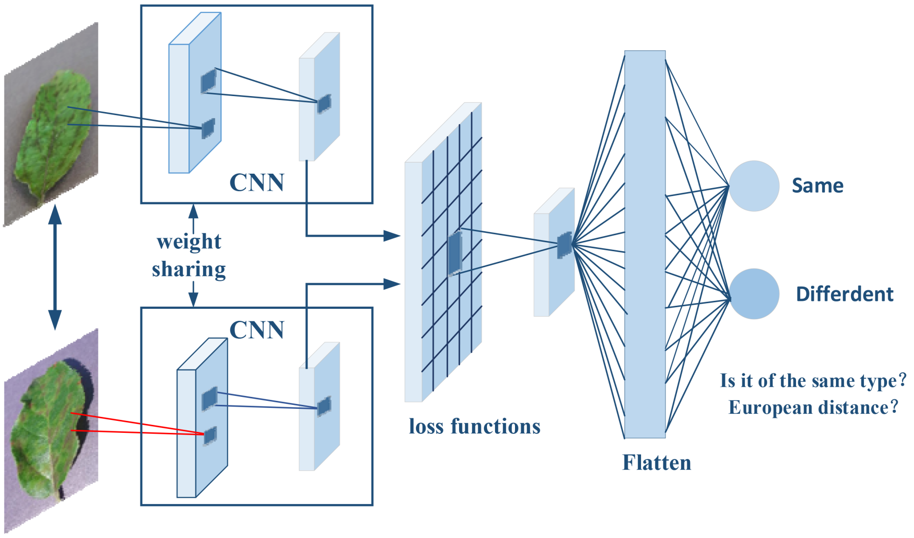


FIGURE 1
The architecture of the Siamese network was used in this study. The network independently analyzes each input image and deconstructs it into a detailed array of features such as lines, edges, textures, and patterns. These features are then transcribed into vectors, representing the images' visual characteristics. The network is trained with pairs of known similar images to enhance its proficiency in detecting fine distinctions and shared characteristics.

critical insight: classification networks not only extract categorical information from images but also implicitly encode spatial location information of targets. CAM generates heatmaps for specific categories by combining the outputs of a Global Average Pooling (GAP) layer with the feature maps from the last convolutional layer, visually indicating the target locations. The implementation process is illustrated in Figure 2.

Initially, an input image is processed through a CNN, producing a set of feature maps. Following the last convolutional layer of the CNN, a GAP layer is employed to calculate the average activation of each feature map, as shown in Equation 1:

$$F_k = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} f_{ij}^k \tag{1}$$

Here, f_{ij}^k denotes the activation value at position (i, j) on the kth feature map, with H and W representing the height and width of the feature map, respectively.

Subsequently, the output from the GAP layer is connected to a fully connected layer, whose weight matrix is used to compute the scores for each category:

$$S_c = \sum_k W_{k,c} \cdot F_k \tag{2}$$

In this formula, W is the weight matrix of the fully connected layer, $W_{k,c}$ represents the weight between the kth feature map and the cth category, and F_k is the average activation of the kth feature map.

Finally, by multiplying each feature map's activation values by their corresponding category weights and summing them up, a class activation map is generated:

$$M_c(i,j) = \sum_k W_{k,c} \cdot f_{i,j}^k \tag{3}$$

This map is the same size as the original image and uses grayscale values to indicate the significance of different areas for the network's prediction. Higher scores indicate greater contributions to the final classification outcome. Converting this grayscale map to a color map can more clearly show which parts of the image are most focused on by the network and which areas are most predictive for a particular category.

The CAM method has been successfully applied in many research tasks, such as using CAM to locate pneumonia in chest X-ray images (Wang et al., 2017). Researchers have developed several variants of CAM, such as Grad-CAM (Selvaraju et al., 2017), which uses category-specific gradient information to weight feature maps, extending CAM's applicability to more CNN architectures. Score-CAM (Wang et al., 2020) and Layer-CAM (Jiang et al., 2020) enhance the usability and interpretative power of CAM methods through model scoring and specific layer visualization, respectively.

3 Plant disease localization model based on Siamese neural networks

The traditional backbone networks often struggle to adequately recognize the subtle variances present in crop disease symptoms. To address this issue, we present SPDNet, a Siamese neural network-based method for weakly supervised localization of plant diseases. SPDNet is ingeniously crafted to tackle the challenges associated with the nuanced differences in infection symptoms and the presence of multiscale features.

The SPDNet model begins by inputting pairs of images that exhibit similar plant disease symptoms, with each pair comprising a query image and a reference image. A Siamese neural network, initialized with shared parameters, processes both the query and reference images. The query image is fed into the first subnet to extract feature maps, while the reference image is processed through the second subnet for feature extraction. Subsequently, a pyramid structure is employed to fuse the multiscale feature maps obtained from both the query and reference images, ensuring a comprehensive representation of disease symptoms across

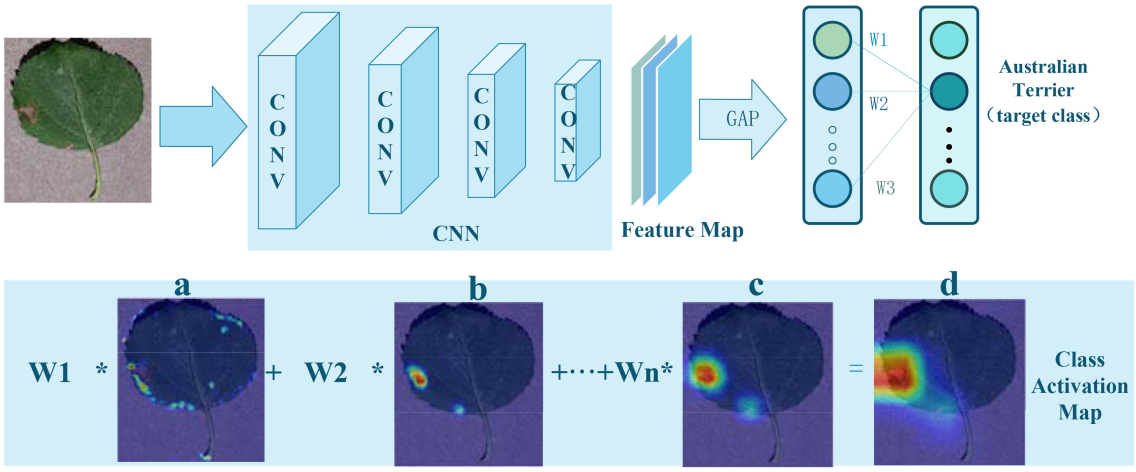


FIGURE 2
Implementation of Class Activation Mapping: **(A–C)**, CAM color heatmaps; **(D)**, original image overlaid with the CAM heatmap.

different scales. These fused multiscale feature maps are then input into the ADPL-CAM-based weakly supervised localization module. This module autonomously generates pseudo-detection bounding boxes to identify potential disease symptom regions. Following this, the ADPL-CAM module's localization results are used to predict bounding boxes around the disease symptoms in the query image, with the disease locations highlighted in red on the heatmap. The SPDNet model is trained using weakly supervised learning methods, leveraging pseudo-labels generated by the ADPL-CAM module instead of precise annotations. During the iterative training process, the model is continuously refined to improve accuracy in disease localization.

The architecture of SPDNet, as illustrated in Figure 3, leverages shared parameters within its Siamese framework to enhance the model's sensitivity to minor discrepancies between input images. The network processes pairs of disease images that share similar characteristics, using one image as the target and the other as a referential guide for localization. This dual-image input strategy enables SPDNet to develop more refined and distinctive feature representations, crucial for distinguishing between subtle disease symptoms.

Within the Siamese network, a pyramid structure is employed to amalgamate multiscale information extracted at various layers, ensuring a thorough representation of disease symptomatology across different scales.

The ADPL-CAM-based weakly supervised localization module is a core component of SPDNet, tailored for effective internal feature mapping during the detection and localization of plant diseases. It autonomously generates pseudo-detection bounding boxes, thereby diminishing the dependency on precisely annotated data. This module's capability to produce pseudo-labels is pivotal for the generation of bounding boxes and the execution of weakly supervised localization tasks.

The employment of weakly supervised learning methodologies is a strategic choice for training SPDNet models. Given the laborious and sometimes unfeasible nature of acquiring fully annotated datasets in the agricultural domain, the weakly supervised approach is exceptionally pertinent. It facilitates the training of SPDNet with a reduced need for meticulously labeled data. The generation of pseudo-labels by SPDNet's localization module acts as a surrogate for detailed annotations, making the training process more scalable and economically viable while preserving effectiveness. Through the study of SPDNet, we have reduced the dependence on precisely annotated data, which enables it to work effectively even in situations where annotated data are scarce, breaking free from the limitations of supervised learning methods like PiTLiD (Liu and Zhang, 2022) on small sample datasets.

3.1 SPDNet Siamese network development

The development of the SPDNet Siamese network aims to overcome a series of challenges faced by traditional CNNs when processing crop disease images, particularly issues related to handling multiscale image features, adapting to spatial transformations like rotation and scaling, and preserving detailed information. The SPDNet employs a dual-branch structure to extract complementary features, which effectively deals with spatial transformations in disease areas and enhances the robustness of localization results. The architecture of the SPDNet Siamese network is shown in Figure 4, featuring this dual-branch structure.

The feature extraction part of the network utilizes a Feature Pyramid Structure (Lin et al., 2017), a strategy for extracting and integrating information across multiple scales. This allows for a comprehensive capture of disease symptom features of varying sizes. By merging features across scales, the network adaptively responds to changes in the size of disease areas, enhancing the robustness of the localization outcomes. In the higher layers, SPDNet incorporates both GAP and Global Max Pooling (GMP) (Zhou et al., 2016) to fuse features, which highlights the most significant features while also considering the average characteristics of the images, thus balancing global and local information. Moreover, SPDNet introduces a Multi-Scale Excitation (MSE) module to boost its representational

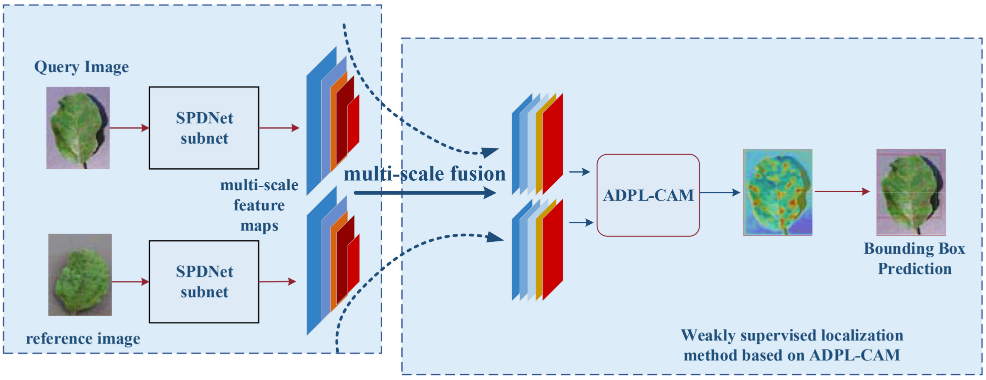


FIGURE 3
Flowchart of a weakly supervised plant disease localization model based on Siamese neural networks, where the blue represents the Siamese neural feature extraction network, and the red denotes the weakly supervised localization module based on ADPL-CAM. In the heatmap, red indicates the location of the disease.

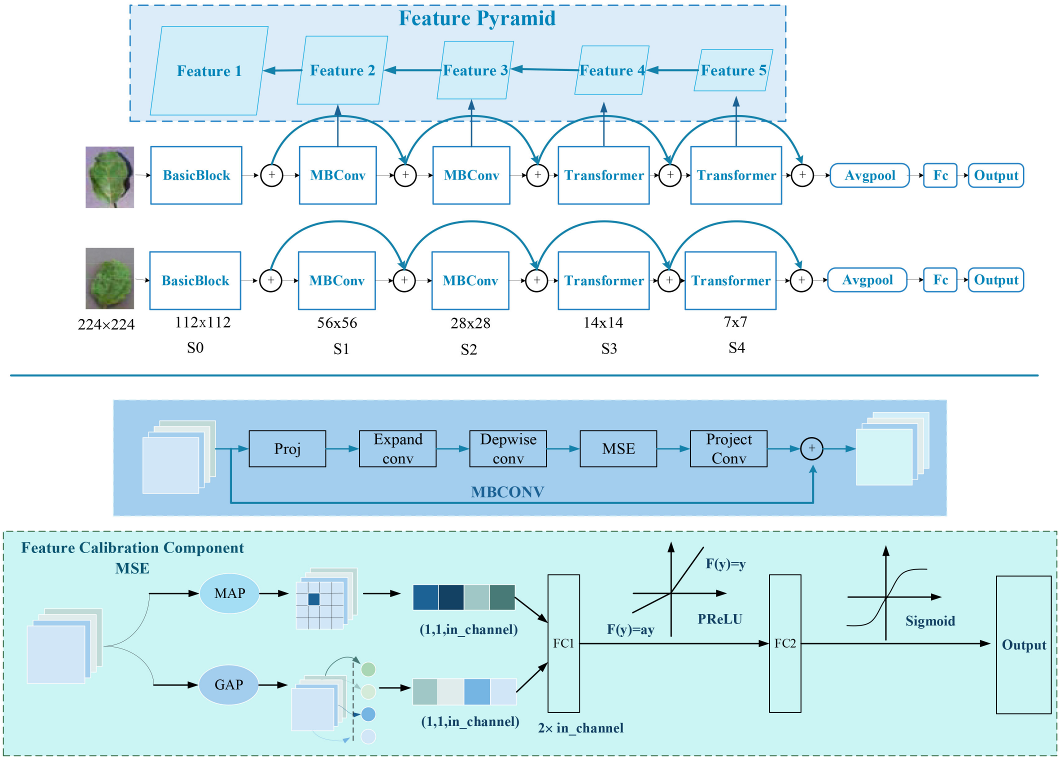


FIGURE 4
Architecture diagram of the SPDNet Siamese network component.

power by adaptively adjusting the weights of different feature channels, focusing on the most pertinent features. The network also includes Parametric Rectified Linear Unit (PReLU) (He et al., 2015) as a nonlinear activation function and a dropout mechanism (Srivastava et al., 2014) for regularization, further enhancing the network's learning capabilities and feature robustness.

3.1.1 Detailed component descriptions

3.1.1.1 Basic block

This consists of a 3 × 3 convolution, batch normalization, and ReLU activation:

$$y = \mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}(x))) \tag{4}$$

where Conv represents 3 × 3 convolution, BN denotes batch normalization, and ReLU is the activation function.

3.1.1.2 Feature Calibration Component MSE

The Feature Calibration Component MSE (Hu et al., 2018) facilitates the modeling of the importance across different semantic feature channels. By utilizing GAP and GMP to extract the average vector vavg and maximum vectorvmax, respectively, and then concatenating them along the channel dimension, the resulting vector is input into a fully connected network to learn channel correlations.

The computation of the channel attention vector is formulated as follows:

$$Z = [GAP(x); GMP(x)] \tag{5}$$

$$A = \sigma(W_2(\delta(W_1 Z))) \tag{6}$$

where σ denotes the Sigmoid function, δ represents the PReLU activation function, [;] indicates the concatenation operation, and $W1$ and $W2$ are learnable weights.

3.1.1.3 More detailed structure and parameter selection

GAP and GMP are employed to compress each channel of the input feature map into a single scalar value, representing the global average and global maximum of that channel, respectively. The pooled features (concatenated results of GAP and GMP, with a dimension twice the number of input channels) are mapped to a hidden layer. The hidden layer's channel count is set to 25% of the input channel count (controlled by the expansion parameter). The weights of the first fully connected layer (FC1) are initialized using the He initialization method. Batch normalization is applied to stabilize the training process. Dropout is used to prevent overfitting, with the dropout rate set to 0.5. The PReLU activation function is applied after the first fully connected layer. The Sigmoid activation function is applied after the second fully connected layer, compressing the output values to the range [0, 1]. The weights of the second fully connected layer (FC2) are initialized using the Xavier initialization method.

3.1.1.4 MBConv module

A mobile-optimized bottleneck residual block structure that introduces the MSE mechanism between input and output (Sandler et al., 2018):

$$x \leftarrow \mathrm{ReLU}\Big(\mathrm{BN}\Big(\mathrm{DWConv}\Big(\mathrm{ReLU}\big(\mathrm{BN}(\mathrm{Expand}(x))\big)\Big)\Big)\Big) \tag{7}$$

$$attention = \sigma(MSE(x)) \tag{8}$$

$$x \leftarrow x + Proj(x \odot \mathrm{attention}) \tag{9}$$

where Expand represents channel expansion via 1x1 convolution, DWConv stands for depthwise separable convolution, and Proj is a 1 × 1 convolution projection.

3.1.1.5 Transformer module

Based on a conventional Attention and FFN transformer encoder structure, the main process involves MST, LayerNorm, Attention computation, and residual connections (Vaswani et al., 2017):

$$x_1 = MST(x) \tag{10}$$

$$z_1 = Attention(LN(x_1)) + x_2 \tag{11}$$

$$z_2 = FFN(LN(z_1)) + z_1 \tag{12}$$

where ×2 is a downsampling or equivalent Identity, LN denotes LayerNorm normalization, and MST represents multiscale integration of different sampling information.

3.1.1.6 Feature Pyramid Structure

After extracting features at each level, a 1 × 1 convolution processes internally before upsampling is combined with the previous layer's feature map, and a 3 × 3 convolution smoothly integrates to ensure consistent output scale and channel number (Lin et al., 2017):

$$C_i = Conv_{1x1} \tag{13}$$

$$P_i = Upsample(P_{i+1}) + C_i \tag{14}$$

$$FPN_i = Conv_{3\times 3}(P_i) \tag{15}$$

By employing a complex design with multiple modules operating at different sampling rates, the SPDNet Siamese network not only captures disease features across various scales but also effectively minimizes localization errors due to changes in disease appearance through its dual-branch structure's complementary characteristics, demonstrating exceptional performance.

3.2 Weakly supervised localization based on ADPL-CAM

To enhance the accuracy and robustness of disease symptom localization in SPDNet, this study introduces an innovative Class Activation Mapping method named Agricultural Disease Precise Localization Class Activation Map (ADPL-CAM). The overall detailed workflow diagram is shown in Figure 5. This method was developed with an understanding of the limitations of traditional CAM technologies in handling agricultural disease images, especially their inadequacies in dealing with multiscale features and background noise. It utilizes multiscale feature maps generated by the two branches of the SPDNet Siamese network. Based on a pair of similar image inputs, categorized into a reference image and a query image (the actual target frame output image), where the reference image enhances the features of the query image. ADPL-CAM extracts two feature matrices and effectively merges feature maps from both branches using upsampling and interpolation methods.

Subsequently, these feature maps undergo pooling to activate hierarchical weight, using weights to absorb the importance of features from different network layers. Ultimately, ADPL-CAM undertakes token learning for the reference image's features: employing global maximum pooling to extract semantic information (i.e., tokens) and then fusing these tokens with the feature maps of the query image. Through token-based fusion, the activation map of the query image prominently represents similar semantic features. This strategy not only intensifies the model's focus on the disease target areas but also significantly reduces its sensitivity to background noise.

Moreover, ADPL-CAM incorporates a NMS strategy to optimize the generation of localization boxes. NMS identifies the local maxima within each potential target area and filters out areas with low scores or high overlap through thresholding, thus enabling more accurate delineation of disease areas and effectively reducing misses. This strategy is particularly aimed at localization challenges in scenarios where similar diseases are clustered, greatly enhancing the model's precision and adaptability in complex agricultural settings.

3.2.1 ADPL-CAM multiscale feature map-weighted fusion

The CAM is formulated as a weighted sum of feature maps:

$$CAM = \sum_{i=1}^{N} w_i \cdot F_i \tag{16}$$

where N is the number of feature maps, w_i are weights obtained via the global average pooling layer, and F_i is the feature map at that scale.

3.2.2 Token-based feature learning

Initially, we define the tokenization process for the reference image's feature maps (feature tokenization) to extract representative feature vectors T_i:

$$T_i = GlobalMaxPool(F_i) \tag{17}$$

Here, the GlobalMaxPool operation performs global maximum pooling, traversing each channel of the feature map and retaining only the maximum value per channel, thus forming a compact feature vector. This vector T_i acts as a token, capturing the most critical visual features. Subsequently, we fuse the target feature map G with the token (T_i), resulting in an enhanced feature map:

$$G' = G + \sum_{i=1}^{N} \alpha \cdot T_i \tag{18}$$

where α represents the learned weights, indicating the contribution of different tokens to the target feature map.

3.2.3 Adaptive threshold function for generating box thresholds

$$T(x, y) = \frac{1}{blocksize^2} \sum_{i,j \in neighborhood} I(i, j) - C \tag{19}$$

Here (Bradley and Roth, 2007), $T(x, y)$ is the threshold at the pixel location (x, y), I (i, j) is the value of the pixels in the neighborhood, C is a constant used to adjust the threshold, and blocksize squared represents the size of the neighborhood considered for local threshold computation.

3.2.4 Non-maximum suppression

Define a set of detection boxes (D = d1,d2,…,dn), each with a corresponding confidence score (si), select the box ($d$max) with the highest score from (D). Calculate the Intersection over Union (IoU) with ($d$max) for the other boxes and remove those with high overlap. Repeat this process until only one box remains.

Thus, ADPL-CAM not only enhances the handling of multiscale features but also improves the accuracy of disease symptom localization, providing robust technical support for precise agricultural disease diagnosis.

4 Experiments and results

4.1 Experimental design

The model's effectiveness is assessed using three main metrics: Top-K Positioning Accuracy, GT-Known Positioning Accuracy, and Average Intersection over Union (Average IoU).

Top-K Positioning Accuracy is defined as the condition where the correct category is among the top-K categories predicted by the model and the IoU between the model's predicted bounding box and the actual bounding box exceeds a specified threshold (set at 0.5). If these conditions are met, the prediction is considered correct.

GT-Known Positioning Accuracy measures whether the model can accurately locate the object when the true category is known. The prediction is deemed accurate if the IoU between the predicted and actual bounding boxes exceeds a predetermined threshold.

Average IoU calculates the mean IoU value between all predicted and actual bounding boxes across all test images to gauge the model's overall localization precision.

We selected Top-K Positioning Accuracy, GT-Known Positioning Accuracy, and Average IoU as our principal metrics for evaluation due to their recognized efficacy and standardization in assessing both classification and localization performances within the field of computer vision. Top-K Positioning Accuracy holds particular significance for applications in the real world, where the ability to generate multiple plausible predictions is often more beneficial than pinpoint accuracy in classification. This metric ensures that the correct category is listed among the top contenders, while the associated IoU threshold criterion guarantees precise object localization within the imagery—a critical factor for practical implementations such as precision agriculture or automated wildlife monitoring.

GT-Known Positioning Accuracy is deployed to gauge the model's proficiency in object localization when the true category is pre-identified—a typical training and tuning scenario for models engaged in detection tasks. This metric singularly focuses on and evaluates the model's spatial discernment capabilities. The Average IoU, on the other hand, extends to provide a cumulative measure of localization accuracy across all tested instances, offering insight into the model's generalization capabilities across a diverse array of categories and conditions. By integrating these tripartite metrics, we ensure a holistic evaluation of the model's competence in not just accurately classifying objects but also in their precise localization, both of which are indispensable for the practical deployment of such models in scenarios where accurate identification and exact object placement are of paramount importance.

In this work, we used two datasets: to further explore the adaptability of the model to changes in different lighting conditions, crop varieties, and disease stages, we constructed a Multi-Conditional Plant Disease Dataset (MCPDD) based on the PlantVillage dataset. This dataset generates image data for different lighting conditions, crop varieties, and disease stages through image processing and classification, specifically for plant disease detection research. MCPDD contains a total of 42 images of different types and degrees of diseases on grape, potato, and tomato leaves under different lighting conditions. This diversity meets the requirements of plant disease detection at different stages, ensuring full consideration of the subtle semantic features of early diseases.

In contrast, the CUB-200 dataset is a fine-grained image classification dataset focused on various animal species. The ADPL-CAM method leverages its capability to capture semantic features within the same class in images. The CUB-200 dataset is not only informative but also serves as a universal benchmark for fine-grained classification tasks. Therefore, evaluating the ADPL-CAM method on this dataset not only validates its overall effectiveness in capturing similar semantic features and generating accurate localization maps but also reaffirms its robustness in fine-grained classification tasks.

This study assesses the feature semantic extraction capabilities of ADPL-CAM using both the PlantVillage and CUB-200 datasets to comprehensively verify the method's universality and effectiveness. The simulation experiments were conducted on a computer equipped with an RTX A5000 GPU and 24GB VRAM. The experimental environment included PyTorch 1.11.0, CUDA 11.6, cuDNN 8.4.0, and Python 3.9.12. Images were resized to 224 pixels × 224 pixels, and data augmentation techniques such as random rotation and Gaussian blur were applied. The training was performed using the AdamW optimizer with an initial learning rate of 0.01, a minimum learning rate of 0.0001, and a cosine annealing learning rate schedule. The training lasted for 100 epochs with a batch size of 16, and the experiments were conducted under consistent hyperparameter settings.

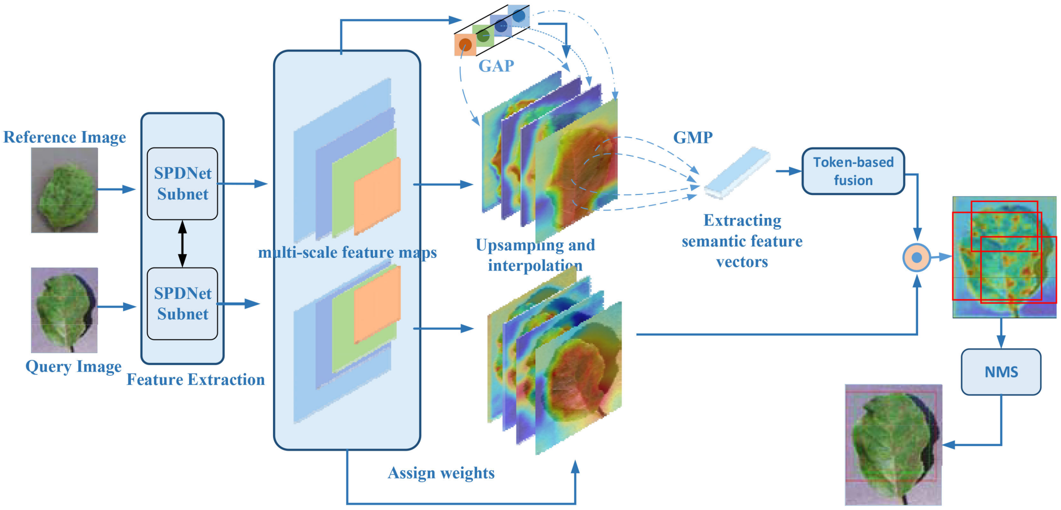


FIGURE 5
ADPL-CAM workflow diagram. Feature extraction: Feature maps are extracted in parallel from the reference and query images via SPNet subnetworks. Hierarchical weight activation: Emphasizes or attenuates the importance of certain features through the network layers. Feature tokenization: The feature maps from the reference image undergo tokenization, transforming these features into a set of compact tokens. Token-based fusion: Tokens from the reference image are fused with the feature maps of the query image, enhancing the feature representation of the query image. Class Activation Mapping: Postfusion, a sequence of processing steps generates the class activation map, highlighting areas of interest in the query image. Non-Maximum Suppression (NMS): To conclude, NMS is applied to the class activation map to suppress overlapping detections, ensuring distinct localization of each detected object.

4.2 Quantitative experiments and discussion

In our quantitative analysis, we used EfficientNet and ResNet50 as comparative classification networks and compared different CAM algorithms, including GradCAM, SmoothCAM, and our proposed ADPL-CAM. The results are shown in Figures 6, 7, and detailed results are shown in Tables 1, 2.

Based on the experimental results, we can draw the following conclusion.

1. Performance comparison: In the CUB-200 and PlantVillage datasets, ADPL-CAM outperformed Grad-CAM and SmoothCAM, especially within the SPDNet framework. Notably, under the SPDNet architecture, ADPL-CAM achieved the best results across all evaluation metrics (accuracy, recall, precision, F1-score, GT-known, and mean IoU). This demonstrates ADPL-CAM's significant advantage in capturing salient regions of target objects and generating more accurate class activation maps.

2. Framework adaptability: The performance improvement of ADPL-CAM in fine-grained tasks when paired with ResNet50 and EfficientNetB0 is relatively modest. This can be attributed to these CNN architectures being primarily designed for general image classification tasks rather than specialized plant disease recognition. However, in the MCPDD dataset, ADPL-CAM's performance is notably outstanding. This indicates that specifically designed network structures, such as SPDNet, can better capture task-specific features in specialized domains.

3. Disease recognition capability: The combination of SPDNet and ADPL-CAM shows significant advantages in plant disease recognition tasks, particularly in terms of various metrics. This suggests that SPDNet can effectively learn feature representations of plant diseases, contributing to more accurate localization maps. Traditional CAM methods (Grad-CAM and SmoothCAM) often perform poorly in complex or challenging disease scenarios, whereas ADPL-CAM maintains high effectiveness, which is crucial for improving model reliability in practical applications. ADPL-CAM excels in covering target areas more comprehensively. Through adaptive multiscale feature fusion and enhanced Class Activation Mapping mechanisms, ADPL-CAM can cover lesion areas more thoroughly, avoiding the omission of key features.

4. Performance deficiencies and potential factors: Despite ADPL-CAM's improvement in overall localization accuracy, this experiment did not validate potential issues in complex scenarios, such as small or overlapping lesion areas, where the model might experience false negatives or misclassifications. The potential reason for this deficiency is that ADPL-CAM's multiscale feature fusion mechanism requires further optimization to better leverage features at different levels. Although we have consciously enhanced fine-grained features in the dataset, the model appears not to have fully learned to recognize subtle disease characteristics. Label-based semantic enhancement may need improvement to distinguish disease samples with minor features. Figure 7 also indicates that ADPL-CAM's localization results are affected by factors such as illumination conditions and crop varieties. Among these, the most significant factor is crop variety, due to the vast semantic differences in characteristics of different plant diseases. Furthermore, ADPL-CAM's generalization ability in small sample datasets might decline, necessitating further optimization of network structures and training strategies to enhance the model's robustness in small sample scenarios.

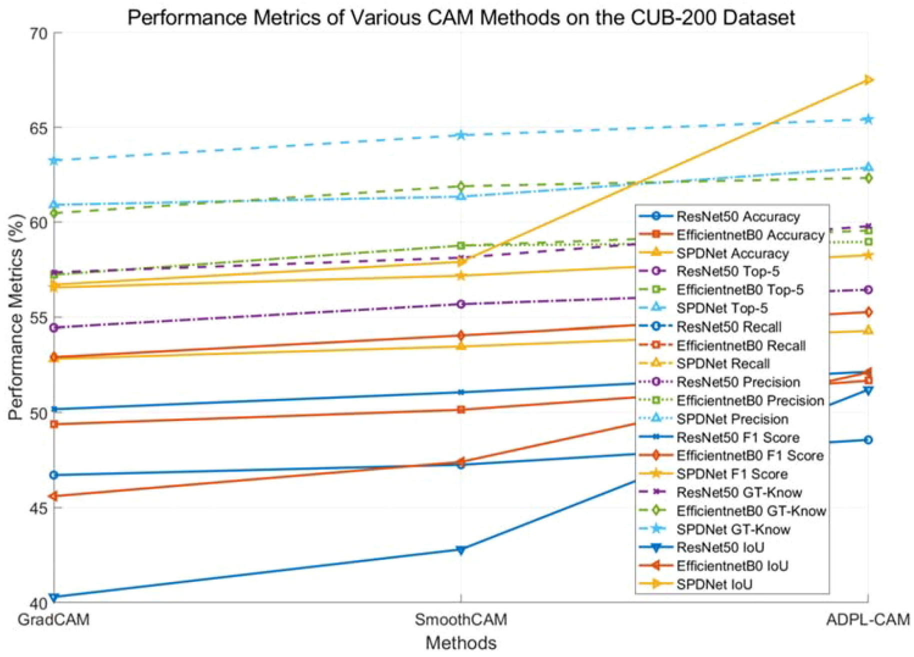


FIGURE 6
Displayed results of different visual models combined with different CAM methods on the CUB-200 dataset.

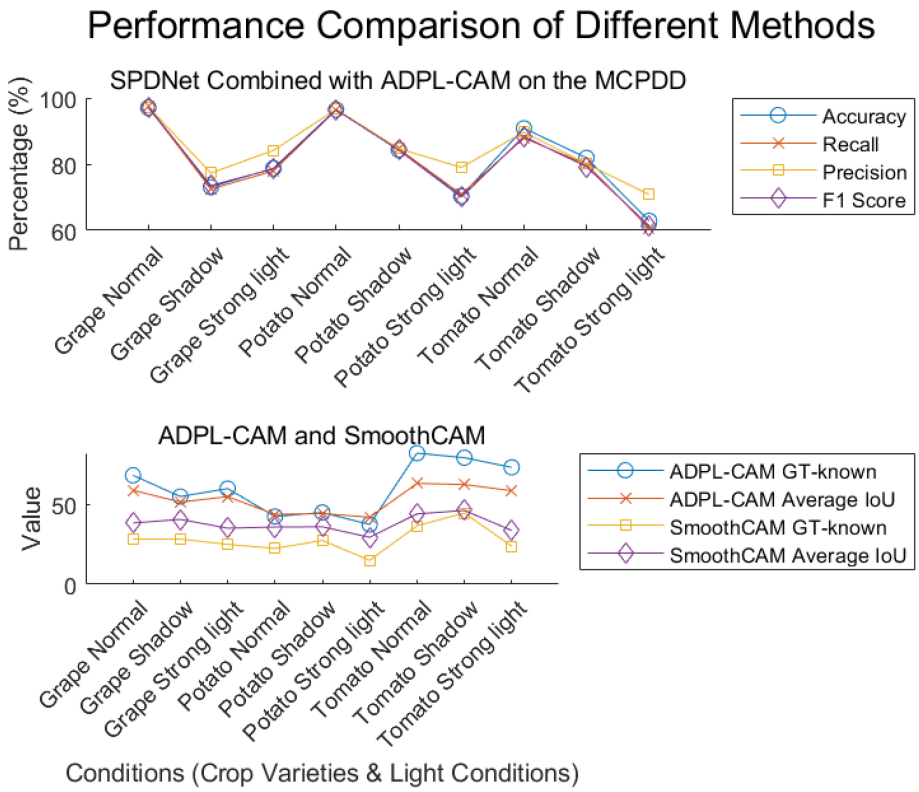


FIGURE 7
Visualization of experimental data. Displayed results of the combination of SPDNet and ADPL-CAM methods on MCPDD.

TABLE 1 Results of various CAM methods on the CUB-200 dataset (units: %).

Method	CNN	Top-1	Top-5	Recall	Precision	F1 score	GT-know	Average IoU
GradCAM	ResNet50	46.71	54.44	46.71	54.44	50.17	57.35	40.3
SmoothCAM	ResNet50	47.25	55.67	47.25	55.67	51.06	58.12	42.8
ADPL-CAM	ResNet50	**48.56**	**56.43**	**48.56**	**56.43**	**52.14**	**59.78**	**51.2**
GradCAM	EfficientnetB0	49.38	57.22	49.38	57.22	52.91	60.47	45.6
SmoothCAM	EfficientnetB0	50.14	58.76	50.14	58.76	54.05	61.89	47.4
ADPL-CAM	EfficientnetB0	**51.67**	**59.55**	**51.67**	**58.95**	**55.25**	**62.33**	**52.1**
GradCAM	SPDNet	52.82	60.91	52.82	60.91	56.55	63.25	56.7
SmoothCAM	SPDNet	53.47	61.34	53.47	61.34	57.17	64.58	57.9
ADPL-CAM	SPDNet	**54.29**	**62.87**	**54.29**	**62.87**	**58.25**	**65.42**	**67.5**

The bold values in the table indicate the optimal performance of each method on the CUB-200 dataset.

4.3 Qualitative experiments and discussion

We conducted our research using the SPDNet+ADPL-CAM strategy to visualize the effectiveness of our proposed method on two datasets and to compare the generated localization bounding boxes with the actual detection bounding boxes, as shown in Figure 8. Additionally, to provide a comprehensive display of this method's performance, we have published all the localization data from our qualitative experiments on GitHub [Qualitative Experiment Visualization (github.com)].

By integrating the ADPL-CAM Class Activation Mapping method with the SPDNet architecture, a series of visualization results were obtained. These results demonstrate the potential advantages of this combination in feature recognition and target localization. From the visualized class activation maps, it is evident that this combination can accurately identify and locate target areas. This not only confirms the efficacy of SPDNet in capturing key features but also illustrates the capability of the ADPL-CAM method in accurately generating target localization frames (annotation boxes). This rapid target localization approach, based on image-level labels, offers significant advantages in reducing training costs and resource consumption. It also provides directions for further optimization of SPDNet and improvements to the ADPL-CAM algorithm.

However, the visualization results also highlighted some areas for improvement. When dealing with widely distributed and scattered disease features, ADPL-CAM tends to recognize only the most prominent parts, which could lead to failures in detecting multiple smaller features. Additionally, the detection outcomes are influenced by lighting conditions, which may affect the accuracy of the localizations.

5 Conclusion

This paper addresses the challenges of multiscale and random distribution of plant disease characteristics by proposing a weakly supervised localization model based on Siamese neural networks. This model is equipped with a proprietary ADPL-CAM algorithm, which accurately identifies and locates areas affected by plant diseases. In early-stage disease detection tasks, the model can timely and accurately identify

TABLE 2 Results of SPDNet combined with ADPL-CAM on the MCPDD.

Crop varieties	Light conditions	Accuracy	Recall	Precision	F1 score	GT-known (ADPL-CAM)	Average IoU (ADPL-CAM)	GT-known (SmoothCAM)	Average IoU (SmoothCAM)
Grape	Normal	97.09	97.34	97.30	97.30	68.33	58.90	28.33	38.39
	Shadow	73.06	72.47	77.31	73.53	55.00	51.56	28.33	40.68
	Strong light	78.77	77.96	84.16	78.57	60.00	54.90	25.00	35.16
Potato	Normal	96.50	96.35	96.55	96.40	42.50	43.79	22.50	35.88
	Shadow	84.33	84.62	84.69	84.24	45.00	44.32	27.50	36.03
	Strong light	70.33	70.88	79.00	70.04	37.50	42.04	15.00	29.42
Tomato	Normal	90.84	87.93	89.67	88.32	82.22	63.25	36.67	44.13
	Shadow	81.91	79.93	80.08	79.23	79.34	62.54	44.69	46.30
	Strong light	62.88	60.74	70.82	61.36	73.33	58.67	23.90	33.74

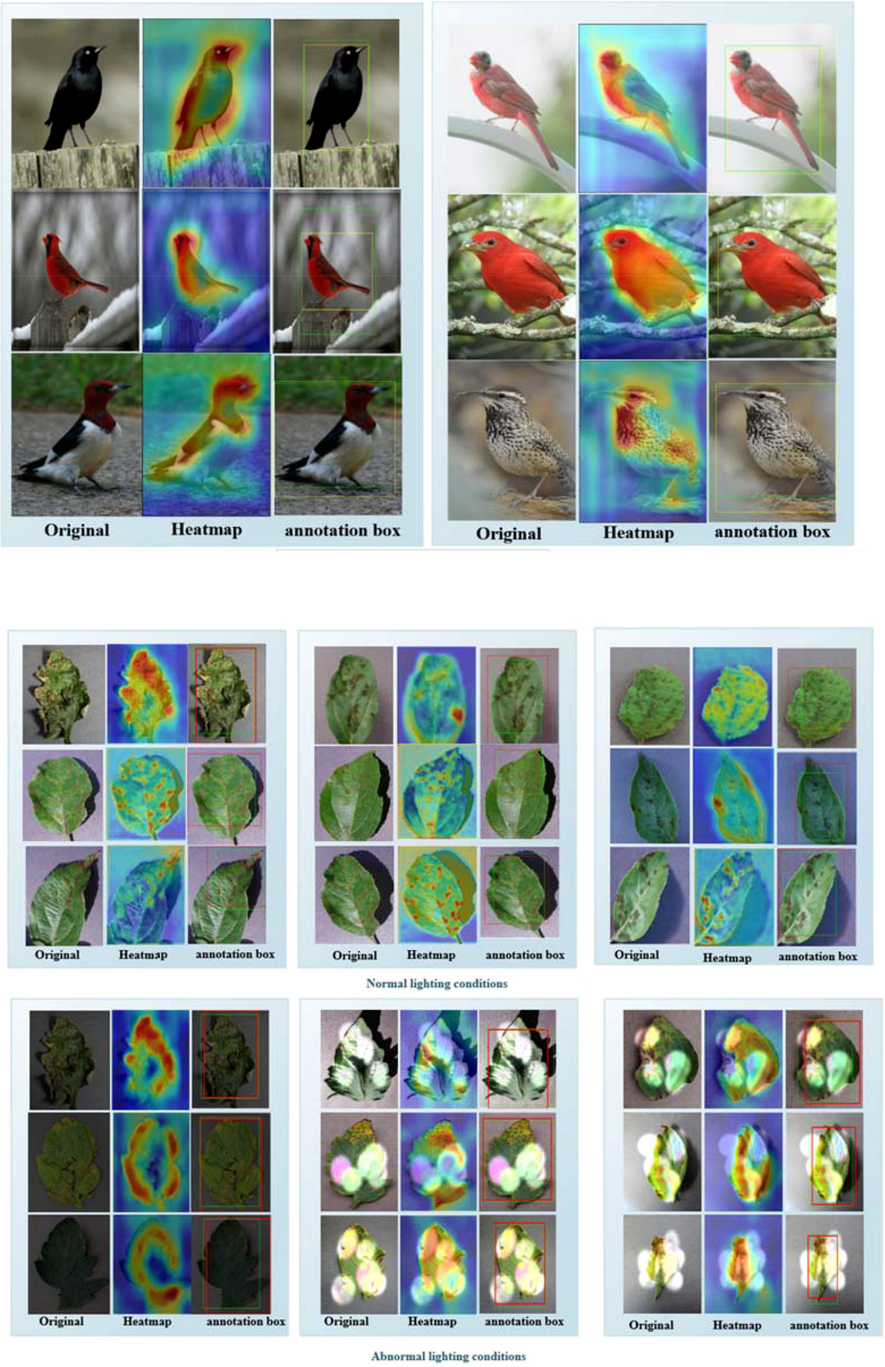


FIGURE 8
(A) Part of the experimental results on the CUB-200 dataset. The first column contains the original images, the second column shows the ADPL-CAM class activation maps, and the third column displays the localization maps. Yellow boxes represent the target boxes, while green boxes indicate the generated boxes. **(B)** Partial experimental results of the MCPDD dataset are described, with the first column being the original plant disease map, the second column being the ADPL-CAM class activation map, and the third column being the localization map. The green box represents the target box, and the red box represents the generated box.

and locate diseased crop leaves. Moreover, the model also demonstrates good performance in other feature recognition tasks. Delving deeply into the ADPL-CAM technology enhances our model's capability to pinpoint plant diseases with remarkable precision. This empowers farmers with prompt and reliable diagnostic insights, mitigating the misuse of pesticides and avoiding the repercussions of misdiagnoses on crop yields. Enhancing the model's resilience to fluctuations in light and extreme conditions is essential, guaranteeing consistent performance amidst the diverse and unpredictable agricultural landscapes. Integrated into an intelligent decision support framework, our model becomes a pivotal tool for farmers, aiding in the rapid identification of plant afflictions and offering strategic management advice, thereby diminishing labor demands and elevating agricultural productivity. Technicians benefit from the model's swift disease detection, enabling them to tailor more effective control strategies, thus bolstering the efficacy of their interventions. For researchers, the model serves as a vigilant sentinel for disease surveillance and a robust data repository, laying down a solid scientific foundation for disease management and the cultivation of new crop varieties.

Future research will focus on the following areas:

1. Exploring ADPL-CAM mechanisms and mapping strategies: We plan to further investigate the mechanisms behind ADPL-CAM and its performance enhancement in various CNN architectures. This includes analyzing how it effectively integrates multiscale features and handles spatial transformations to optimize methods or develop more efficient CAM variants. Considering the limitations of ADPL-CAM in handling complex features, exploring new activation mapping techniques could be beneficial. For instance, introducing an attention-based Class Activation Mapping might help the model focus better on multiple key areas of the target.

2. Enhancing model robustness: Although ADPL-CAM maintains good performance in complex disease scenarios, enhancing the model's adaptability to extreme variations (such as very small or concealed disease features) is also crucial. This might be achieved by integrating more fine-grained feature extraction mechanisms or using deeper learning strategies. The impact of lighting conditions on image recognition is a complex but critical issue. Model robustness to lighting variations could be improved through data augmentation (e.g., introducing a variety of lighting conditions during training) or by incorporating lighting-invariant features.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://AutoGo-Lab/SPDNet: Qualitative Experiment Visualization.

Author contributions

JC: Conceptualization, Methodology, Data curation, Formal analysis, Validation, Writing – original draft. JG: Conceptualization, Funding acquisition, Methodology, Project administration, Writing – review & editing. HZ: Investigation, Validation, Writing – original draft. ZL: Conceptualization, Investigation, Visualization, Writing – review & editing. SW: Methodology, Project administration, Supervision, Validation, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work is supported in part by the Dongguan Science and Technology of Social Development Program (20221800905102) and the Project of Education Department of Guangdong Province (2022ZDZX4053, pdjh2022b0512).

Acknowledgments

During the preparation of this work, the authors used ChatGPT in order to improve language and readability. After using this tool/service, the authors reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Conflict of interest

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

Publisher's note

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

References

Atila, Ü., Uçar, M., Akyol, K., and Öztürk, Ş. (2020). Efficient deep learning techniques for the classification of plant leaf diseases: application of transfer learning. *J. Plant Dis. Prot.* 127, 603–613. doi: 10.1016/j.ecoinf.2020.101182

Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., and Torr, P. H. (2016). "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*. Computer Vision–ECCV 2016 Workshops. (Amsterdam, The Netherlands: Springer International Publishing). 850–865.

Bradley, D., and Roth, G. (2007). Adaptive thresholding using the integral image. *J. Graphics Tools* 12, 13–21. doi: 10.1080/2151237X.2007.10129236

Carbonneau, M. A., Cheplygina, V., Granger, E., and Gagnon, G. (2018). Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition* 77, 329–353. doi: 10.1016/j.patcog.2017.10.009

Chen, X., Wu, S., and Zhang, D. (2018). Deep learning with transfer learning for plant disease recognition. *Commun. Comput. Inf. Sci.* 895, 245–257. doi: 10.1109/ICoDT252288.2021.9441512

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

Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17, 2022. doi: 10.3390/s17092022

Hadsell, R., Chopra, S., and LeCun, Y. (2006). "Dimensionality reduction by learning an invariant mapping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Los Alamitos, CA, IEEE Computer Society). 1735–1742. doi: 10.1109/CVPR.2006.100

He, K., Zhang, X., Ren, S., and Sun, J. (2015). "Delving deep into rectifiers: surpassing human-level performance on imageNet classification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (Santiago, Chile: IEEE Computer Society). 1026–1034.

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 (CVPR)*. (Las Vegas, NV: IEEE Computer Society). 770–778.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.* 9, 1735–1780. doi: 10.1162/neco.1997.9.8.1735

Hu, J., Shen, L., and Sun, G. (2016). "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Salt Lake City, UT: IEEE Computer Society). 7132–7141.

Jaderberg, M., Simonyan, K., Zisserman, A., and Kavukcuoglu, K. (2015). Spatial transformer networks. *Adv. Neural Inf. Process. Syst. (NIPS)* 28, 2017–2025. doi: 10.48550/arxiv.1506.02025

Jiang, Z., Zhang, H., Wang, L., Li, Z., and Lv, Q. (2020). Layer-CAM: exploring hierarchical class activation maps for localization. *IEEE Trans. Image Process.* 29, 2121–2133. doi: 10.1109/TIP.2021.3089943

Kumar, A., Lee,, and Y., S. (2020). K-nearest neighbors and a kernel density estimator for classification of plant disease images. *Comput. Electron. Agric.* 170, 105202. doi: 10.1109/ICACCS.2019.8728325

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

Lee, J., Kim, H. S., and Lee, S. H. (2018). Data augmentation for plant disease detection using generative adversarial networks. *Plant Pathol. J.* 34, 545–552. doi: 10.5423/PPJ.OA.02.2018.01

Li, Y., Guo, J., Qiu, H., Chen, F., and Zhang, J. (2023). Denoising Diffusion Probabilistic Models and Transfer Learning for citrus disease diagnosis. *Front. Plant Sci.* 14, 1267810. doi: 10.3389/fpls.2023.1267810

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Honolulu, HI: IEEE Computer Society). 2117–2125.

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 (ECCV)*. (Amsterdam, Netherlands: Springer). 21–37. doi: 10.1007/978-3-319-46448-0_2

Liu, K., and Zhang, X. (2022). PiTLiD: identification of plant disease from leaf images based on convolutional neural network. *IEEE/ACM Trans. Comput. Biol. Bioinf.* 20, 1278–1288. doi: 10.1109/TCBB.2022.3195291

Mahlein, A. K. (2016). Plant disease detection by imaging sensors – Parallels and specific demands for precision agriculture and plant phenotyping. *Plant Dis.* 100, 241–251. doi: 10.1094/PDIS-03-15-0340-FE

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7, 1419. doi: 10.3389/fpls.2016.01419

Mueller, J., and Thyagarajan, A. (2016). "Siamese recurrent architectures for learning sentence similarity," in *Proceedings of the AAAI Conference on Artificial Intelligence*. (V, AAAI Press). 2786–2792.

Poornima, S. T., and Pushpalatha, M. P. (2021). PlantXViT: A model for plant disease identification using convolution neural network and vision transformer. *Plant Methods* 17, 1–16. doi: 10.1186/s13007-021-00738-0

Qualitative Experiment Visualization (github.com). Available online at: https://AutoGo-Lab/SPDNet. (Accessed September 18, 2024).

Rumpf, T., Mahlein, A. K., Steiner, U., Oerke, E. C., Dehne, H. W., and Plümer, L. (2010). Early detection and classification of plant diseases with Support Vector Machines based on hyperspectral reflectance. *Comput. Electron. Agric.* 74, 91–99. doi: 10.1016/j.compag.2010.06.009

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. C. (2018). "MobileNetV2: inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Salt Lake City, UT: IEEE Computer Society). 4510–4520.

Sankaran, S., Mishra, A., Ehsani, R., and Davis, C. (2010). A review of advanced techniques for detecting plant diseases. *Comput. Electron. Agric.* 72, 1–13. doi: 10.1016/j.compag.2010.02.007

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). "FaceNet: A unified embedding for face recognition and clustering," in *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (Boston, MA: IEEE Computer Society). 815–823.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). "Grad-CAM: visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (Boston, MA: IEEE Computer Society). 618–626.

Singh, A., Ganapathysubramanian, B., and Singh, A. K. (2018). Machine learning for high-throughput stress phenotyping in plants. *Trends Plant Sci.* 21, 110–124. doi: 10.1016/j.tplants.2015.10.015

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., and Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Comput. Intell. Neurosci.* 2016, 1–8. doi: 10.1155/2016/3289801

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958. doi: 10.5555/2627435.2670313

Sumaya, M., and Uddin, M. S. (2021). A review on deep learning approaches for 3D data representations in plant phenotyping. *Plant Methods* 17, 28. doi: 10.1109/ACCESS.2020.2982196

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). "DeepFace: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1701–1708.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inf. Process. Syst. (NIPS)* 30, 5998–6008. doi: 10.48550/arXiv.1706.03762

Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., and Summers, R. M. (2017). "ChestX-ray8: hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Honolulu, HI: IEEE Computer Society). 3462–3471.

Wang, H., Wang, Z., Du, M., Yang, F., Zhang, Z., Ding, S., et al. (2020). "Score-CAM: score-weighted visual explanations for convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. (Seattle, WA: IEEE Computer Society). 24–25.

Zeiler, M. D., and Fergus, R. (2014). "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision (ECCV)*. (Zurich, Switzerland: Springer) 818–833.

Zhang, X., Zhao, J., and LeCun, Y. (2017). Character-level convolutional networks for text classification. *Adv. Neural Inf. Process. Syst. (NIPS)* 30, 649–657. doi: 10.48550/arXiv.1509.01626

Zhou, Z. H., and Feng, J. (2017). "Deep forest: towards an alternative to deep neural networks," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*. (Melbourne, Australia: IJCAI) 3553–3559.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). "Learning deep features for discriminative localization," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Las Vegas, NV: IEEE Computer Society) 2921–2929. doi: 10.1109/CVPR.2016.319

Zj, L., Wang, Y., and Zhao, X. (2019). Multitask learning for plant diseases and pests recognition based on improved VGG16 model. *Trans. Mach. Learn. Artif. Intell.* 7, 21–34. doi: 10.14738/tmlai.72.6236

TYPE Original Research
PUBLISHED 22 October 2024
DOI 10.3389/fpls.2024.1423238

LT-DeepLab: an improved DeepLabV3+ cross-scale segmentation algorithm for Zanthoxylum bungeanum Maxim leaf-trunk diseases in real-world environments

Tao Yang[1], Jingjing Wei[2], Yongjun Xiao[1], Shuyang Wang[1], Jingxuan Tan[1], Yupeng Niu[1], Xuliang Duan[1,3], Fei Pan[1,3*] and Haibo Pu[1,3*]

[1]College of Information Engineering, Sichuan Agricultural University, Ya'an, China, [2]School of Physics and Optoelectronic Engineering, Nanjing University of Information Science and Technology, Jiangsu, China, [3]Ya'an Digital Agricultural Engineering Technology Research Center, Sichuan Agricultural University, Ya'an, China

Introduction: Zanthoxylum bungeanum Maxim is an economically significant crop in Asia, but large-scale cultivation is often threatened by frequent diseases, leading to significant yield declines. Deep learning-based methods for crop disease recognition have emerged as a vital research area in agriculture.

Methods: This paper presents a novel model, LT-DeepLab, for the semantic segmentation of leaf spot (folium macula), rust, frost damage (gelu damnum), and diseased leaves and trunks in complex field environments. The proposed model enhances DeepLabV3+ with an innovative Fission Depth Separable with CRCC Atrous Spatial Pyramid Pooling module, which reduces the structural parameters of Atrous Spatial Pyramid Pooling module and improves cross-scale extraction capability. Incorporating Criss-Cross Attention with the Convolutional Block Attention Module provides a complementary boost to channel feature extraction. Additionally, deformable convolution enhances low-dimensional features, and a Fully Convolutional Network auxiliary header is integrated to optimize the network and enhance model accuracy without increasing parameter count.

Results: LT-DeepLab improves the mean Intersection over Union (mIoU) by 3.59%, the mean Pixel Accuracy (mPA) by 2.16%, and the Overall Accuracy (OA) by 0.94% compared to the baseline DeepLabV3+. It also reduces computational demands by 11.11% and decreases the parameter count by 16.82%.

OPEN ACCESS

EDITED BY
Chunlei Xia,
Chinese Academy of Sciences (CAS), China

REVIEWED BY
Anqing Zhang,
George Washington University, United States
Sunil GC,
North Dakota State University, United States

*CORRESPONDENCE
Fei Pan
fei.pan@sicau.edu.cn
Haibo Pu
puhb@sicau.edu.cn

RECEIVED 25 April 2024
ACCEPTED 30 September 2024
PUBLISHED 22 October 2024

CITATION
Yang T, Wei J, Xiao Y, Wang S, Tan J, Niu Y, Duan X, Pan F and Pu H (2024) LT-DeepLab: an improved DeepLabV3+ cross-scale segmentation algorithm for Zanthoxylum bungeanum Maxim leaf-trunk diseases in real-world environments.
Front. Plant Sci. 15:1423238.
doi: 10.3389/fpls.2024.1423238

Discussion: These results indicate that LT-DeepLab demonstrates excellent disease segmentation capabilities in complex field environments while maintaining high computational efficiency, offering a promising solution for improving crop disease management efficiency.

KEYWORDS

Zanthoxylum bungeanum Maxim, cross-scale, real-world environments, disease segmentation, small target, deep learning, attention mechanism

1 Introduction

Zanthoxylum bungeanum Maxim, a woody plant in the Rutaceae family, is widely distributed across Asia. It serves as a significant economic resource, offering valuable seasoning, spice, and woody oilseed derivatives from its branches, leaves, and fruits. These parts also possess high nutritional and medicinal values (Lu et al., 2022). Under large-scale cultivation, the plant frequently suffers from diseases, particularly in its branches and leaves, significantly impacting yield. Early disease detection is crucial for preventing substantial economic losses in agriculture (Savary et al., 2019). Consequently, rapid and accurate monitoring and analysis of leaf and trunk diseases in Zanthoxylum bungeanum Maxim are essential. Traditional disease identification methods, which predominantly rely on subjective manual visual observation, are labor-intensive, slow, and prone to misclassification (Cruz et al., 2019).

In terms of traditional segmentation techniques, threshold-based methods are prevalent. Gao and Lin (2019) enhanced and extracted leaf veins to segment medicinal plant leaves using direct processing of RGB images and OTSU methods. Barbedo (2016) developed a semi-automatic algorithm for segmenting plant leaf disease symptoms by manipulating the histograms of the H-channel in HSV and the a-channel in Lab color space. Clustering-based approaches are also utilized; for instance, Shedthi et al. (2023) designed a plant disease recognition system using hybrid clustering algorithms to improve upon the local optimization limitations of the k-means algorithm. Javidan et al. (2023) employed new image processing algorithms and multi-class support vector machines for diagnosing and classifying grapevine leaf diseases, achieving up to 98.97% accuracy with PCA and GLCM feature selection. Additionally, there are region-based methods: Ma et al. (2017) proposed a segmentation method for vegetable leaf lesions using color information and region-growing techniques. They composed a comprehensive color feature using the red index, the H component in HSV color space, and the b component in Lab color space. Based on this feature, an interactive region-growing method was used to segment leaf lesions against a complex background. Li et al. (2018) developed a single-leaf segmentation method for indoor ornamental plant leaves using over-segmentation with small planes and region growing with small planes in a dense plant point cloud, achieving an average precision and recall rate exceeding 90%. These methods, while less computationally demanding and straightforward, often lack robustness in complex backgrounds due to subtle gray-scale variations and small diseased spot sizes on leaves.

With the continuous advancements in computer vision, high-performance models have been increasingly utilized for image classification, detection, and recognition tasks (Attri et al., 2023). There are currently three principal approaches for analyzing plant diseases using deep learning: image-based classification, bounding box-based object detection, and semantic segmentation based on pixel classification. Nahiduzzaman et al. (2023) developed a lightweight deep separable CNN model, PDS-CNN, achieving accuracies of 95.05% in triple classification and 96.06% in binary classification with a compact model size of 6.3M. Pal and Kumar (2023) combined traditional INC-VGGN and Kohonen-based networks for plant disease detection and severity classification. Thai et al. (2023) introduced FormerLeaf for cassava leaf disease detection, employing the Least Important Attention Pruning (LeIAP) algorithm to enhance Transformer models by reducing model size by 28% and improving accuracy by approximately 3%. Additionally, they utilized the sparse matrix multiplication method (SPMM) to decrease the model's complexity, reducing training time by 10%. Liu et al. (2024) proposed Fusion Transformer YOLO, a real-time and lightweight detection model that integrates VoVNet into the backbone to enhance accuracy and incorporates an improved dual-stream PAN+FPN structure in the neck, achieving an average model accuracy of 90.67%. Jodas et al. (2021) merged deep residual blocks with UNet for semantic segmentation, achieving an IoU of 81.47% by refining the segmentation region to exclude irrelevant binary areas. Zhang et al. (2023) improved the sensory field in a grapevine leaf disease segmentation model by inverting the residual convolution and replacing the downsampling operation with reversible attention, increasing IoU performance by 4.04% over the baseline model. Compared to traditional methods, semantic segmentation offers more practical and complex functionalities, making it highly suitable for precision agriculture applications (Deng et al., 2023).

Unlike previous studies, our task requires cross-scale segmentation due to varying sizes of diseased trunks and frost-damaged parts, which differ from the smaller diseased leaves and spots. The ASPP structure of Deeplabv3+ is particularly apt for cross-scale feature extraction due to its varied receptive fields, making it an ideal baseline for our study on Zanthoxylum bungeanum Maxim trunk and leaf disease segmentation. In our experiments, we identified two main challenges: (1) significant loss of target edge information in complex backgrounds, leading to poor segmentation under varied environmental conditions and blurred target boundaries, and (2) the difficulty in detecting and segmenting small disease spots on leaves due to their irregular size and presence.

To address the issue of complex backgrounds, Wang et al. (2021) fused DeepLabV3 and UNet in a two-stage model for cucumber leaf lesion segmentation, initially segmenting leaves in complex backgrounds with DeepLabV3 followed by lesion segmentation with UNet. Mzoughi and Yahiaoui (2023) segmented diseases based on local disease signature features, reducing the impact of common backgrounds. To mitigate computational costs, this paper designs a lightweight dual-attention mechanism that concurrently extracts features from both channel and spatial dimensions, focusing the model on target regions while disregarding background noise.

To tackle the problem of overlooking small leaf spots, Qi and Jia (2023) enhanced segmentation accuracy for small infrared targets by modifying the expansion rate of the ASPP module and introducing a position enhancement module. Deng et al. (2023) developed a cross-layer attention fusion mechanism to differentiate tiny spots from healthy areas. This paper enhanced the ASPP module by altering its data flow, adding deformable convolution, and incorporating our proposed CRCC module to better detect small target spots. Additionally, standard convolution is replaced with depth-separable convolution to reduce parameter count while improving accuracy. Furthermore, deformable convolution is applied to shallow extracted features before their integration with deep features to more effectively transfer shallow information.

In this paper, a cross-scale disease segmentation network is proposed, LT-DeepLab, for Zanthoxylum bungeanum Maxim trunks and leaves. The contributions of this study are summarized as follows: (1) A dual-attention module CRCC structure is designed, combining spatial and channel attention mechanisms to enhance segmentation in complex backgrounds. (2) An improved ASPP module (FDCASPP) is proposed, incorporating an enhanced attention mechanism with variability convolution to boost cross-scale feature extraction and using lightweight deep separable convolution to minimize redundant information. (3) The model employs a deep supervision technique that does not increase parametric quantities and incorporates an auxiliary loss during training to enhance accuracy. (4) This paper innovatively applies semantic segmentation techniques to Zanthoxylum bungeanum Maxim disease segmentation, producing a scientific dataset of Zanthoxylum bungeanum Maxim leaf and trunk disease and facilitating cross-scale segmentation of leaf and trunk diseases, thereby bridging the research gap in this area.

2 Materials and methods

2.1 Data collection and processing

2.1.1 Data collection

This study examines the segmentation of diseases on the leaves and trunks of Zanthoxylum bungeanum Maxim trees within complex environments. The image data were collected from a Zanthoxylum bungeanum Maxim plantation located in Dongba Town, Nanbu County, Sichuan Province, China. To accommodate diverse lighting conditions in natural settings, the dataset was compiled at various times in July, specifically in the morning (8:00-10:00), at noon (12:00-14:00), and in the afternoon (15:00-17:00), with additional images captured post-rainfall. The categories of images include leaf spot, rust, and frost damage. Representative examples of these images are presented in Figure 1.

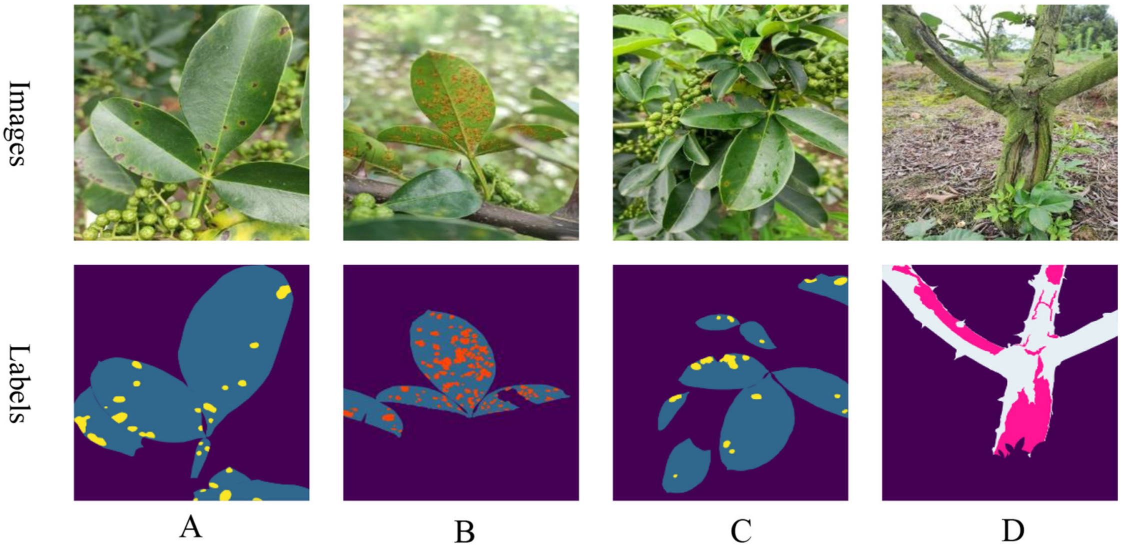


FIGURE 1
Samples of various diseases: **(A)** leaf spot; **(B)** rust; **(C)** postrainy leaf spot; **(D)** frost damage.

2.1.2 Data processing

A total of 1,200 raw images were captured with an initial resolution of 3472x4624. Leaf and disease spots, as well as trunk and frost damage, were annotated using Labelme software under expert guidance to create mask maps. The dataset was then split into a training set of 960 images and a test set of 240 images, adhering to an 8:2 ratio. The annotated images are depicted as labels in Figure 1.Since the initial training set comprised only 960 images, this paper applies data augmentation to boost the model's robustness and generalization capabilities. To maintain a reasonable training speed, the augmentation process involved randomly scaling the length of the images to a range between 2048 and 512 pixels, then cropping them to a size of 512x512 pixels, and finally applying random flips. Additionally, transformations in terms of brightness, contrast, and saturation were used to further improve the model's performance. These data augmentation techniques were applied consistently across all experiments to ensure uniformity among the different models.

2.2 Improved methods

Based on DeepLabv3+, this paper proposes a segmentation network named LT-DeepLab designed for cross-dimensional segmentation of Zanthoxylum bungeanum Maxim trunks, leaves, and lesions. The network primarily consists of deformable convolutions, a fission feature pyramid with depth-separable convolutions, and an improved CRCC dual attention module. The CRCC module combines the Criss-Cross module with the Convolutional Block Attention Module, allowing for feature complementation in both spatial and channel dimensions, and is used in both the backbone and FDCASPP modules. Furthermore, the FDCASPP module incorporates deformable convolutions and depth-separable convolutions, reducing the parameter count while maintaining or even improving model accuracy.

2.2.1 DeepLabv3+ network structure

DeepLabv3+ is a prominent semantic segmentation architecture distinguished by its Atrous Spatial Pyramid Pooling (ASPP) module, which employs dilated convolution to capture contextual information across various scales (Chen et al., 2018). This is achieved by applying differing dilation rates to feature maps processed by deep neural networks, which are then combined with low-level features to produce the prediction map. However, the original model had a high parameter count and did not perform well in segmentation for this specific task, prompting us to make several improvements.

2.2.2 LT-DeepLab structure

In real environments, the segmentation of leaf and trunk diseases is complicated by various factors such as light, weather, shading, and complex backgrounds, particularly when imaging leaf spots and frost-damaged trunk portions. This study addresses both the larger-sized trunk and frost-damaged parts as well as the smaller leaves and even smaller diseased spots, with the inherent data imbalance increasing segmentation difficulty. Although the traditional DeepLabv3+ network, with its ASPP module capable of multi-scale feature extraction, achieves satisfactory segmentation results on leaves and trunks against a single background, it struggles with more complex backgrounds. The performance deteriorates further due to the cross-pixel feature extraction of the expansion convolution within the ASPP module, often failing to adequately capture features of leaves and smaller spots, which is critical for segmentation tasks involving small targets (Zhu et al., 2023). To address these challenges, this study introduces an enhanced version of DeepLabv3+, LT-DeepLab, as depicted in Figure 2. This model

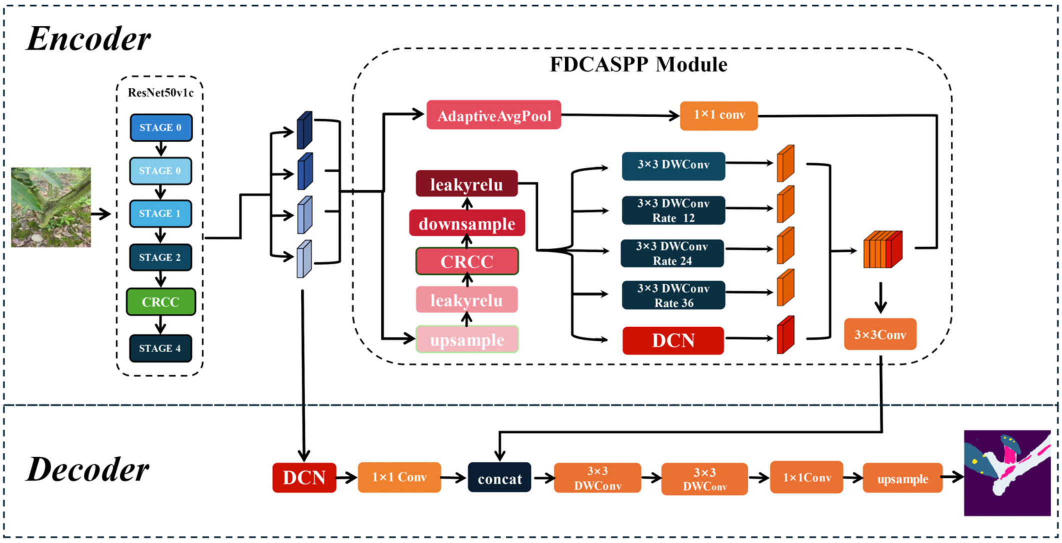


FIGURE 2
LT-DeepLab network structure.

integrates an improved Criss-Cross attention mechanism to boost the feature extraction capability of the backbone. In the decoder, features from the backbone are fused with outputs from the ASPP module, incorporating deformable convolution to better preserve features of small targets like leaves and disease spots. Furthermore, this paper proposes a new encoder, the Separation Fission Depth Separable with CRCC Atrous Spatial Pyramid Pooling, called FDCASPP module. This encoder replaces standard convolution with depth-separable convolution, retaining the multi-scale feature extraction of ASPP while integrating an enhanced CRCC module and deformable convolution, thus addressing the insufficient feature extraction capability for small targets.

2.2.3 CRCC module structure

In real-world scenarios, the segmentation of Zanthoxylum bungeanum Maxim leaves and trunks is challenged by frequent occlusions and the phenotypic similarity between healthy and diseased leaves. To improve the model's focus on relevant features and enhance information filtration from complex backgrounds, this paper introduces and refines the Criss-Cross attention module (Huang et al., 2019). The enhanced CCNet efficiently captures contextual information from the surrounding pixels via cross-path operations. This mechanism enables each pixel to ascertain the remote dependencies of all other pixels through a cyclic operation, thereby improving segmentation accuracy. The original Criss-Cross module is computed as, given a local feature map $H \in \mathbb{R}^{C\times W\times H}$, the feature maps K, Q, and V are first generated for the leaf and trunk lesion feature maps after three 1 × 1 convolutions, where $\{Q, K\} \in \mathbb{R}^{C'\times W\times H}$, and C' is the number of channels less than C. After that, this paper generates the feature maps Q and K by **Affinity** operation to generate the attention map $A \in \mathbb{R}^{(H+W-1)\times W\times H}$, and for the position u of the feature map Q in the spatial dimension, which can obtain $Q_u \in \mathbb{R}^{C'}$, and for the same row or column of the same position u in K, it can obtain $\mathbf{\Omega}_u \in \mathbb{R}^{(H+W-1)\times C'}$. $\mathbf{\Omega}_{i,u} \in \mathbb{R}^{C'}$ is the ith element of $\mathbf{\Omega}_u$. Define the **Affinity** operation as follows:

$$d_{i,u} = Q_u \Omega_{i,u} \tag{1}$$

Where $d_{i,u} \in D$ is the degree of correlation between feature Q_u and $\mathbf{\Omega}_{i,u}$, $i = [1, \ldots, |\mathbf{\Omega}_u|]$, $D \in \mathbb{R}^{(H+W-1)\times W\times H}$. Then, a softmax layer on D over the channel dimension is applied to calculate the attention map A.

Correspondingly, for the previously generated feature map $V \in \mathbb{R}^{C\times W\times H}$ and each position u in the spatial dimension, $V_u \in \mathbb{R}^{C}$ and a set $\mathbf{\Phi}_u \in \mathbb{R}^{(H+W-1)\times C}$ are obtained, where the set $\mathbf{\Phi}_u$ is the set of feature vectors in V that are in the same row or column as position u. **Aggregation** is defined as:

$$H_u^{'} = \sum_{i\in|\Phi_u|} A_{i,u}\Phi_{i,u} + H_u \tag{2}$$

where $H_u^{'}$ is the output feature maps $H' \in \mathbb{R}^{C\times W\times H}$ at position u. $A_{i,u}$ is the scalar value at a for channel i and position u.

However, while the Criss-Cross Attention module effectively contextualizes features spatially, it does not adequately connect spatial information (Huang et al., 2019). To address this limitation, this paper integrated it with the Convolutional Block Attention Module (CBAM) (Woo et al., 2018), creating a dual attention mechanism named CRCC, as illustrated in Figure 3. In this mechanism, the feature maps K, Q, and V from H are fused before being processed by the CBAM module, which then weights these features to ensure a cohesive channel connection. The integration process is detailed as follows:

$$H_c = CBAM(concat(K, Q, V)) \tag{3}$$

Finally, the output of the CRCC module is combined with that of the Criss-Cross module. To capture the global connections in a cyclic manner, a two-dimensional convolution is applied to integrate the features and compress their dimensionality. This process preserves the original spatial context provided by the Criss-Cross module while incorporating the spatial and channel-weighted features from the CBAM. The final output of the CRCC module after one iteration is described below:

$$H_{out} = Conv2d(H_c + H_u^{'}) \tag{4}$$

Additionally, this paper has integrated the CRCC module into the enhanced Fission Depth Separable with CRCC Atrous Spatial

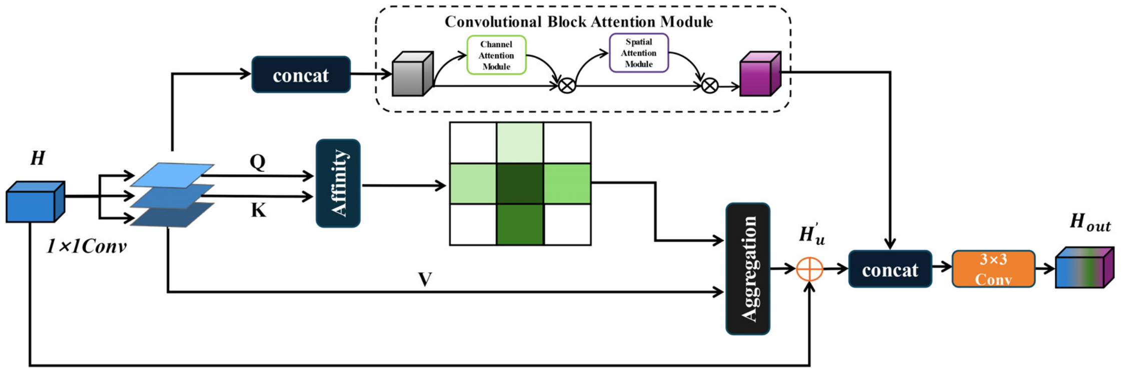


FIGURE 3
CRCC module structure.

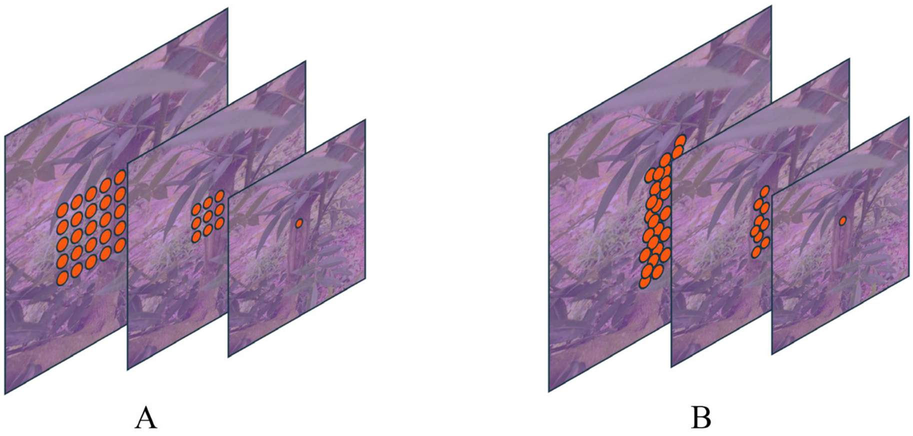


FIGURE 4
Comparison of conventional and deformable convolution receptive fields: Panel **(A)** displays a schematic diagram of the receptive field for a standard 3x3 convolution, while Panel **(B)** illustrates the receptive field of deformable convolution.

Pyramid Pooling (FDCASPP) structure to further optimize our segmentation network.

2.2.4 Deformable convolution structures

Conventional convolution operates with a fixed kernel shape, which may not adequately address the irregular shapes of leaf and trunk lesions. To overcome this limitation, this paper introduces deformable convolution, which modifies the convolution process to adapt to these irregularities before feature fusion. In standard 2D convolution, feature maps for leaf and trunk lesions are initially sampled using a regular grid network R. The sampled values are then multiplied by their corresponding weights w, and subsequently summed. The output at each position P_0 on the feature maps y, can be described by the following equation (Dai et al., 2017):

$$y(P_0) = \sum_{P_n \in R} W(P_n) \cdot X(P_0 + P_n) \tag{5}$$

where P_n represents each position on the convolution kernel. As illustrated in Figure 4, deformable convolution modifies standard convolution by introducing an offset to the receptive field. Consequently, Equation 5 transforms into Equation 6. This offset is learnable, allowing it to adapt closely to the actual contours of the object. Through ablation studies, deformable convolution has demonstrated enhanced segmentation capabilities, significantly improving the model's performance.

$$y(P_0) = \sum_{P_n \in R} W(P_n) \cdot X(P_0 + P_n + \Delta P_n) \tag{6}$$

2.2.5 FDCASPP module structure

The traditional Atrous Spatial Pyramid Pooling module utilizes specific expansion rates to obtain different receptive fields for multi-scale feature extraction. However, due to the integration of pooling and convolution with strides, there is significant loss of boundary information in the segmented targets. Additionally, the extensive use of expansive convolutions through a deep convolutional neural network with a high number of channels results in a large parameter count. To address these issues, this paper proposes a Fission Depth Separable with CRCC Atrous Spatial Pyramid Pooling (FDCASPP). This module divides the feature maps into two data streams: one stream undergoes global average pooling followed by a 1x1 convolution for global feature statistics, while the other stream reduces the feature map resolution to balance accuracy with computation. As demonstrated by Xu et al. (2015), LeakyReLU outperforms ReLU in scenarios involving small datasets. To enhance model expressiveness, the reduced feature map is activated using LeakyReLU, then processed through the CRCC dual attention mechanism, and subsequently enhanced for activation. It is then integrated into the multidimensional joint feature extraction section, which replaces the standard convolution in the original ASPP module with depth-separable convolution (Sifre and Mallat, 2014) to minimize redundant parameters. This section sets expansion rates at 12, 24, and 36 to accommodate various target sizes. Additionally, deformable convolution is employed to refine the segmentation of targets. Finally, the feature maps from both data streams in the FDCASPP are fused to enhance feature integration.

2.2.6 Auxiliary head loss

To optimize the training process, Zhao et al. (2017) demonstrated in their study on PSPNet that employing auxiliary loss can significantly enhance training effectiveness. They established that setting the weight α of the auxiliary loss to 0.4 is optimal. Notably, the auxiliary head, which processes the feature maps from the backbone network to generate segmentation masks and calculate the auxiliary loss using the cross-entropy loss function, is active only during the training phase. Consequently, it does not add to the computational load or the parameter count during model inference. This paper adopts a similar approach by introducing auxiliary loss generated by the FCN auxiliary head, applying a cross-entropy function, with the weight also set to α=0.4.

2.3 Model training

The hardware configurations for training and testing in this study include an 18 vCPU AMD EPYC 9754 128-Core Processor with 60GB of RAM and an NVIDIA RTX 3090 GPU with 24GB of video memory. The software environment consists of CUDA version 11.1, PyTorch version 1.8.1, and Python version 3.8.10. To mitigate the influence of hyper-parameters on experimental outcomes, this paper standardizes settings across all tests. Specifically, the Stochastic Gradient Descent optimizer was employed with an initial learning rate of 0.01. A polynomial decay strategy, PolyLR, was used to adjust the learning rate during the experiments. The experiments were conducted over 10,000 iterations with a batch size of 4.

2.4 Evaluation metrics

In this study, the effectiveness of disease segmentation on Zanthoxylum bungeum Maxim leaves and trunks is quantitatively assessed using three principal evaluation metrics: mean Intersection over Union (mIoU), mean Pixel Accuracy (mPA), and Overall Accuracy (OA). These metrics are chosen to provide a comprehensive evaluation of the segmentation performance across all tested networks. mIoU, mPA, and OA were formulae are calculated as follows, respectively:

$$mIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{\sum_{j=0}^{k}P_{ij} + \sum_{j=0}^{k}P_{ji} - P_{ii}} \tag{7}$$

$$mPA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{P_{ii}}{\sum_{j=0}^{k}P_{ij}} \tag{8}$$

$$OA = \frac{\sum_{i=0}^{k}P_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k}P_{ij}} \tag{9}$$

where k denotes the number of classes, excluding background P_{ij} denotes the number of pixels that refer to the prediction of category i as category jFor the number of parameters of the model and the amount of computation, it is calculated as:

$$Parameters = C_{in} \times C_{out} \times K \times K \tag{10}$$

$$\text{FLOPs} = C_{out} \times (C_{in} \times K^2) \times W \times H \tag{11}$$

Where C_{in} denotes the number of input channels, C_{out} represents the number of output channels, K refers to the size of the convolutional kernel, and W and H indicate the width and height of the feature map, respectively.

Equation (10) describes how the number of parameters is calculated in each convolutional layer, the smaller the number of parameters is calculated, the lighter the model is and the easier it is to deploy. Equation (11) describes how the amount of computation in each convolutional layer is calculated, the smaller the amount of computation in the model, the smaller the computational burden of the model and the faster the inference. The bias terms in the convolutional layers are not considered in either of the above calculations.

2.5 Normalization of the confusion matrix

In this study, we utilized confusion matrices to evaluate the performance of the baseline model and our proposed model on the test set. To provide a more intuitive understanding of each category's performance, we applied row normalization to the original confusion matrix. This process converts the absolute counts into proportions, indicating the percentage of samples within each category that are predicted to belong to respective categories. Specifically, each element in the normalized confusion matrix, denoted as CM_{norm}, can be expressed as:

$$CM_{norm}[i,j] = \frac{CM[i,j]}{\sum_{k=1}^{N}CM[i,k]} \tag{12}$$

Where $CM[i,j]$ represents the element in the ith row and jth column of the original confusion matrix, indicating the number of samples from the actual category i that are predicted as category j, and $\sum_{k=1}^{N}CM[i,k]$ is the sum of all elements in the ith row, representing the total number of samples in the actual category i.

2.6 Statistical testing method

In this study, we employed a t-test to compare the performance differences between the improved LT-DeepLab model and the baseline model. To facilitate a statistical comparison, we recorded the results of five experiments conducted on the same dataset for both models. We utilized an independent samples t-test to assess whether the mean difference between these two models is statistically significant. Specifically, we calculated the means, variances, and t-statistics for the two samples, and determined the p-value by consulting the t-distribution table. The p-value represents the probability of observing the current t-statistic, or a more extreme value, under the null hypothesis that there is no significant difference between the means of the two groups. If the calculated p-value is less than the predetermined significance level (e.g., 0.05), we reject the null hypothesis, indicating that the mean difference between the two datasets is statistically significant. The calculation method is as follows:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n}X_i \tag{13}$$

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2 \tag{14}$$

$$t = \frac{\left|\overline{X_1} - \overline{X_2}\right|}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}} \tag{15}$$

$$df = \frac{\left(\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}\right)^2}{\frac{\left(\frac{S_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{S_2^2}{n_2}\right)^2}{n_2 - 1}} \tag{16}$$

In this context, $\bar{X}$ represents the sample mean, with $\overline{X_1}$ and $\overline{X_2}$ denoting the means of the two groups. S^2 denotes the sample variance, with S_1^2 and S_2^2 representing the variances of the two groups. n stands for the sample size, with n_1 and n_2 indicating the sample sizes of the two groups. The t-statistic t measures the difference between sample means relative to the variability within the samples, offering a standardized metric of the mean difference. The degrees of freedom df are used to consult the t-distribution table to determine the p-value.

3 Experiments and analysis of results

3.1 Comparison experiments

To establish the superior segmentation capabilities of LT-DeepLab, this paper performes comparative experiments using an identical dataset across various state-of-the-art semantic segmentation networks. Each competing CNN-based model utilized a ResNet50V1c backbone, was subjected to the same data augmentation techniques, and employed a transfer learning approach. All networks were initialized with pre-trained weights from the Cityscapes dataset. The comparative analysis included models such as FCN, CCNet, DANet, PSPNet, Non_Local, UNet, and Segformer. The outcomes of these experiments are detailed in Table 1. As illustrated in Table 1, among the nine networks compared, our network, LT-DeepLab, consistently achieves the best results across all metrics, underscoring its distinct effectiveness for this task. Specifically, LT-DeepLab shows improvements over the baseline network by 3.59% in mIoU, 2.49% in mPA, and 0.63% in OA. Segformer, incorporating the advanced Transformer architecture, ranks second but still trails by 2.73%, 2.16%, and 0.94% in mIoU, mPA, and OA, respectively. Further comparisons reveal that our network surpasses CCNet, which utilizes the original Criss-Cross module. Our enhanced Criss-Cross attention module improves performance in mIoU, mPA, and OA by substantial margins of 3.23%, 1.54%, and 0.78%, respectively. Additionally, the inclusion of an auxiliary FCN head in our architecture enables it to outperform the native FCN network by 3.43% in mIoU, 2.62% in mPA, and 0.69% in OA. Against the classical PSPNet and DANet, LT-DeepLab also shows superior performance, leading by 3.28%, 2.79%, 0.50% and 2.90%, 2.01%, 0.52% in mIoU, mPA, and OA, respectively.

TABLE 1 Comparison of common semantic segmentation networks.

Model	mIoU	mPA	OA
DeepLabV3+(baseline)	72.99	83.53	95.36
FCN	73.15	83.40	95.30
CCNet	73.35	84.48	95.21
DANet	73.68	84.01	95.47
PSPNet	73.30	83.23	95.49
Non_Local	73.11	83.93	95.02
Segformer	73.85	83.86	95.05
UNet	70.42	81.70	93.02
LT-DeepLab	**76.58**	**86.02**	**95.99**

Bold indicates that this metric has the best performance.

Figure 5 visualizes the performance trends and stability across models. Figure 5A depicts the mIoU trends per 50 iterations, highlighting that LT-DeepLab reaches higher mIoU levels faster and maintains greater stability compared to others, surpassing other models' mIoU at 2500 iterations versus their 10000 iterations. Figure 5B shows the mPA change curves, with our model achieving significantly higher mPA after 3750 iterations. Figure 5C compares the OA curves, demonstrating that our model's curve is markedly more stable. Lastly, Figure 5D presents the loss variation curves; despite using the same cross-entropy loss function as other models, our network employs auxiliary loss, aiding in quicker convergence and resulting in a slightly higher initial loss value. These data further affirm the superior performance of our model.

To demonstrate the distinct performance of various networks more effectively, this paper compares their prediction results as depicted in Figure 6. This comparison highlights the challenges posed by real-environment field conditions and varying scales. The baseline network utilizes the original ASPP module for feature extraction, which fails to adequately detail the leaf edges and poorly integrates areas where trunks meet leaves, indicating limited segmentation capability.

The UNet network, despite its success in healthcare applications, underperforms on our dataset. This is likely due to its smaller number of parameters and the U-shaped with skip connections, optimized for simpler semantic tasks. In contrast, the complexity of leaf and stem disease segmentation in Zanthoxylum bungeanum Maxim proves too challenging, resulting in suboptimal outcomes. Non_Local excels in capturing long-distance dependencies and uniquely succeeds in correctly segmenting distant diseased spots as seen in Figure 6E. However, it still struggles with accurate feature extraction at the edges of leaves, diseased spots, and trunk regions. FCN, a classic semantic segmentation network, retains spatial feature information effectively using a fully convolutional structure. It performs well in identifying larger targets within images but is unable to adequately segment smaller, less significant ones, often ignoring them completely. PSPNet, which incorporates a pyramid pooling module, manages boundary information more effectively than many networks by capturing contextual details at various scales. Yet, like FCN, it often overlook minor targets, needing further improvements in overall segmentation. CCNet, designed to reduce the computational intensity inherent in Non_Local through its Criss-Cross Attention mechanism, slightly outperforms Non_Local in segmenting target edges according to the comparative prediction images. DANet, which integrates both spatial and channel attention mechanisms, achieves the highest accuracy among the traditional CNN networks. Nonetheless, it still neglects elements in the distance. Our proposed LT-DeepLab network outshines all compared networks by delivering superior segmentation of target boundary information—such as leaf-to-lesion, leaf-to-background, trunk-to-leaf, and trunk-to-background transitions. It markedly surpasses other models, especially in segmenting very small leaf lesions, underscoring its superiority over common semantic segmentation networks.

In addition to the classical networks mentioned above, this paper also compares several recently proposed models, as shown in Figure 7. Mask2Former (Cheng et al., 2022) integrates the masking

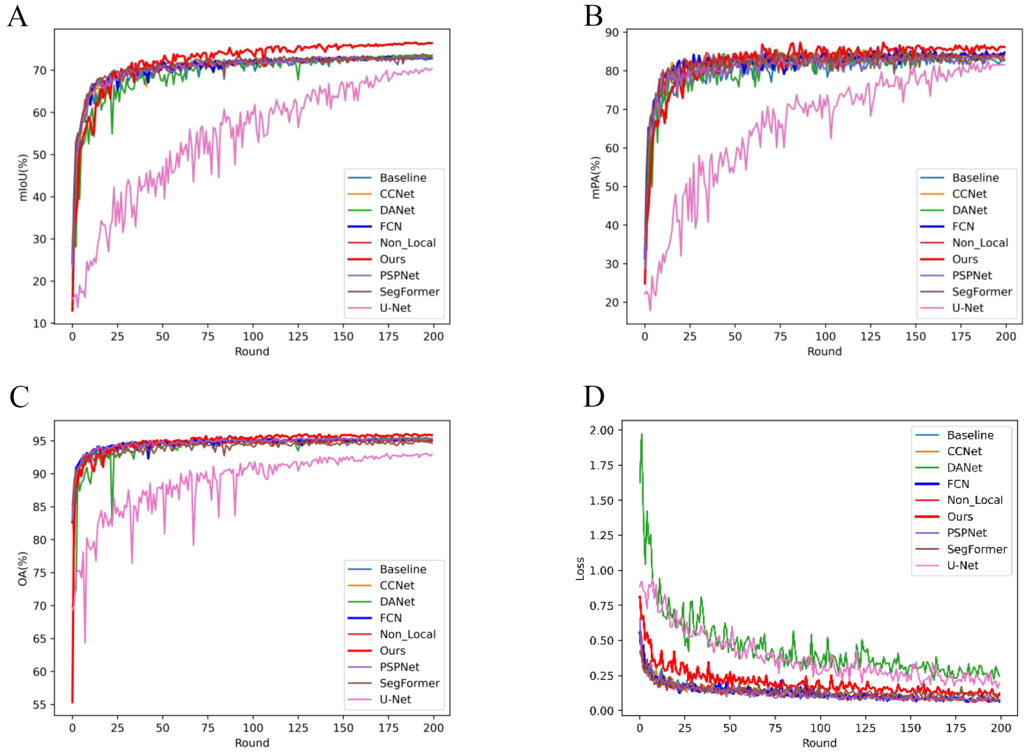


FIGURE 5
Evaluation Metrics and Loss Curve Analysis: This figure illustrates the progression of evaluation metrics and loss curves throughout the training process. Each Round consists of 50 iterations, with a total of 10,000 iterations completed.

technique and self-attention mechanism into a fully convolutional network, achieving performance second only to our proposed LT-DeepLab, with an mIoU of 76.19%. SegNeXt (Guo et al., 2022) updates the design of the traditional convolutional block and utilizes multi-scale convolutional features to evoke spatial attention through simple elemental multiplication, achieving an mIoU of 71.31%. SAN (Xu et al., 2023) and PID (Xu et al., 2023), which focus more on lightweight design, perform poorly on our dataset, with mIoU values of 62.48% and 65.24%, respectively.

To demonstrate the effectiveness of the attention mechanism proposed in this paper, we replaced the CRCC attention mechanism in the LT-DeepLab model with various other attention mechanisms while keeping all other conditions constant. The results are presented in the Table 2. Using the Criss-Cross Attention (CCA) alone results in lower accuracy due to insufficient contextual connections of channel features. The CBAM alone achieves better results by focusing on both channel and spatial features. The CRCC module proposed in this paper, which enhances channel features using CBAM while retaining the spatial contextual linking capability of Criss-Cross Attention module, achieves the best results across all metrics. The ELA module (Xu and Wan, 2024) extracts feature vectors in the horizontal and vertical directions using band-pooling in the spatial dimension, resulting in mIoU, mPA, and OA values of 76.13%, 85.86%, and 95.74%, respectively. The CA module (Hou et al., 2021) employs global average pooling of feature maps in both the width and height directions, then merges the two parallel phases, achieving mIoU, mPA, and OA values of 76.22%, 85.33%, and 95.89%, respectively. The EMA module (Ouyang et al., 2023) reshapes some channels to obtain the batch dimension and groups them into multiple sub-features to preserve channel information, resulting in mIoU, mPA, and OA values of 76.01%, 85.41%, and 95.86%, respectively. The ECA module (Wang et al., 2020) captures inter-channel dependencies using one-dimensional convolution, avoiding the complex upscaling and downscaling process, with mIoU, mPA, and OA values of 76.04%, 85.41%, and 95.86%, respectively. These data further confirm the effectiveness and superiority of the attention mechanism proposed in this paper.

3.2 Heat map visualization

To visually demonstrate the enhancements in our network, this paper employes gradient-weighted class activation mapping (Grad-CAM) (Selvaraju et al., 2017) to illustrate how effectively the model discriminates between different classes. Grad-CAM is a technique that visualizes neural network decisions by analyzing gradients in the final convolutional layer to determine the significance of each feature map relative to a specific class. This method generates heat maps that highlight areas of the image most relevant to the model's predictions. In Figure 8, this paper compares the heat maps from both the baseline and the improved versions of our model to showcase the differences pre- and post-enhancement. Each class is visualized separately to assess how effectively the network

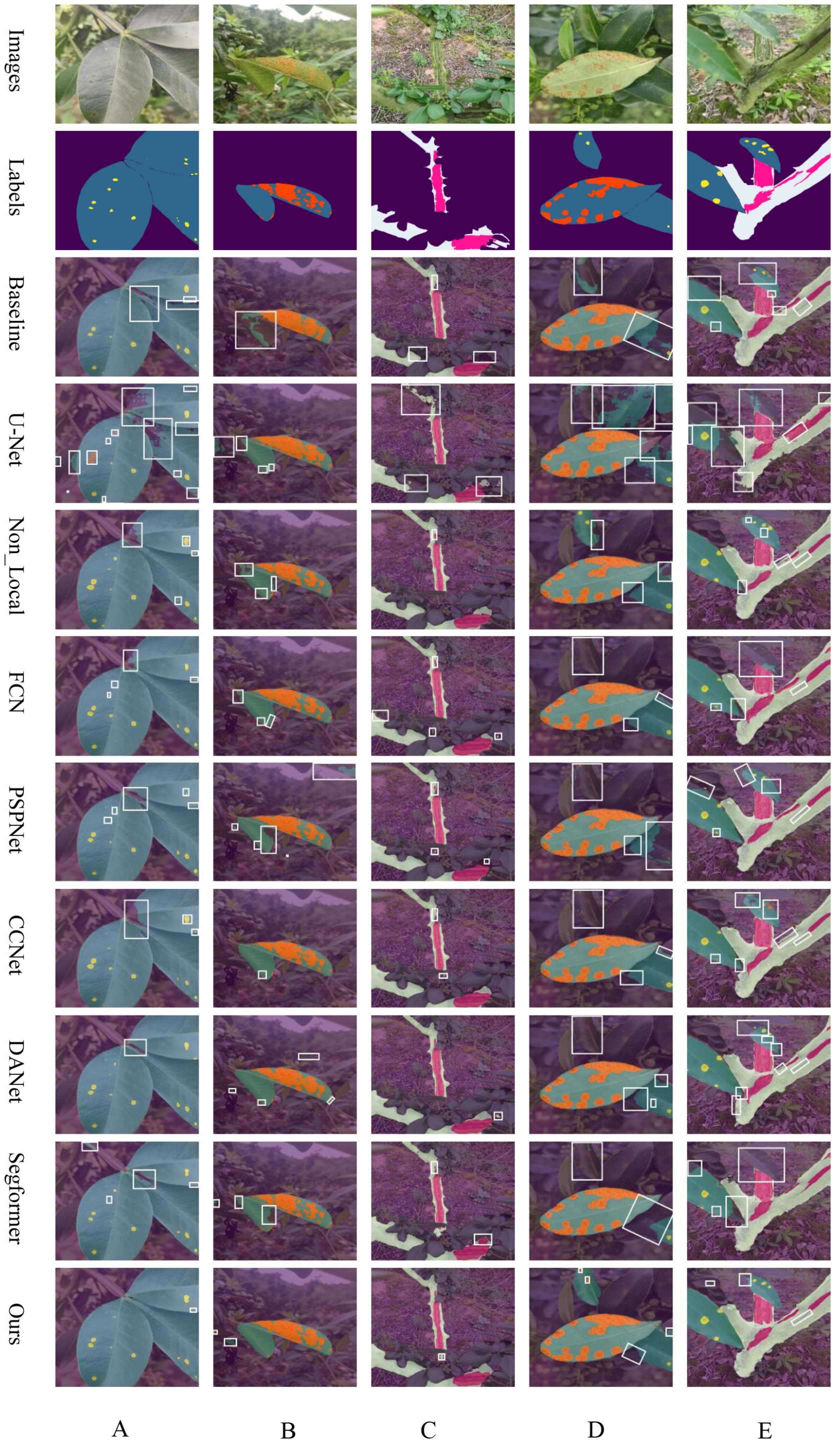


FIGURE 6
Overlay of Predicted Results from Common Networks: **(A)** leaf spot, **(B)** rust, **(C)** frost damage, **(D)** simultaneous rust and leaf spot, and **(E)** concurrent leaf spot and frost damage, respectively.

activates in response to that class. In the images, darker and more focused colors within the designated segmentation regions indicate stronger network activations, signifying better model performance and learning capability.

Upon comparison, it is evident that our model demonstrates superior category-specific activation compared to the baseline model. For instance, as shown in Figure 8A, our model is able to detect multiple diseased leaves against a complex background simultaneously, whereas the baseline model tends to recognize only few leaves. Similarly, in Figure 8D, while the baseline network struggles to accurately identify normal trunk sections, often misactivating diseased parts and some background areas, our model distinctly and correctly activates the normal trunk category. Overall, these observations confirm that our model

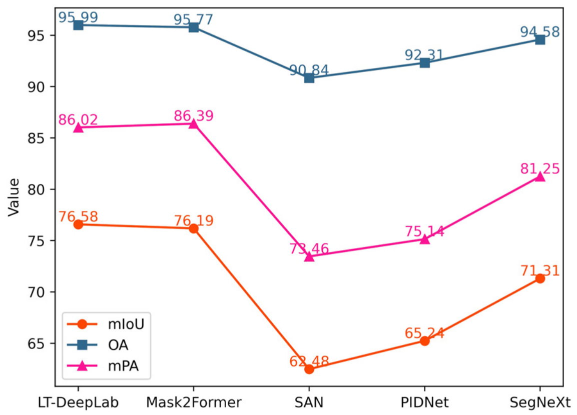


FIGURE 7
Frontier Model Comparison Line Chart.

achieves more precise class activation than the baseline model, validating the effectiveness of our enhanced attention mechanism.

3.3 Confusion matrix

Figures 9A, B present the confusion matrices for the baseline model and our proposed model on the test set, respectively. By applying row normalization to the confusion matrices, the diagonal values in each matrix reflect the pixel accuracy of each category. A comparison of these values clearly demonstrates that our model achieves superior segmentation performance across all categories. For instance, an analysis of the first row shows that our model more effectively distinguishes between each category and the background, exhibiting significantly better performance in real-world environment segmentation compared to the baseline model.

3.4 Ablation experiments

To evaluate the effectiveness of our improved modules, this paper initially examines the impact of employing multiple auxiliary heads, as detailed in Table 3. Additionally, sixteen sets of ablation experiments are conducted to validate each of the four modules discussed in this paper, focusing on various metrics including model size and computational efficiency. These experiments are summarized in Table 4, which reports on metrics such as mIoU, mPA, OA, the floating-point operations (FLOPs), and the number of parameters (Params).The baseline configuration, native DeepLabv3+, utilizes a multilayer convolutional structure within the original ASPP, leading to a high computational demand of 0.27 TFLOPs and a parameter count of 65.74M, achieving an mIoU of 72.99%. As demonstrated in Table 4 (2), the implementation of auxiliary loss results in a 0.15% improvement in mIoU. Importantly, since auxiliary loss only influences the training phase, it does not increase the number of parameters or computational load during the inference process. In experiment 3(3), integrating deformable convolution with low-dimensional feature maps resulted in a 1.47% increase in mIoU. Table 4 (5) highlights the enhancements to the ASPP module through lightweight and attention-fused deformable convolutions, which not only elevate mIoU by 2.54% but also reduce the computational demand by 0.05 TFLOPs and decrease the parameter count by 13.37M. Further, as shown in Table 4 (9), the inclusion of our improved CRCC module based on CCNet slightly raised the mIoU by 0.01%. A comprehensive comparison from Tables 4 (8) and 3(16) demonstrates that the collective application of all improvements, with and without the CRCC module, boosts the mIoU by 0.46%, mPA by 0.01%, and OA by 0.28%, confirming the overall efficacy of our enhancements.

TABLE 2 Comparison of different attention mechanisms in LT-DeepLab.

Attention	mIoU	mPA	OA
CCA	75.81	86.02	95.77
CBAM	76.29	85.65	95.91
ELA	76.13	85.86	95.74
CA	76.22	85.33	95.89
EMA	76.01	85.86	95.80
ECA	76.04	85.41	95.86
CRCC (our)	**76.58**	**86.02**	**95.99**

Bold indicates that this metric has the best performance.

In pursuit of an optimal auxiliary head to further optimize training, this paper evaluates four different designs that generate auxiliary loss during training only. The results are summarized in Table 3, which led us to select the FCNHead as our auxiliary head due to its superior performance in mIoU and OA.

Figure 10 is a scatter Plot of mIoU versus Number of Parameters. This scatter plot illustrates the trade-off between model complexity and segmentation accuracy across the entire set of ablation experiments. It visualizes the relationship between the accuracy of each experimental group and their corresponding number of parameters. Notably, our model attains the highest mIoU while maintaining a relatively low parameter count, demonstrating its efficiency and effectiveness in segmentation tasks.

To validate the robustness of the LT-DeepLab network proposed in this paper, various backbone networks were employed for feature extraction. As depicted in Table 5, the experiments were divided into five groups. Each group compared the original DeepLabV3+ model with the corresponding backbone to the proposed LT-DeepLab model using the same backbone. Notably, in each group, the LT-DeepLab architecture consistently achieved the best segmentation performance across all metrics. The most significant improvement was observed in the first group, with an increase of 17.84% in mIoU, 18.14% in mPA, and 3.58% in OA. These enhancements can be attributed to the superior contextualization and feature integration capabilities of the LT-DeepLab network. The results clearly illustrate that the LT-DeepLab architecture is robust and versatile, making it a suitable choice for various feature extraction backbone networks.

Table 6 presents the IoU values for each category in our proposed model, indicating that the IoU for leaf spot disease is the lowest. One probable reason for this is the small size and irregular boundaries of these spots. As depicted in Figure 11, these spots have a range of faded green areas around the brown spots, and

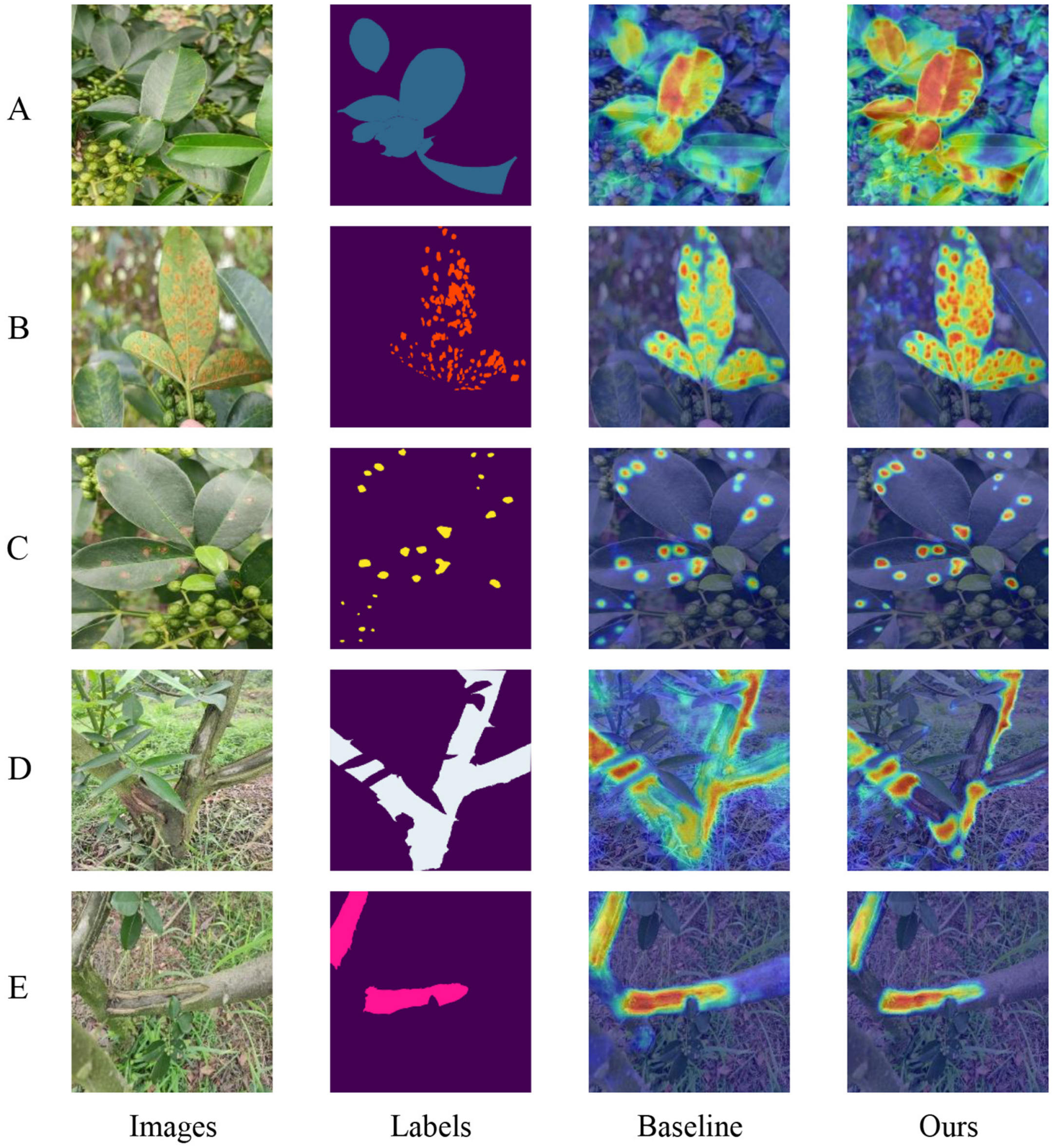


FIGURE 8
Category Weight Activation Maps: Panels A through E display the category weight activation maps for different conditions: **(A)** diseased leaves, **(B)** rust, **(C)** leaf spot, **(D)** diseased trunks, and **(E)** frost-damaged areas.

in the early stages of the disease, only faded green spots are present without any brown spots. To ensure that the model can detect this type of disease even in its early stages, this paper includes the faded green areas in our labeling. However, the faintness of these boundaries results in less accurate information extraction, leading to lower segmentation accuracy for this category. Although this may reduce accuracy, it is crucial for identifying early-stage disease and enabling timely intervention to prevent further damage.

3.5 Module efficiency

To evaluate the computational efficiency of the primary modules proposed by LT-DeepLab, this study examines the number of parameters and computation time of the CRCC attention module and the FDCASPP structure. The results are presented in Table 7. The CRCC module is utilized in both the backbone feature extraction network and the FDCASPP structure, with the only difference being the number of intermediate channels. In the backbone, the CRCC module has 256 intermediate channels, whereas in the FDCASPP, it has 2048 intermediate channels. This design choice balances computational speed and segmentation accuracy. The average frame rate of the LT-DeepLab model during inference is 12.56 fps.

3.6 Significance test results

The results of the independent samples t-test indicate that the mean performance of the improved LT-DeepLab model is

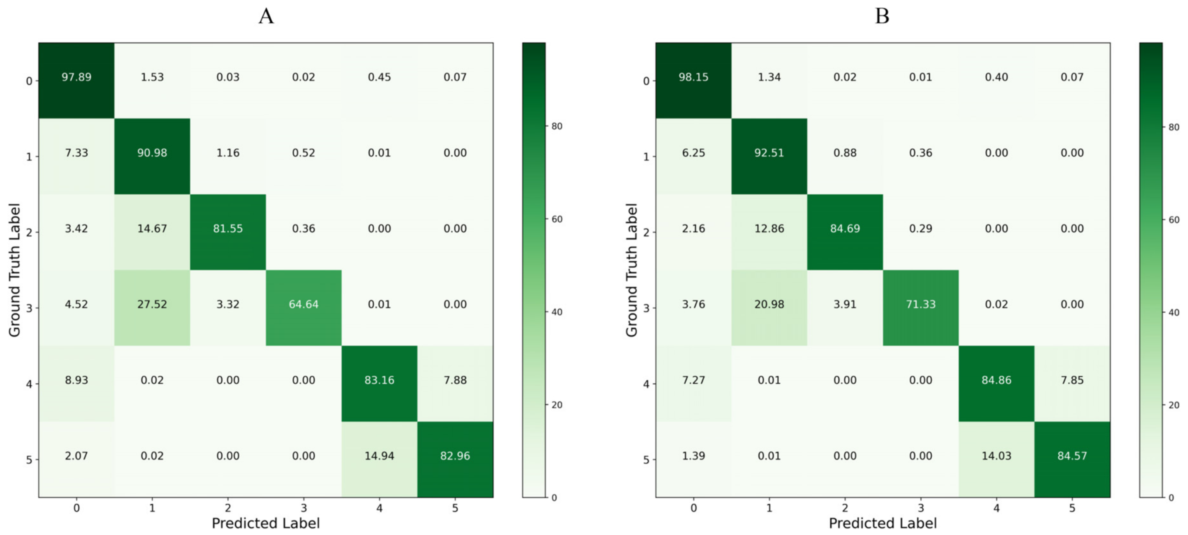


FIGURE 9
Confusion Matrix Comparison: Panel **(A)** displays the confusion matrix for the baseline model on the test set, and Panel **(B)** shows the confusion matrix for the LT-DeepLab model on the test set.

significantly higher than that of the baseline model. Specifically, the five experimental outcomes for LT-DeepLab were 76.48, 76.49, 76.49, 76.58, and 76.46, while the five outcomes for the baseline model were 72. 85, 72.88, 72.85, 72.99, and 72.82. The p-value obtained from the t-test was substantially lower than the commonly accepted significance level of 0.05, allowing us to reject the null hypothesis, thereby confirming that the mean difference between the two models is statistically significant. This finding demonstrates that LT-DeepLab significantly outperforms the baseline model in enhancing performance.

4 Discussion

In this study, we propose a novel CRCC attention mechanism integrated with DeepLabV3+, which simultaneously considers both spatial and channel-wise features. This mechanism skillfully combines the Criss-Cross attention with the CBAM mechanism, addressing the Criss-Cross attention's limitation in capturing channel-wise features and enhancing overall performance. Additionally, we introduce a new cross-scale solution header, FCDASPP, which, compared to the original ASPP module, reduces the number of parameters by employing depthwise separable convolutions. This approach, when combined with the CRCC attention mechanism, significantly improves segmentation capabilities. Furthermore, the inclusion of an FCN auxiliary head enhances segmentation performance during training by participating solely in the loss computation, thereby avoiding any additional overhead during inference. The incorporation of deformable convolutions allows the convolutional kernels to learn offsets, facilitating the effective handling of shallow features and enabling more efficient extraction and fusion of shallow and deep features.

TABLE 3 Comparison of auxiliary head performance.

Name	mIoU	mPA	OA
DWFCNHead	76.28	86.07	95.95
DAHead	76.13	**86.33**	95.84
PSPHead	76.29	85.91	95.91
FCNHead	**76.58**	86.02	**95.99**

Bold indicates that this metric has the best performance.

In the field of research on segmentation of leaf or trunk diseases in Zanthoxylum bungeanum Maxim, Yang et al. (2021) introduced a fifth ASPP branch into DeepLabv2 to segment rust disease in a controlled laboratory environment, achieving an mIoU of 84.99%. Zhang et al. (2024) proposed a lightweight U-shaped perceptual

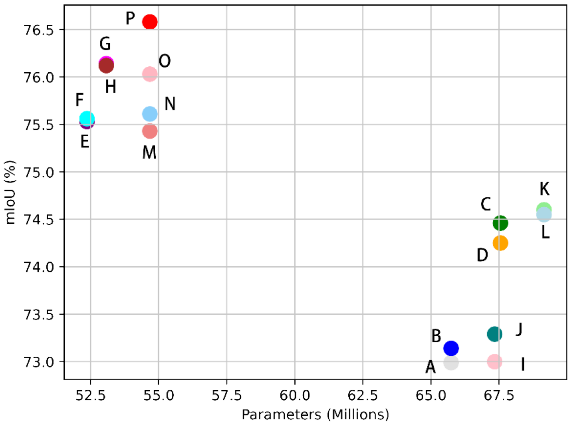


FIGURE 10
Scatter Plot of Evaluation Indicators versus Number of Parameters: Points A through P on the plot correspond to data entries 1 through 16 in Table 4, respectively.

TABLE 4 Comparison of evaluation indexes of ablation experiments with parametric quantities and calculation quantities.

Num	CRCC	FDCASPP	DCN	Aux_Loss	mIoU	mPA	OA	FLOPs/T	Params/M
1	✗	✗	✗	✗	72.99	83.53	95.36	0.27	65.74
2	✗	✗	✗	✓	73.14	84.03	95.34	0.27	65.74
3	✗	✗	✓	✗	74.46	84.26	95.52	0.29	67.55
4	✗	✗	✓	✓	74.25	84.05	95.48	0.29	67.55
5	✗	✓	✗	✗	75.53	85.15	95.69	**0.22**	**52.37**
6	✗	✓	✗	✓	75.56	85.44	95.70	**0.22**	**52.37**
7	✗	✓	✓	✗	76.14	85.63	95.66	0.23	53.08
8	✗	✓	✓	✓	76.12	86.01	95.71	0.23	53.08
9	✓	✗	✗	✗	73.00	83.13	95.35	0.28	67.34
10	✓	✗	✗	✓	73.29	83.52	95.46	0.28	67.34
11	✓	✗	✓	✗	74.60	84.54	95.14	0.29	69.15
12	✓	✗	✓	✓	74.55	84.97	95.71	0.29	69.15
13	✓	✓	✗	✗	75.43	85.11	95.75	0.23	54.68
14	✓	✓	✗	✓	75.61	85.43	95.78	0.23	54.68
15	✓	✓	✓	✗	76.03	85.40	95.88	0.24	54.68
16	✓	✓	✓	✓	**76.58**	**86.02**	**95.99**	0.24	54.68

In the table, a checkmark (✓) indicates that a specific module was included in that group of experiments, while a cross (✗) signifies that the module was not incorporated in that particular experimental setup.
Bold indicates that this metric has the best performance.

transformer for grape leaf disease segmentation, which strikes a balance between performance and efficiency. However, this method may not be suitable for field conditions and is limited to a small number of diseases, indicating that its practical applicability needs improvement. Wang et al. (2021) employed a two-stage segmentation approach for cucumber leaf disease in complex environments, using two networks sequentially to segment different targets, thereby achieving higher segmentation accuracy. However, this method incurs significant computational overhead, making it less practical for real-world production applications. The model proposed in this paper maintains accuracy while reducing the size compared to the benchmark network, enabling robust segmentation of multiple diseases simultaneously. Similar to the study by (Zhou et al., 2024), we also chose to improve the DeepLabV3+ model. The difference lies in their introduction of a gated pyramid feature fusion structure, which connects features of different scales using a specialized gating mechanism while capturing different receptive fields. The FDCASPP structure proposed in this paper is fundamentally designed to fuse multi-scale features and enhance the connections between them through the CRCC attention mechanism. Furthermore, Wang et al. (2024) demonstrated that integrating the CBAM attention mechanism with residuals into the UNet network enhances its feature extraction capability and ability to capture fine-grained information, utilizing an improved ASPP module. Our study extends this idea by complementing CBAM with the Criss-Cross attention mechanism, incorporating this combination into the

TABLE 5 Comparison of different feature extraction backbone networks in LT-DeepLab.

Model	mIoU	mPA	OA
DeepLabV3Plus+replknet	53.75	64.96	90.75
LT-DeepLab+replknet	**71.59**	**83.10**	**94.33**
DeepLabV3Plus+vgg16	60.66	70.94	93.20
LT-DeepLab+vgg16	**73.49**	**83.91**	**95.30**
DeepLabV3Plus +MobileNetV3	56.79	67.30	92.19
LT-DeepLab+ MobileNetV3	**70.07**	**81.18**	**94.51**
DeepLabV3Plus+ResNeSt	69.76	80.28	94.56
LT-DeepLab+ ResNeSt	**72.59**	**83.03**	**95.06**
DeepLabV3Plus +ResNet(baseline)	72.99	83.53	95.36
LT-DeepLab+ResNet(our)	**76.58**	**86.02**	**95.99**

Bold indicates that this metric has the best performance.

TABLE 6 Comparison of individual category IoUs between the baseline model and our model.

	0	1	2	3	4	5
Baseline	95.59	84.20	64.75	50.68	72.84	69.89
Ours	96.22	86.47	70.33	59.94	75.14	71.38

where 1-5 denote background, diseased leaves, rust, leaf spot, diseased trunks, frost damage respectively.

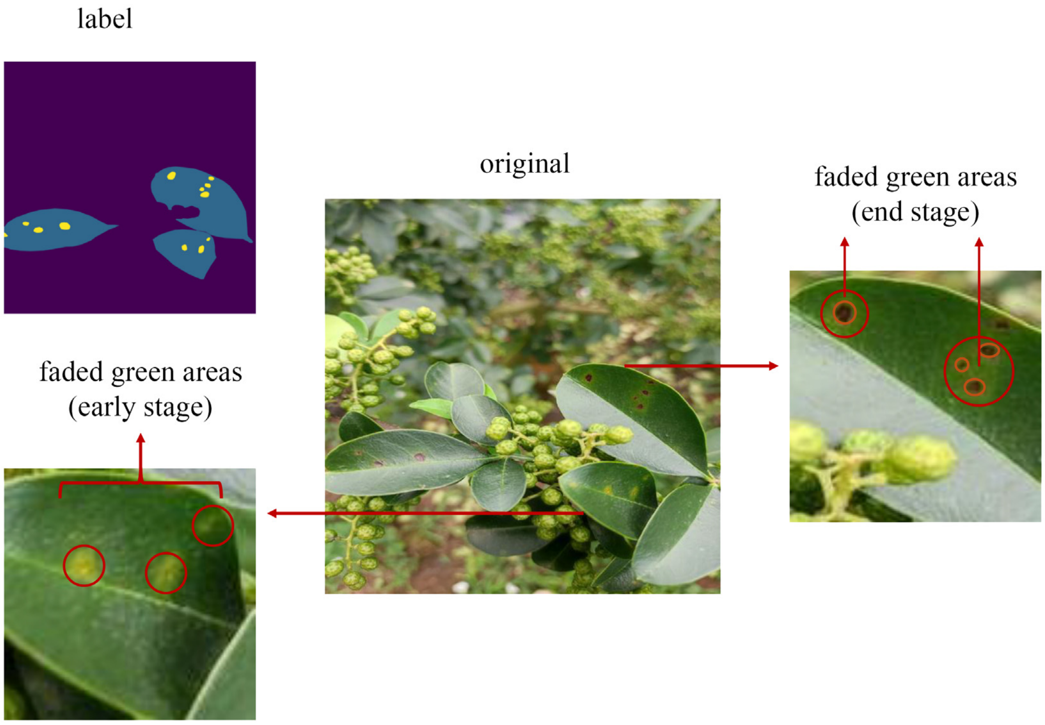


FIGURE 11
Detail of faded green areas.

FDCASPP structure to further enhance multi-scale feature extraction. Zhu et al. (2024) replaced dilated convolutions in the ASPP module with deformable convolutions to address issues such as poor segmentation accuracy for irregular defects in navel orange surface defect detection. Similarly, this study employs deformable convolutions in the self-proposed FDCASPP structure to assist in feature extraction, particularly in the decoder stage, to better fit irregular lesions on leaves and trunks, further demonstrating the effectiveness of deformable convolutions in extracting features from irregular targets. However, when comparing our study with the UNet network, unlike (Han et al., 2024) success in brain tumor image segmentation by integrating the Criss-Cross attention mechanism into UNet, the U-Net network performed poorly on our dataset. This could be attributed to UNet's skip connection structure, which directly combines low-level and high-level features, leading to issues such as unclear boundaries when dealing with targets that have fuzzy edges or are difficult to distinguish from the background. In our dataset, diseased leaves are particularly challenging to differentiate from healthy ones in a complex environment, with the boundaries of the lesions being especially indistinct.

The model proposed in this paper strikes a balance between accuracy and model size when compared to the baseline network. It also demonstrates strong robustness by effectively segmenting multiple diseases simultaneously. Future work will focus on validating the model's generalizability using other datasets. Despite the current model's success in reducing computational load and parameter count, there remains room for further optimization. Future research will aim to develop a lighter and more efficient network, facilitating easier deployment in field

TABLE 7 Comparison of different feature extraction backbone networks in LT-DeepLab.

Module	Location	Params /M	-	Time/s	LT-DeepLab
CRCC	CRCC_backbone	1.1	Train_time	0.0273	12.56fps
			Test_time	0.0091	
	CRCC_FDCASPP	6.1	Train_time	0.0361	
			Test_time	0.0145	
FDCASPP	–	17.1	Train_time	0.0991	
			Test_time	0.0465	

The data dimensions of the aforementioned modules during training are (4, 256, 64, 64), while during testing, the dimensions are (1, 256, 86, 64). This discrepancy in feature map sizes between training and testing is attributable to the differing data augmentation strategies employed.

environments. Additionally, the possibility of segmenting peppercorn fruits will be explored, enabling real-time monitoring and segmentation of Zanthoxylum bungeanum Maxim diseases to enhance the efficiency and productivity of peppercorn cultivation.

5 Conclusion

To address the challenge of integrated segmentation of diseases on Zanthoxylum bungeanum Maxim leaves and trunks, this research proposes an enhanced version of DeepLabv3+, named LT-DeepLab. This method innovatively applies semantic segmentation technology for joint disease targeting on both leaves and trunks. This paper have improved the Criss-Cross Attention module by integrating the channel-space attention capabilities of the CBAM, and proposed a new attention mechanism, the CRCC module, which accurately extracts edge information of leaves and trunks. Additionally, a deformable convolution module has been implemented to effectively capture low-dimensional information, enhancing the fusion with high-dimensional feature maps. Addressing the issue of the original ASPP module's high parameter count and limited cross-scale information extraction capability, this paper has developed the FDCASPP module, designed to enhance the extraction of multi-scale information and improve target segmentation in complex backgrounds. Experimental results demonstrate that LT-DeepLab's segmentation capabilities in complex environments surpass those of other commonly used semantic segmentation networks. Relative to the baseline model, LT-DeepLab not only reduces the number of parameters and computational demands but also achieves superior performance across all evaluation metrics.

Data availability statement

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

Author contributions

TY: Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing. JW: Data curation, Formal analysis, Investigation, Writing – original draft. YX: Data curation, Visualization, Writing – original draft. SW: Data curation, Visualization, Writing – original draft. JT: Validation, Writing – original draft. YN: Validation, Writing – original draft. XD: Funding acquisition, Project administration, Writing – original draft. FP: Funding acquisition, Project administration, Supervision, Writing – review & editing. HP: Project administration, Supervision, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work is supported by Sichuan Agricultural University Research Initiation Project (No. 2022996013).

Acknowledgments

Thanks to Prof. Jingyan Wang, and the support from Agricultural Information Engineering Higher Institution Key Laboratory of Sichuan Province, Ya'an Digital Agricultural Engineering Technology Research Center, the AI lab at Sichuan Agricultural 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.

References

Attri, I., Awasthi, L. K., Sharma, T. P., and Rathee, P. (2023). A review of deep learning techniques used in agriculture. *Ecol. Inf.* 77, 102217. doi: 10.1016/j.ecoinf.2023.102217

Barbedo, J. (2016). A novel algorithm for semi-automatic segmentation of plant leaf disease symptoms using digital image processing. *Trop. Plant Pathol.* 41, 210–224. doi: 10.1007/s40858-016-0090-8

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *15th European Conference on Computer Vision*, Berlin, Springer.

Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). "Masked-attention mask transformer for universal image segmentation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ: IEEE.

Cruz, A., Ampatzidis, Y., Pierro, R., Materazzi, A., Panattoni, A., De Bellis, L., et al. (2019). Detection of grapevine yellows symptoms in Vitis vinifera L. with artificial intelligence. *Comput. Electron. Agric.* 157, 63–76. doi: 10.1016/j.compag.2018.12.028

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., et al. (2017). "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Piscataway, NJ: IEEE.

Deng, Y., Xi, H., Zhou, G., Chen, A., Wang, Y., Li, L., et al. (2023). An effective image-based tomato leaf disease segmentation method using MC-UNet. *Plant Phenomics* 5, 0049. doi: 10.34133/plantphenomics.0049

Gao, L., and Lin, X. (2019). Fully automatic segmentation method for medicinal plant leaf images in complex background. *Comput. Electron. Agric.* 164, 104924. doi: 10.1016/j.compag.2019.104924

Guo, M.-H., Lu, C.-Z., Hou, Q., Liu, Z., Cheng, M.-M., and Hu, S.-M. (2022). Segnext: Rethinking convolutional attention design for semantic segmentation. *Adv. Neural Inf. Process. Syst.* 35, 1140–1156.

Han, X., Liu, J., and Zhao, J. (2024). "U-CCNet: brain tumor MRI image segmentation model with broader global context semantic information abstraction," in *2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. (Piscataway, NJ: IEEE), 1550–1554.

Hou, Q., Zhou, D., and Feng, J. (2021). "Coordinate attention for efficient mobile network design," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ: IEEE.

Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., and Liu, W. (2019). "Ccnet: Criss-cross attention for semantic segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Piscataway, NJ: IEEE.

Javidan, S. M., Banakar, A., Vakilian, K. A., and Ampatzidis, Y. (2023). Diagnosis of grape leaf diseases using automatic K-means clustering and machine learning. *Smart Agric. Technol.* 3, 100081. doi: 10.1016/j.atech.2022.100081

Jodas, D. S., Brazolin, S., Yojo, T., De Lima, R. A., Velasco, G. D. N., MaChado, A. R., et al. (2021). "A deep learning-based approach for tree trunk segmentation," in *2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Piscataway, NJ: IEEE.

Li, D., Cao, Y., Tang, X.-s., Yan, S., and Cai, X. (2018). Leaf segmentation on dense plant point clouds with facet region growing. *Sensors* 18, 3625. doi: 10.3390/s18113625

Liu, Y., Yu, Q., and Geng, S. (2024). Real-time and lightweight detection of grape diseases based on Fusion Transformer YOLO. *Front. Plant Sci.* 15, 1269423. doi: 10.3389/fpls.2024.1269423

Lu, J., Xiang, J., Liu, T., Gao, Z., and Liao, M. (2022). Sichuan pepper recognition in complex environments: A comparison study of traditional segmentation versus deep learning methods. *Agriculture* 12, 1631. doi: 10.3390/agriculture12101631

Ma, J., Du, K., Zhang, L., Zheng, F., Chu, J., and Sun, Z. (2017). A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. *Comput. Electron. Agric.* 142, 110–117. doi: 10.1016/j.compag.2017.08.023

Mzoughi, O., and Yahiaoui, I. (2023). Deep learning-based segmentation for disease identification. *Ecol. Inf.* 75, 102000. doi: 10.1016/j.ecoinf.2023.102000

Nahiduzzaman, M., Chowdhury, M. E., Salam, A., Nahid, E., Ahmed, F., Al-Emadi, N., et al. (2023). Explainable deep learning model for automatic mulberry leaf disease classification. *Front. Plant Sci.* 14, 1175515. doi: 10.3389/fpls.2023.1175515

Ouyang, D., He, S., Zhang, G., Luo, M., Guo, H., Zhan, J., et al. (2023). "Efficient multi-scale attention module with cross-spatial learning," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Piscataway, NJ: IEEE.

Pal, A., and Kumar, V. (2023). AgriDet: Plant Leaf Disease severity classification using agriculture detection framework. *Eng. Appl. Artif. Intell.* 119, 105754. doi: 10.1016/j.engappai.2022.105754

Qi, M., and Jia, G. (2023). "Infrared small target detection algorithm based on improved deepLabV3+," in *2023 8th International Conference on Intelligent Computing and Signal Processing (ICSP)*, Piscataway, NJ: IEEE.

Savary, S., Willocquet, L., Pethybridge, S. J., Esker, P., McRoberts, N., and Nelson, A. (2019). The global burden of pathogens and pests on major food crops. *Nat. Ecol. Evol.* 3, 430–439. doi: 10.1038/s41559-018-0793-y

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Piscataway, NJ: IEEE.

Shedthi, B. ,. S., Siddappa, M., Shetty, S., Shetty, V., and Suresh, R. (2023). Detection and classification of diseased plant leaf images using hybrid algorithm. *Multimedia Tools Appl.* 82, 32349–32372. doi: 10.1007/s11042-023-14751-0

Sifre, L., and Mallat, S. (2014). Rigid-motion scattering for texture classification. *arXiv*. arXiv:1403.1687. doi: 10.48550/arXiv.1403.1687

Thai, H.-T., Le, K.-H., and Nguyen, N. L.-T. (2023). FormerLeaf: An efficient vision transformer for Cassava Leaf Disease detection. *Comput. Electron. Agric.* 204, 107518. doi: 10.1016/j.compag.2022.107518

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., and Zhu, H. (2021). A cucumber leaf disease severity classification method based on the fusion of DeepLabV3+ and U-Net. *Comput. Electron. Agric.* 189, 106373. doi: 10.1016/j.compag.2021.106373

Wang, J., Jia, J., Zhang, Y., Wang, H., and Zhu, S. (2024). RAAWC-UNet: an apple leaf and disease segmentation method based on residual attention and atrous spatial pyramid pooling improved UNet with weight compression loss. *Front. Plant Sci.* 15, 1305358. doi: 10.3389/fpls.2024.1305358

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

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). "Cbam: Convolutional block attention module," in *15th European Conference on Computer Vision*, Berlin: Springer.

Xu, B., Wang, N., Chen, T., and Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint*. arXiv:1505.00853. doi: 10.48550/arXiv.1505.00853

Xu, J., Xiong, Z., and Bhattacharyya, S. P. (2023). "PIDNet: A real-time semantic segmentation network inspired by PID controllers," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ: IEEE.

Xu, M., Zhang, Z., Wei, F., Hu, H., and Bai, X. (2023). "Side adapter network for open-vocabulary semantic segmentation," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ: IEEE.

Xu, W., and Wan, Y. (2024). ELA: efficient local attention for deep convolutional neural networks. *arXiv preprint*. arXiv:2403.01123. doi: 10.48550/arXiv.2403.01123

Yang, F., Xu, J., Wei, H., Ye, M., Xu, M., Fu, Q., et al. (2021). Multi-scale image segmentation model for fine-grained recognition of Zanthoxylum rust. *Computers, Materials and Continua* 71, 2963–2980. doi: 10.32604/cmc.2022.022810

Zhang, X., Li, F., Jin, H., and Mu, W. (2023). Local Reversible Transformer for semantic segmentation of grape leaf diseases. *Appl. Soft Computing* 143, 110392. doi: 10.1016/j.asoc.2023.110392

Zhang, X., Li, F., Zheng, H., and Mu, W. (2024). UPFormer: U-shaped perception lightweight transformer for segmentation of field grape leaf diseases. *Expert Syst. Appl.* 249, 123546. doi: 10.1016/j.eswa.2024.123546

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). "Pyramid scene parsing network," in *2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ: IEEE.

Zhou, H., Peng, Y., Zhang, R., He, Y., Li, L., and Xiao, W. (2024). GS-deepLabV3+: A mountain tea disease segmentation network based on improved shuffle attention and gated multidimensional feature extraction. *Crop Prot.* 183, 106762. doi: 10.1016/j.cropro.2024.106762

Zhu, Y., Liu, S., Wu, X., Gao, L., and Xu, Y. (2024). Multi-class segmentation of navel orange surface defects based on improved deeplabv3+. *J. Agric. Eng.* 55, 263–275. doi: 10.4081/jae.2024.1564

Zhu, S., Ma, W., Lu, J., Ren, B., Wang, C., and Wang, J. (2023). A novel approach for apple leaf disease image segmentation in complex scenes based on two-stage DeepLabv3+ with adaptive loss. *Comput. Electron. Agric.* 204, 107539. doi: 10.1016/j.compag.2022.107539

TYPE Original Research
PUBLISHED 19 November 2024
DOI 10.3389/fpls.2024.1453761

OPEN ACCESS

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

REVIEWED BY
Yueming Wang,
CAS, China
Zipeng Zhang,
Xinjiang University, China
Elio Romano,
Centro di ricerca per l'Ingegneria e le Trasformazioni agroalimentari (CREA-IT), Italy
Yahui Guo,
Central China Normal University, China

*CORRESPONDENCE
Jing Zhao
✉ zhaojscau@163.com
Si Sun
✉ SUNSI@scau.edu.cn
Tianyi Liu
✉ tianyiliu0202@126.com

RECEIVED 24 June 2024
ACCEPTED 29 October 2024
PUBLISHED 19 November 2024

CITATION
Liu W, Xie Z, Du J, Li Y, Long Y, Lan Y, Liu T, Sun S and Zhao J (2024) Early detection of pine wilt disease based on UAV reconstructed hyperspectral image.
Front. Plant Sci. 15:1453761.
doi: 10.3389/fpls.2024.1453761

Early detection of pine wilt disease based on UAV reconstructed hyperspectral image

Wentao Liu[1,2,3], Ziran Xie[1,3], Jun Du[1,3], Yuanhang Li[1,3], Yongbing Long[1,3], Yubin Lan[1,3], Tianyi Liu[2]*, Si Sun[2]* and Jing Zhao[1,3]*

[1]College of Electronic Engineering (College of Artificial Intelligence), South China Agricultural University, Guangzhou, China, [2]College of Forestry and Landscape Architecture, South China Agricultural University, Guangzhou, China, [3]National Center for International Collaboration Research on Precision Agricultural Aviation Pesticides Spraying Technology, Guangzhou, China

Pine wilt disease (PWD) is a highly destructive infectious disease that affects pine forests. Therefore, an accurate and effective method to monitor PWD infection is crucial. However, the majority of existing technologies can detect PWD only in the later stages. To curb the spread of PWD, it is imperative to develop an efficient method for early detection. We presented an early stage detection method for PWD utilizing UAV remote sensing, hyperspectral image reconstruction, and SVM classification. Initially, employ UAV to capture RGB remote sensing images of pine forests, followed by labeling infected plants using these images. Hyperspectral reconstruction networks, including HSCNN+, HRNet, MST++, and a self-built DW3D network, were employed to reconstruct the RGB images obtained from remote sensing. This resulted in hyperspectral images in the 400-700nm range, which were used as the dataset of early PWD detection in pine forests. Spectral reflectance curves of infected and uninfected plants were extracted. SVM algorithms with various kernel functions were then employed to detect early pine wilt disease. The results showed that using SVM for early detection of PWD infection based on reconstructed hyperspectral images achieved the highest accuracy, enabling the detection of PWD in its early stage. Among the experiments, MST++, DW3D, HRNet, and HSCNN+ were combined with Poly kernel SVM performed the best in terms of cross-validation accuracy, achieving 0.77, 0.74, 0.71, and 0.70, respectively. Regarding the reconstruction network parameters, the DW3D network had only 0.61M parameters, significantly lower than the MST++ network, which had the highest reconstruction accuracy with 1.6M parameters. The accuracy was improved by 27% compared to the detection results obtained using RGB images. This paper demonstrated that the hyperspectral reconstruction-poly SVM model could effectively detect the Early stage of PWD. In comparison to

UAV hyperspectral remote sensing methods, the proposed method in this article offers a same precision, but a higher operational efficiency and cost-effectiveness. It also enables the detection of PWD at an earlier stage compared to RGB remote sensing, yielding more accurate and reliable results.

KEYWORDS

pine wilt disease (PWD), early stage detection, UAV remote sensing, hyperspectral image reconstruction, supervised classification

1 Introduction

Pine wilt disease (PWD) is a highly destructive conifer disease with a global impact, affecting numerous countries and regions. PWD initially originated in North America (Ikegami and Jenkins, 2018) but has now spread extensively across East Asia, resulting in significant damage to billions of pine trees and substantial economic and ecological losses (Zhao et al., 2020a; Hao et al., 2022). Therefore, timely detection of PWD is crucial as it enables us to comprehend the current infection situation and implement appropriate measures to prevent further spread. The existing classification of PWD disease cycles primarily defines infection stages based on resin secretion, growth vigor, and needle color (Yu, R et al., 2021a; 2021b). The traditional and widely used method for PWD detection involves analyzing wood samples collected from trees in the chemistry laboratory to detect the presence of pine wood nematode (PWN) (Futai, 2013). However, this method necessitates expertise in nematology and is time-consuming. Additionally, collecting samples over large areas becomes challenging due to the complex terrain of mountainous regions. Thus, there is an urgent need to develop a rapid, non-destructive, and large-scale early detection method.

Recently, UAV-based remote sensing has emerged as a promising approach for detecting forest pests and diseases (Yang et al., 2017; Xiao et al., 2022; Du et al., 2024). The integration of UAV-based remote sensing in plant disease detection has garnered extensive acceptance due to its cost-effectiveness, flexibility, and superior spatial resolution, surpassing conventional manual detection methods (Bagheri, 2020; Sangaiah et al., 2024; Shan et al., 2024; Zhao et al., 2023; Guo et al., 2023). On the other hand, a large number of studies show that there are differences in the visial-infrared spectral band of the canopy before and after PWD infection (Kim et al., 2018; Yu et al., 2021a). Based on this fact, researchers have begun using the method of spectral method to do the PWD detection (Li et al., 2022; Yu, R et al., 2021b; Rao et al., 2023; Oide et al., 2022), the U-shaped network architecture, known for its efficacy in capturing features across varied scales of image space dimensions, is adopted. (Zeng et al., 2021). So to combine these two methods together has become a hotspot in PWD research. Certain research methods integrating UAV and spectral images with target detection algorithms enables direct image-based detection (Wu et al., 2021; Yu et al., 2021c). Take Wu, B and Yu, R., for example. Wu, B et al. trained four deep learning models for detection purposes, which utilized collected images in the 390-760nm range to detect Early stage PWD, resulting in an average detection accuracy of 50.2%. Yu, R., et al. used the random forest based on cart decision tree model to detect PWD early infection stage, which extracted 37 spectral variables from hyperspectral data and the recognition accuracy reached 71.67%. Nevertheless, the mainstream UAV hyperspectral cameras operate on push-broom imaging principles, compromising spatial resolution in favor of spectral information, which leads to low image acquisition efficiency and strict demands on stable environmental illumination. This limitation hampers their suitability for outdoor environments and large-area detection. Conversely, the advancement of deep learning technology has notably enhanced the performance of hyperspectral image reconstruction networks. A viable approach involves utilizing UAV RGB cameras to capture color remote sensing images and conduct hyperspectral image reconstruction, enabling the extraction of spectral information for disease testing purposes.

Currently, hyperspectral images reconstruction networks can accurately retrieve spectral information from RGB images. The accuracy of hyperspectral reconstruction has been consistently enhanced in recent computer vision summit challenges, including NTIRE2018 (Arad et al., 2018), NTIRE2020 (Arad et al., 2020), and NTIRE2022 (Arad et al., 2022). Among them, the NTIRE2018 champion network HSCNN+ is a convolutional neural network-based method for hyperspectral image reconstruction (Shi et al., 2018). In the NTIRE2020 champion network HRNET, a 4-level hierarchical regression network method is used to reconstruct hyperspectral images (Zhao et al., 2020b). The network utilizes PixelUnShuffle and Pixelshuffle layers for inter-layer downsampling and upsampling (Shi et al., 2016), thereby preserving more spectral details. In NTIRE2022, the champion network MST++ employed the Transformer (Ashish et al., 2017) as the fundamental framework for hyperspectral image reconstruction (Cai et al., 2022). To enhance the performance of spectral reconstruction, particularly for reconstructing UAV remote sensing hyperspectral images in large-scale forest monitoring, this study introduces a novel spectral reconstruction network called DW3D. DW3D employs depth-separable three-dimensional convolution to minimize the network parameter count (Rahim et al., 2021) and utilizes a U-shaped network architecture to extract features at various scales (Ronneberger et al., 2015), resulting in improved reconstruction performance. This network architecture exhibits strong expressive and generalization abilities in spectral reconstruction tasks. This article employs the three champion reconstruction networks and the self-built DW3D network architecture to conduct spectral reconstruction on RGB images captured by UAV remote sensing, specifically for the task of early PWD detection.

Therefore, we presented a novel method for early detection of PWD, specifically based on UAV remote sensing hyperspectral image reconstruction and SVM classification. The contributions of this article can be summarized as follows:

1. This study combined the hyperspectral image reconstruction method with an SVM classification model to achieve Early stage detection of PWD using UAV RGB remote sensing images.
2. This paper proposed a novel hyperspectral reconstruction network that combines three-dimensional depth-separable convolution and a U-shaped network to enhance detection efficiency and enable real-time detection of UAV remote sensing over large areas.
3. Establish a UAV hyperspectral PWD image dataset containing healthy pine trees and pine trees in the Early stages of PWD.

2 Materials

This paper utilized two datasets: a self-built dataset for detecting PWD and a public dataset for training the hyperspectral image reconstruction network. Furthermore, by combining needle analysis, ground plant observations, and drone images, PWD is categorized into five stages: (I) Green pine, (II) Early stage, (III) Middle stage, (IV) Heavy stage, and (V) Gray stage (Yu, R et al., 2021). The remote sensing images provided by this article are also divided into 5 categories, as depicted in Figure 1.

2.1 The self-built dataset of pine forest hyperspectral images

The pine remote sensing dataset utilized in this experiment was sourced from a forest farm located in Hecheng Street, Gaoming District, Foshan City, Guangdong Province, China(22°54′N, 112°51′E). The forest farm spans 57 hectares and predominantly consists of masson pine trees. The dataset comprises UAV remote sensing images captured at the same location over a duration of eight months, from May to December 2022. Data collection was performed using the DJI Phantom 4RTK drone, with each flight carried out at an altitude of 350 meters. Data collection took place in sunny weather conditions between 10:00 and 14:00, with a light intensity of approximately 100,000 Lux, to maintain data collection consistency. The RGB camera had a spatial resolution of 1600x1300 pixels, as depicted in Figure 2.

This study conducted 10 remote sensing data collections spanning from May to December 2022. The study utilized the coordinates of mid to late-infected pine trees in more recent images to identify early-infected pine trees in earlier images. Additionally, ground inspection confirmed that the target plants were infected with pine wilt disease (PWD) As shown in Figure 3.

Following manual annotation of the original remote sensing images, the UAV remote sensing images were cropped to achieve a resolution of 512x482. As shown in Figure 4A. The cropped images were processed by the hyperspectral reconstruction network, resulting in the creation of the corresponding dataset of pine forest hyperspectral images. The spectral information of the corresponding plant is extracted from the reconstructed hyperspectral image based on the annotation. As shown in Figure 5.

A total of 320 target plants, consisting of equal proportions of healthy and diseased plants, were obtained based on the annotations. Eventually, a spectrum dataset consisting of 320 samples were obtained, with each sample containing 31 spectral features and a label indicating the presence or absence of disease. A total of 31 spectral features are utilized for machine learning classification.

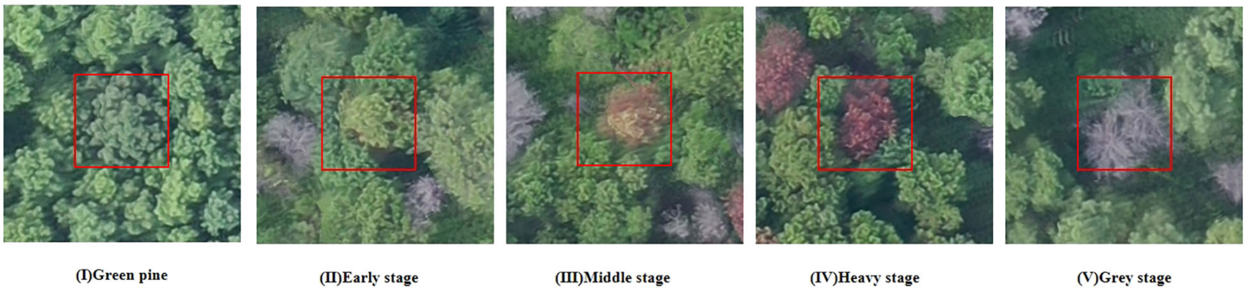


FIGURE 1
Unmanned Aerial Vehicle (UAV) images of pine trees at different PWD infection stages.

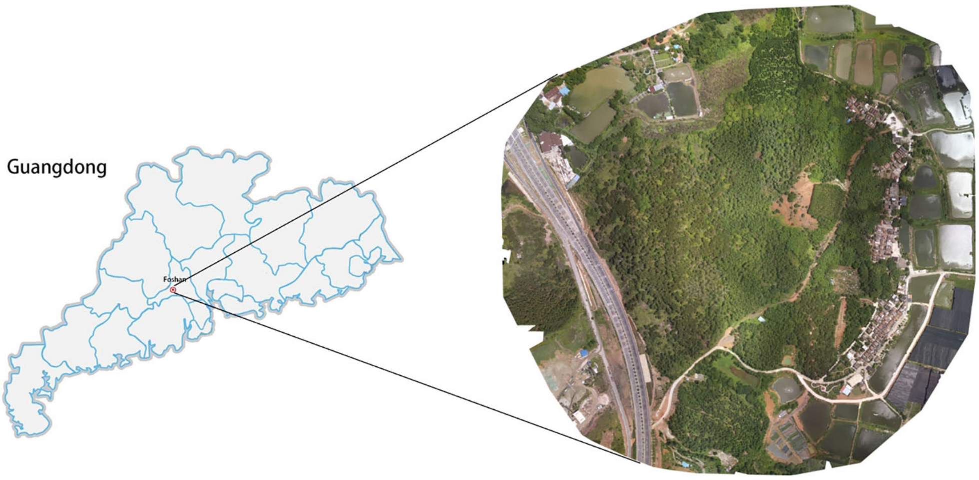


FIGURE 2
A stitched image of the study area.

2.2 The public dataset

The NTIRE2022 spectral reconstruction public dataset (Arad et al., 2022) is employed in this article to pre-train the four reconstruction networks utilized in our study. The dataset comprises 1000 RGB-HSI pairs, which are partitioned into training, validation, and test subsets in an 18:1:1 ratio. Each HSI data possesses a spatial resolution of 482 × 512 and encompasses 31 wavelengths ranging from 400 nm to 700 nm, as depicted in Figure 4B.

3 Methods

Deep networks are employed in this article for the reconstruction of hyperspectral images. The champion networks, HSCNN+, HRNet, and MST++ from the NTIRE2018, NTIRE2020, and NTIRE2022 Spectral Reconstruction Challenges, were utilized. Additionally, the self-built spectral reconstruction network DW3D from this article was employed to reconstruct hyperspectral images of pine forest remote sensing.

Simultaneously, this study employed the SVM classification model to classify and detect Early stage of PWD using the reflectance spectral features of target plants obtained through crown segmentation from the pine forest reconstructed hyperspectral images.

3.1 Hyperspectral reconstruction network

3.1.1 DW3D self-built network

DW3D is a hyperspectral reconstruction network that combines three-dimensional convolution and the U-Net architecture. Depthwise separable three-dimensional convolution is employed to reduce the parameter count, contributing to a more efficient network. It achieves improved reconstruction performance by

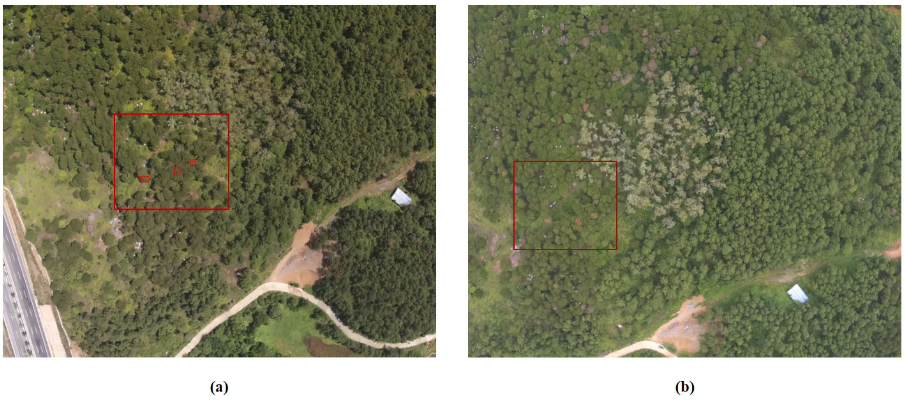


FIGURE 3
(A) Correspond to the images captured on May 26 2022, and **(B)** correspond to the images captured on June 29 2022. Mid to late-infected pine trees in the newer image identifying early-infected pine trees in the older image by coordinates.

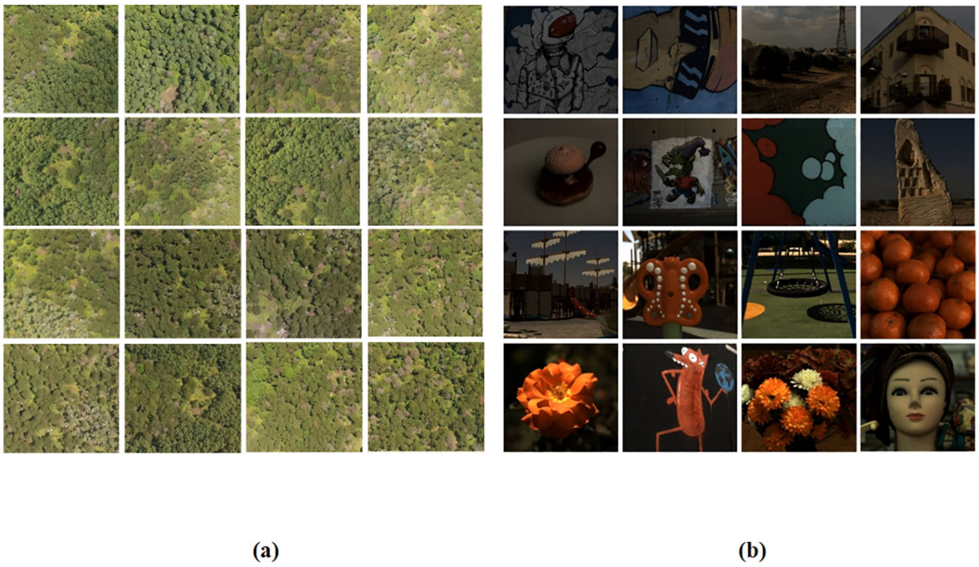


FIGURE 4
(A) UAV RGB remote sensing image dataset, and **(B)** the NTIRE2022 Spectral Reconstruction Challenge open source dataset.

performing multi-scale feature extraction through a U-shaped network architecture.

Figure 6 illustrates that the input of the DW3D network is an RGB image. Initially, the feature channels of the RGB input undergo expansion using a separate two-dimensional convolution layer, followed by a matrix dimension transformation to include a depth dimension. Subsequently, a deep feature extraction module is constructed using three encoder modules and three decoder modules, forming a combination of U-shaped networks. To alleviate the issues of information loss and blurring during network transmission, a skip connection is employed, allowing direct transfer of intermediate features from the encoder module to the decoder module. Residual connections are incorporated in the deep feature extraction module to address the challenges of gradient disappearance and gradient explosion. The encoder module comprises two residual separable 3D convolution blocks (RS3CB), and the decoder module includes two depth-separable 3D convolutions (D3C) (Rahim et al., 2021). To enhance convergence speed and improve the model's generalization ability, batch normalization and the PReLU activation function are applied. RS3CB is composed of two depthwise separable 3D convolutions and a PReLU activation function. Batch normalization is applied prior to the activation function, and residual connections are utilized within RS3CB. Downsampling in the U-shaped network is performed using a three-dimensional maximum pooling layer, whereas upsampling is achieved using a three-dimensional transposed convolution layer. In contrast to conventional upsampling approaches like nearest neighbor interpolation and bilinear interpolation, the three-dimensional transposed convolution layer is equipped with learnable parameters (Dumoulin and Visin, 2016) and can dynamically adjust the upsampling parameters based on the network. Lastly, the output feature cube from the deep feature extraction module is transformed into a feature cube with 31 channels using a three-dimensional convolution operation. Subsequently, a hyperspectral image comprising 31 channels is obtained by reshaping the matrix dimensions.

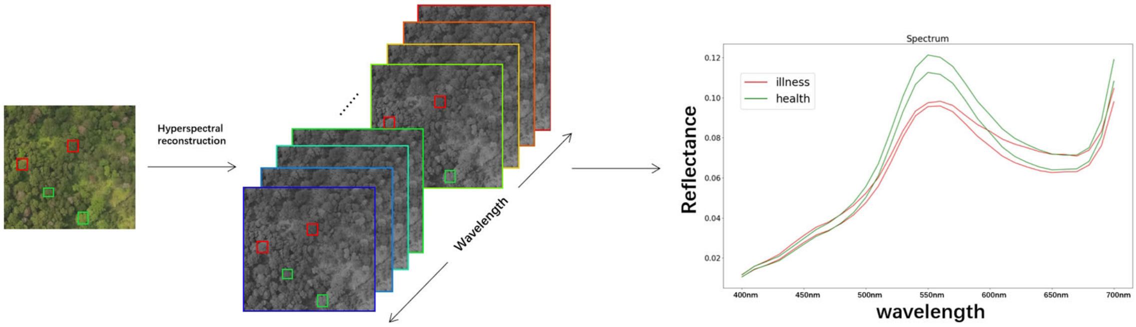


FIGURE 5
Hyperspectral images reconstruction and reflectance spectral feature acquisition.(The red box represents the diseased plant, the green box represents the healthy plant, and the curve depicts the spectral characteristic curve of the plant.).

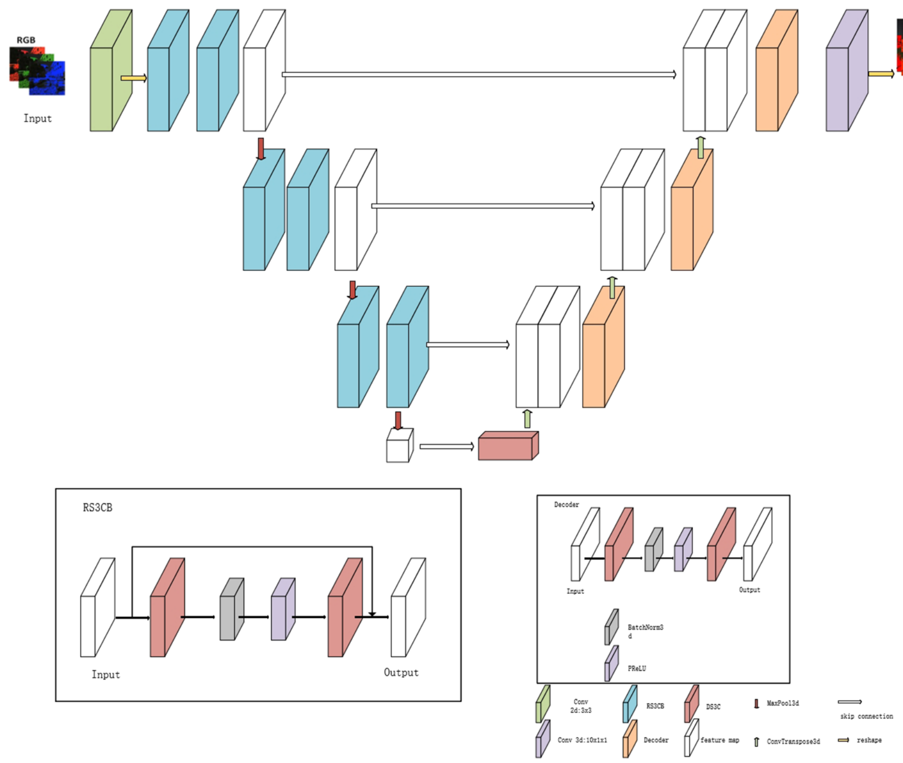


FIGURE 6
Self-built DW3D structure.

In this study, Figure 7 illustrates the employment of the Depthwise Separable 3D Convolution Module (D3C). Given that 3D convolution possesses an additional dimension compared to 2D convolution, it offers advantages in channel dimensionality. Concurrently, the U-shaped network architecture, known for its efficacy in capturing features across varied scales of image space dimensions, is adopted. Traditional 3D convolution bases its convolution kernel count on input and output channel numbers, leading to potential parameter redundancy. To address this, the study suggests utilizing Depthwise Separable 3D convolution to trim unnecessary parameters while maintaining the reconstruction quality. This technique segments the convolution layer into two components: a grouped 3D convolution and a pointwise 3D convolution. Initially, kernels with matching input channels extract diverse channel features, followed by one-dimensional kernels for channel information fusion. This approach balances feature extraction and channel information fusion, effectively reducing parameters without compromising reconstruction outcomes.

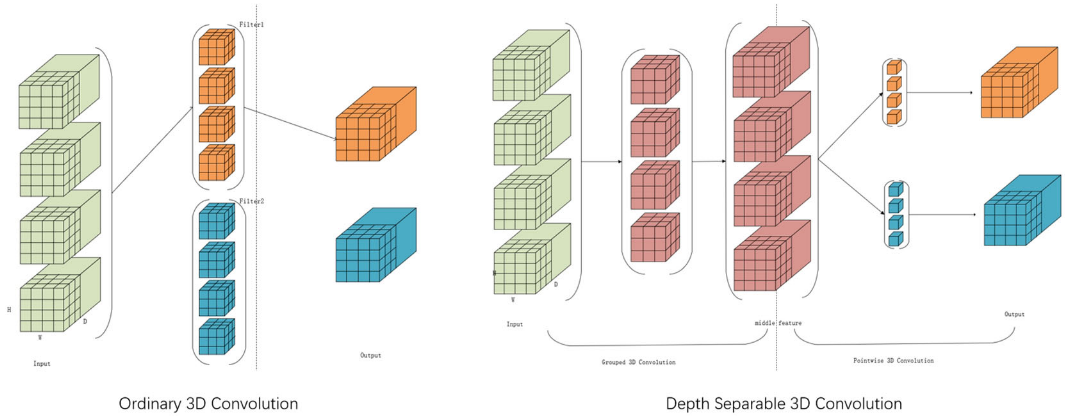


FIGURE 7
Depthwise Separable 3D Convolutional Module (D3C).

3.1.2 Hyperspectral reconstruction network evaluation

To assess the performance of DW3D proposed in this article on the NTIRE2022 dataset in an objective manner, the scoring criteria provided by the NTIRE2022 Spectral Reconstruction Challenge are followed. The evaluation of this article employs RMSE(root mean square error), MRAE(mean relative absolute error), and PSNR (Peak signal-to-noise ratio) as indicators. MRAE is selected as the ranking criterion to prevent the introduction of weighting errors in high-brightness areas of the test image, rather than using RMSE. The calculation of MRAE, RMSE and PSNR is as follows:

$$\mathrm{MRAE} = \frac{1}{N}\sum_{p=1}^{N}\left(\frac{\left|I_{HSI}^{(p)} - I_{SR}^{(p)}\right|}{I_{HSI}^{(p)}}\right) \tag{1}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{p=1}^{N}\left(I_{HSI}^{(p)} - I_{SR}^{(p)}\right)^2} \tag{2}$$

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right) \tag{3}$$

$$\mathrm{MSE} = \frac{1}{N}\sum_{p=1}^{N}\left(I_{HSI}^{(p)} - I_{SR}^{(p)}\right)^2 \tag{4}$$

where $I_{HSI}^{(p)}$ and $I_{SR}^{(p)}$ denote the p-th pixel value of the ground truth and the spectral reconstructed HSI. MAX represents the maximum value among all image points. A smaller MRAE or RMSE indicate better performance.

3.2 Machine learning classification model

3.2.1 Classification algorithm

This article employs the support vector machine (SVM) as the classification algorithm (Cortes et al., 1995). It utilizes the soft margin SVM and employs various high-dimensional space mapping methods to transform the spectral data into linearly separable data in high-dimensional space.

Furthermore, this article employs logistic regression. It utilizes decision tree classification and Bayesian classification for detection and compares them with SVM classification results. Based on the outcomes in section 4.2, the study adopts the optimal SVM classification model.

3.2.2 Evaluation

This article utilized the 10-fold cross-validation accuracy (Cross-Validation Accuracy) as the ranking metric to assess the classification results. Additionally, precision, recall, and accuracy were employed to evaluate the detection performance, as shown below.

$$\mathrm{recall} = \frac{TP}{TP + FN} \tag{5}$$

$$\mathrm{precision} = \frac{TP}{TP + FP} \tag{6}$$

$$\mathrm{accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

Here, TP(A True Positive) occurs when a positive example is correctly classified as positive, while FN(a False Negative) is recorded when a positive example is erroneously classified as negative. Similarly, TN(a True Negative) is noted when a negative example is correctly classified as negative, and FP(a False Positive) is registered when a negative example is mistakenly classified as positive.

This article expanded the evaluation of classification problems by incorporating ten-fold cross-validation to assess the accuracy and stability of the machine learning model. Ten-fold cross-validation involves the random division of the sample data into 10 parts, with 9 parts selected as the training set in each iteration, and the remaining 1 part used as the test set. After each round, 9 new samples are randomly chosen for training. Following several rounds (fewer than 10), the model and parameters that yield optimal loss function evaluation are selected. This method provides a more comprehensive reflection of the model's stability and accuracy.

4 Results

4.1 Training of hyperspectral image reconstruction network

This article compared a self-built hyperspectral reconstruction network with an existing state-of-the-art (SOTA) hyperspectral reconstruction network, including the champion networks HSCNN+ and HRNET from the NTIRE2018 and NTIRE2020 Spectral Reconstruction Challenge, as well as the NTIRE2022's champion network MST++. The results were presented in Table 1. The self-built DW3D reconstruction yields improved results in terms of reconstruction accuracy compared to HSCNN + and HRNet. However, DW3D exhibited a noticeable gap when compared to MST++. Our self-constructed DW3D network exhibits the lowest number of parameters and computational load. This demonstrates the superior efficiency of our DW3D network. GFLOPS, an acronym for floating-point operations, serves as a metric to assess the computational speed of a model. Typically, a lower GFLOPS value correlates with higher computational speed.

This article presented visual HSI reconstruction results and their corresponding error maps for various methods. Figure 8 displays the error heat map for band 21. Darker colors indicate smaller errors, while lighter colors indicate larger errors. The MRAE error is uniformly distributed across the range from 0 to 10, depicted on the far right of the figure. Based on these figures, it was evident that MST++ exhibits the most favorable reconstruction effect and fidelity, with the DW3D method following closely.

TABLE 1 Comparison of four hyperspectral reconstruction network evaluation indicators.

Methods	Parameter (M)	FLOPS n(G)	MRAE	RMSE	PSNR
HSCNN+	4.65	302.89	0.3814	0.0588	26.36
HRNet	31.70	164.20	0.3476	0.0550	26.89
MST++	1.62	23.10	0.1645	0.0248	34.32
DW3D	0.61	16.42	0.3177	0.0446	28.40

4.2 PWD predict results

This study performed conducts a comparative analysis of several different hyperspectral image reconstruction networks combined with SVM classification models for the detection of Early stage PWD. HSCNN+, HRNet, MST++, and the self-built DW3D network were used for hyperspectral image reconstruction and combined with different kernel function SVM classification models, logistic regression classification models, decision tree classification models, and Bayesian classifiers to conduct binary classification experiments. For the classification experiment, the samples underwent shuffling, and a random number seed divided the training and test sets into a 7:3 ratio. The experimental results are shown in Table 2.

Based on the experimental results, the SVM classification model outperforms the other three classification models. We studied the best matching value of the kernel function coefficient (gamma) and the regularization parameter C in the Radial Basis Function (RBF) SVM for PWD data classification in the paper. The gamma value determines the coefficient of the kernel function. The greater the gamma value, the kernel function has a greater impact on the classification results. The experiments demonstrated that the classification cross-validation accuracy was optimal when the kernel function coefficient gamma was set to 0.1 and C was set to 10.

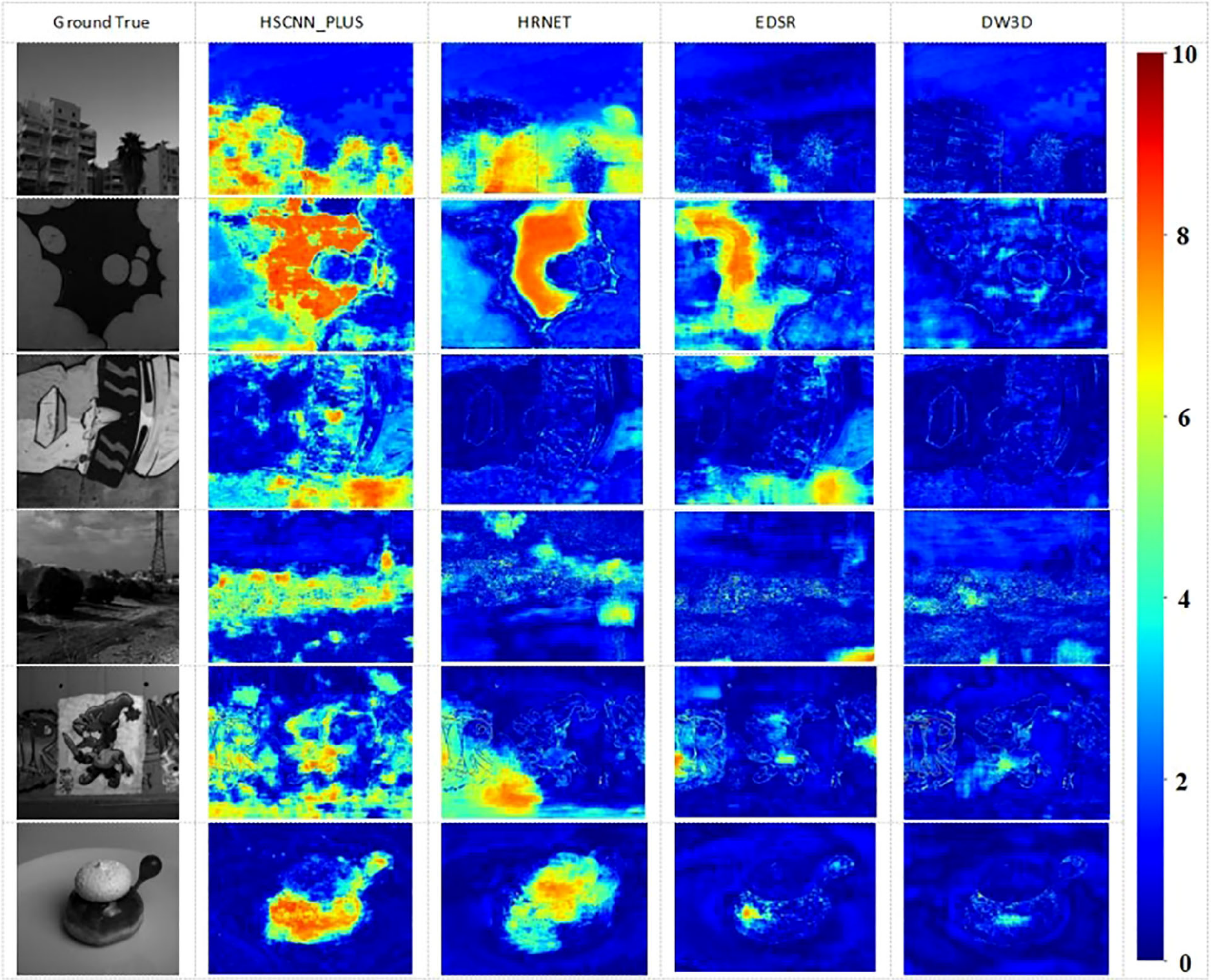


FIGURE 8
The Visual results of the 21-th band and the reconstruction error images of an HSI chosen from validation set.

This paper investigated the influence of the degree of the polynomial kernel function and the constant term on PWD data classification in polynomial SVM. Modifying the values of degree and constant term alters the model's capacity to capture nonlinear features. The optimal degree and coef0 are determined by evaluating the cross-validation accuracy. Additionally, the study examined the influence of different regularization parameters (L1 and L2) and various loss functions in the linear kernel function SVM on PWD data classification. Finally, this article employed grid search to identify the optimal parameters by exploring all potential combinations among the candidate parameters. The parameters yielding the best cross-validation accuracy were then selected.

This study ranks using the average accuracy obtained from ten-fold cross-validation and assesses the performance of the classification model by considering standard deviation, accuracy, precision, and recall indicators. Table 2 reveals the utilization of the self-constructed DW3D network for reconstruction. The poly kernel SVM cross-verification accuracy stands at 74.0%, with the linear kernel SVM achieving the highest accuracy of 82.47%. The MST++ network exhibited a cross-validation accuracy of 77% using the poly kernel SVM, with the highest accuracy of 83.5% achieved using the linear kernel SVM. The results revealed that the self-built hyperspectral reconstruction network outperforms HSCNN+ and HRNet in terms of detection accuracy, generalization, and application performance. It is noteworthy that we achieved a cross-validation accuracy of 74% using our self-built DW3D approach, with the number of model parameters amounting to only 38% of MST++.

In Figure 9, This study utilized the MST++-polySVM model to present the visualization results of selected Early stage PWD samples from the test set (the subsequent result analysis defaults to this model). Each sample in the study is assigned a unique number during manual annotation. The annotations of the test set samples are then selected based on their respective test numbers for visual representation. The prediction results are manually annotated on the image based on the plants' location information,

TABLE 2 Predict results of different reconstruction networks and different kernel SVMs.

Reconstructed Methods	Predicted model	Cross validation	Standard Deviation	Accuracy	Precision	Recall
MST++	Poly-SVM	0.77	0.0744	0.824	0.77	0.82
	Linear-SVM	0.704	0.0774	0.835	0.87	0.80
	Rbf-SVM	0.707	0.0902	0.763	0.77	0.72
	Logistic Regression	0.70	0.0923	0.824	0.71	0.78
	Decision Tree	0.67	0.0855	0.730	0.72	0.80
	Bayes Classifier	0.65	0.0896	0.710	0.70	0.62
DW3D	Poly-SVM	0.74	0.0971	0.793	0.82	0.71
	Linear-SVM	0.669	0.1058	0.8247	0.84	0.66
	Rbf-SVM	0.66	0.0965	0.773	0.86	0.64
	Logistic Regression	0.69	0.1208	0.783	0.78	0.71
	Decision Tree	0.68	0.1032	0.762	0.69	0.73
	Bayes Classifier	0.63	0.1158	0.65	0.71	0.64
HRNet	Poly-SVM	0.719	0.0722	0.763	0.79	0.70
	Linear-SVM	0.70	0.0834	0.793	0.77	0.70
	Rbf-SVM	0.66	0.1003	0.773	0.74	0.68
	Logistic Regression	0.704	0.0856	0.763	0.79	0.70
	Decision Tree	0.64	0.0961	0.793	0.73	0.72
	Bayes Classifier	0.65	0.1143	0.723	0.70	0.68
HSCNN+	Poly-SVM	0.70	0.0658	0.814	0.89	0.70
	Linear-SVM	0.70	0.0657	0.79	0.86	0.77
	Rbf-SVM	0.641	0.1148	0.773	0.78	0.77
	Logistic Regression	0.70	0.1189	0.804	0.83	0.71
	Decision Tree	0.67	0.1139	0.78	0.85	0.77
	Bayes Classifier	0.65	0.1135	0.742	0.62	0.73

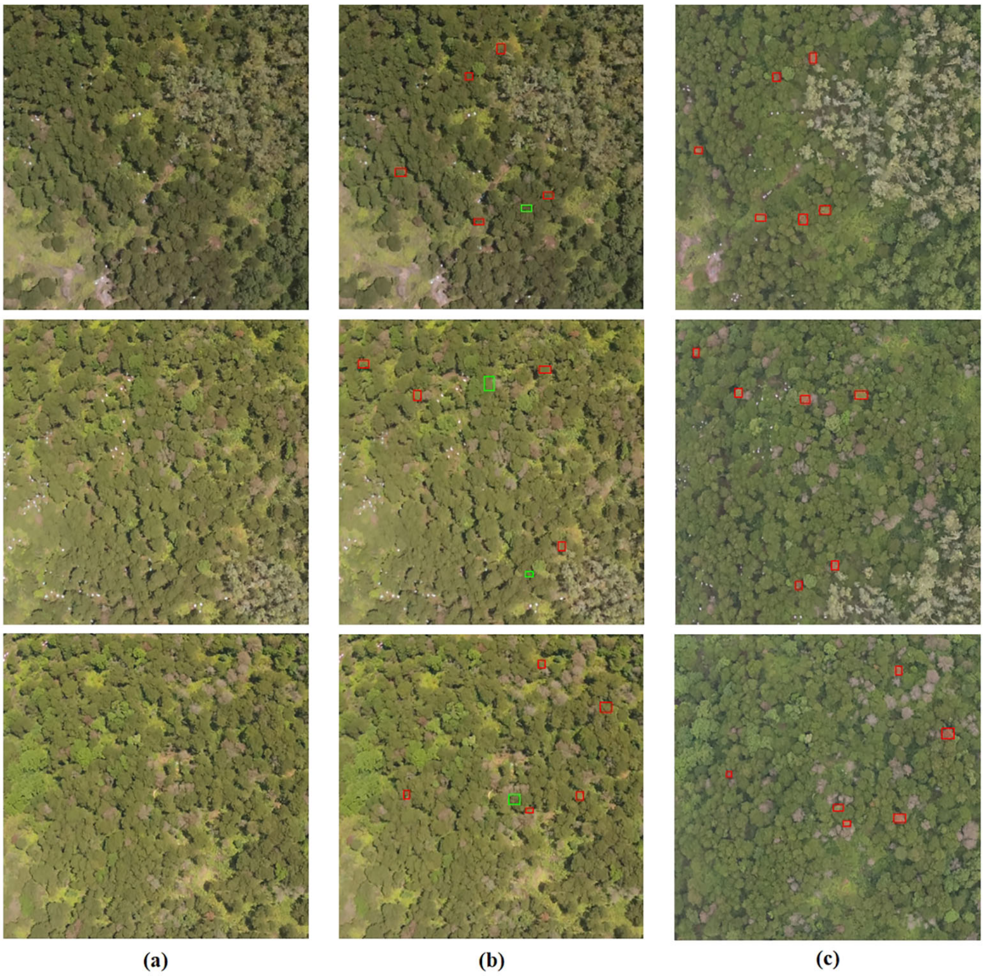


FIGURE 9
The visualization results of Early stage PWD samples from the test set are presented, with successful classifications denoted in red boxes and incorrect classifications denoted in green boxes. **(A)** represents the original image, while **(B)** represents the predicted result. **(A, B)** Correspond to the images captured on May 26, while **(C)** represents the real image taken on June 29.

corresponding to the test sample number. Correct classifications are indicated by a red box, while incorrect ones are marked with a green box. In the early PWD detection samples randomly selected, our model can achieve 77% detection accuracy.

This paper currently employs the SVM model to achieve a 77% classification accuracy on a dataset comprising 320 samples, despite having shown promising results in early PWD detection studies. However, there exists ample room for in-depth analysis and enhancement within this study's findings. Firstly, the quantity and quality of samples play a crucial role. The high flight altitude of the drone in this study results in limited pixels per tree canopy, potentially impacting hyperspectral image reconstruction. Moreover, due to the sample size constraint, the SVM's performance was not fully realized, potentially influencing the model's efficacy. Secondly, the selection of model parameters significantly affects accuracy. Experimentation with various kernel functions and parameter combinations has been conducted to optimize detection outcomes. Lastly, feature engineering stands out as a pivotal aspect for enhancing model performance. Future research aims to delve deeper into feature engineering techniques, including feature combination and dimensionality reduction methods, to enhance the model's data representation capabilities.

5 Discussion and conclusion

5.1 Advantages of DW3D networks

The self-constructed DW3D model in this paper exhibits substantially fewer parameters and floating-point operations—accounting for only 38% and 71% of MST++, respectively. This indicates that the DW3D model introduced in this study offers accelerated image processing. This makes it feasible to integrate our lightweight reconstruction model into embedded devices, unlike existing reconstruction algorithms that are impeded by their high parameter count. By consuming fewer memory resources and operating with greater speed, our DW3D model enables seamless integration with embedded devices. Moreover, these embedded devices can be incorporated into drone platforms, facilitating rapid, non-destructive, and large-scale detection of early pine wilt disease.

5.2 RGB images are directly used for early PWD detection

This paper compared the utilization of RGB images directly, without employing a hyperspectral reconstruction network. The results are presented in Table 3. The highest achieved cross-validation accuracy is only 0.50, which means that the use of hyperspectral images is more conducive to early PWD detection.

This study employs a decision tree classification model to assess feature importance. The model evaluates the significance of features by examining their impact on the target variable. Notably, in constructing the decision tree, the optimal partitioning attribute is chosen based on the feature's classification efficacy on the dataset, utilizing metrics like information gain or information gain ratio. This paper examines and contrasts the contributions of reconstructed hyperspectral images and RGB images in label classification. Figure 10 displays the ranking of feature importance for labels using 31 features of the reconstructed hyperspectral image and 3 features of the RGB image. The 610nm band exhibits the highest feature importance, succeeded by the 570nm and 630nm bands, as depicted in the figure. The approximate ranges of the three channels in the RGB image are as follows: R (around 700 nm), G (around 550 nm), and B (around 430 nm). The feature importance ranking of hyperspectral images reveals that the features near these channels are of lesser significance. Simultaneously, the feature ranking of RGB images indicates that the importance of the R, G, and B channels is relatively moderate. These three features appear to contain limited classification information, resulting in suboptimal classification performance when using RGB images.

This article employed a three-dimensional display diagram to compare and analyze the visualization results of three features (R, G, and B) comprising 80 random samples and the channel values (570nm, 610nm, and 630nm) that exert the highest influence on the label. Figure 11 reveals that the labels in the RGB visualization results lack a distinct classification tendency, whereas the visualization results of the 18th, 22nd, and 24th channels in HSI exhibit a noticeable classification tendency. These findings imply that incorporating features from reconstructed hyperspectral images may enhance the classification performance in this task.

Some research methodologies integrate RGB images with deep learning target detection algorithms for direct detection (Yu, R et al., 2021b; Wu et al., 2021);. Yu, R et al. utilized Faster R-CNN, YOLOv4, random forest, and support vector machine algorithms to detect early PWD. The target detection algorithm categorized PWD into four classes, with early PWD accuracy at 48.88%. Machine learning was employed for multi-classification tasks of PWD, achieving an overall classification accuracy of 75.33%. Wu, B et al. trained four deep learning models (Faster R-CNN ResNet50, Faster R-CNN ResNet101, YOLOv3 DarkNet53, and YOLOv3 MobileNet) for detection. Images ranging from 390 to 760nm were utilized for early PWD detection, with an average detection accuracy of 50.2%. While the target detection algorithm based on deep learning in RGB images is effective for mid- and late-stage PWD, its efficacy for early-stage PWD is moderate. This is primarily due to minimal changes in canopy color of early PWD diseased plants, leading to challenges in accurately identifying them using the deep learning-based target detection algorithm. In conclusion, reconstructing the spectral data from RGB images and extracting spectral information prove to be valuable and beneficial for the early detection of PWD.

TABLE 3 Comparison of classification results using RGB color images directly.

Methods	Cross validation	Accuracy
MST++_SVM	0.77	0.824
DW3D_SVM	0.74	0.793
HRNet_SVM	0.719	0.763
HSCNN+_SVM	0.70	0.79
RGB_SVM	0.504	0.531

5.3 The robustness of the model

This study randomly selected 87 samples in May, June, and August respectively as test sets to study the test results in different months. The confusion matrix of the test results was shown in Figure 12. Specifically, the samples in May were used as the test set, and the precision of the disease was 0.674, and the recall was 0.659; the samples in June were used as the test set, the precision of the disease was 0.68, and the recall was 0.84; Lastly, the samples in August were used as the test set, the precision of the disease was 0.67, recall was 0.77. The slight variation might stem from the sample discrepancies across various months, consequently influencing the classification outcomes. Moreover, the chi-square test conducted on 10 test results in this study revealed no significant statistical variance. Consequently, the hyperspectral image reconstruction network presented in this study, in conjunction with the SVM-based Early Stage PWD detection model, demonstrates robustness across samples from various months.

5.4 Conclusion

To summarize, In this paper, we proposed a method that utilizes UAV remote sensing color images, deep learning-based hyperspectral image reconstruction networks, and SVM classification to achieve early detection of pine wilt disease (PWD) in pine forests. The method achieved an accuracy of 77%. The key to achieving early detection of PWD in this study was the utilization of hyperspectral image reconstruction networks to reconstruct hyperspectral images The accuracy of the early detection method proposed in this study was comparable to or surpasses that of methods utilizing drone hyperspectral remote sensing technology. This article introduces a new hyperspectral image reconstruction network called DW3D, with the number of parameters being only 38% of MST++, the computational load is merely 71% of that in MST++, while achieving a relatively high-quality reconstruction. Hyperspectral remote sensing necessitates costly sensors for accurate data collection and entails time-

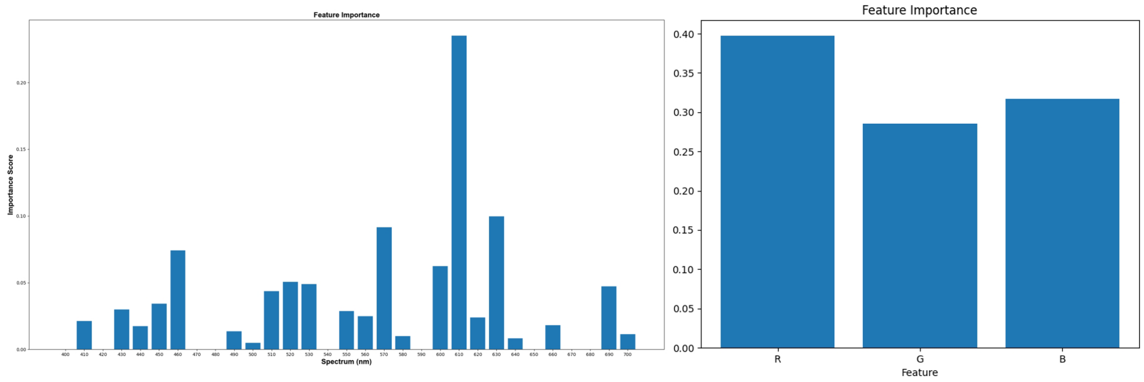
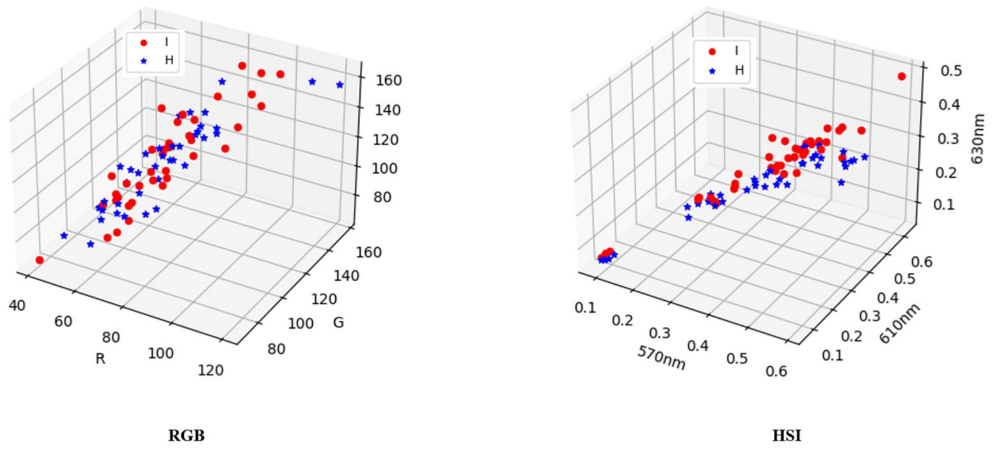


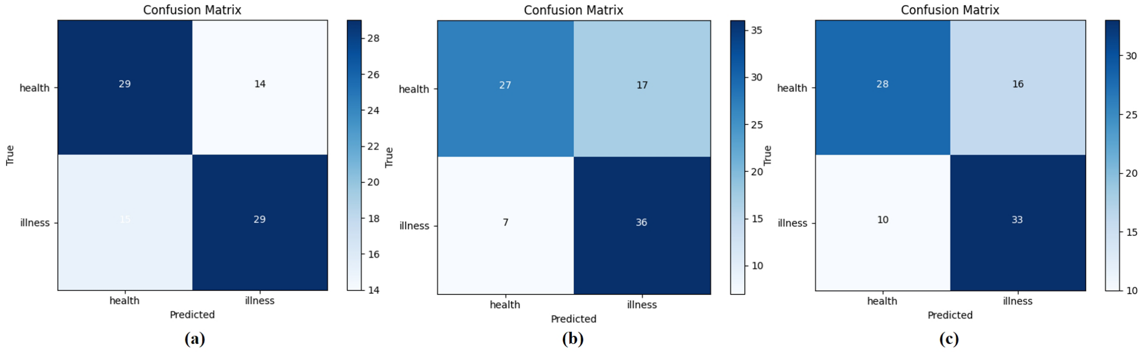
FIGURE 10
Feature importance of 31 channel features of hyperspectral image (400-700nm, spectral resolution of 10nm) and 3 features of RGB image (R, G, B) to labels.



RGB

HSI

FIGURE 11
A three-dimensional display of 80 random samples by selecting the three features R, G, and B, and a three-dimensional display of 80 samples by selecting 570nm, 610nm, and 630nm, which have the highest impact on the label.



(a)

(b)

(c)

FIGURE 12
Samples from May, June, and August are selected as test sets respectively. **(A–C)** Respectively represent the confusion matrix of the test set on May 26th, the test set on June 8th, and the test set on August 17th.

consuming and complex data processing. In contrast, the method proposed in this paper reduces the detection cost, enhances data collection efficiency, and offers more advantages for large-area detection tasks. The establishment of the pine wilt disease (PWD) remote sensing dataset in this study serves as a foundation for early detection of PWD. As the dataset size grows in the future, the detection accuracy will also improve.

Current research encounters challenges in large-scale detection. The primary challenge involves integrating classification processing into reconstruction to establish an end-to-end detection model. The secondary challenge pertains to the limited richness of current data samples. Future endeavors should focus on enhancing early PWD samples to boost the detection model's accuracy, facilitating more effective early PWD detection models in extensive agricultural regions.

Data availability statement

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

Author contributions

WL: Software, Writing – original draft. ZX: Investigation, Project administration, Writing – review & editing. JD: Formal analysis, Visualization, Writing – review & editing. YHL: Validation, Writing – review & editing. YBLo: Funding acquisition, Project administration, Resources, Writing – review & editing. YBLa: Conceptualization, Resources, Writing – review & editing. TL: Project administration, Resources, Writing – review & editing. SS: Resources, Writing – review & editing. JZ: Investigation, Methodology, Software, Supervision, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by Forest Science and Technology Innovation Project of Guangdong province (No.2023KJCX018), Science and Technology Plan Project of Guangzhou (2023B03J1392), and Specific University Discipline Construction Project of Guangdong Province (2023B10564002).

Acknowledgments

The authors were also grateful to many of colleagues for stimulating discussion in this field.

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

Arad, B., Liu, D., Wu, F., Lanaras, C., Galliani, S., Schindler, K., et al. (2018). "NTIRE 2018 challenge on spectral reconstruction from RGB images," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (New York, USA:: Ieee). doi: 10.1109/cvprw.2018.00138

Arad, B., Timofte, R., Ben-Shahar, O., Lin, Y.-T., and Finlayson, G. (2020). NTIRE 2020 Challenge on Spectral Reconstruction From an RGB Image., in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (Ieee), 446–447. Available online at: https://objects.scraper.bibcitation.com/user-pdfs/2024-06-15/d7c7ee61-901c-4010-8bce-33f0a3816375.pdf (Accessed June 15, 2024)

Arad, B., Radu, T., Rony, Y., Nimrod, M., Amir, B., Cai, Y., et al. (2022). "NTIRE 2022 spectral recovery challenge and data set," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (New York, USA: Ieee). Available online at: https://objects.scraper.bibcitation.com/user-pdfs/2024-06-15/2cf5eb67-1823-44c1-a0de-0fea91ab6cd4.pdf.

Ashish, V., Noam, S., Niki, P., Jakob, U., Llion, J., Aidan, G., et al. (2017). "Attention is All you Need," in *Advances in neural information processing systems(NIPS).* (Cambridge, Massachusetts, USA: MIT Press). Available online at: https://objects.scraper.bibcitation.com/user-pdfs/2024-06-15/b6696026-087a-49a8-a4f0-05bad918858d.pdf.

Bagheri, N. (2020). Application of aerial remote sensing technology for detection of fire blight infected pear trees. *Comput. Electron. Agric.* 168, 105147. doi: 10.1016/j.compag.2019.105147

Cai, Y., Lin, J., Lin, Z., Wang, H., Zhang, Y., Pfister, H., et al. (2022). "MST++: multi-stage spectral-wise transformer for efficient spectral reconstruction," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (New York, USA: Ieee). doi: 10.1109/cvprw56347.2022.00090

Du, Z., Wu, S., Wen, Q., Zheng, X., Lin, S., and Wu, D. (2024). Pine wilt disease detection algorithm based on improved YOLOv5. *Front. Plant Sci.* 15. doi: 10.3389/fpls.2024.1302361

Dumoulin, V., and Visin, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv.org.* Available online at: https://arxiv.org/abs/1603.07285.

Futai, K. (2013). Pine wood nematode,Bursaphelenchus xylophilus. *Annu. Rev. Phytopathol.* 51, 61–83. doi: 10.1146/annurev-phyto-081211-172910

Guo, Y., Xiao, Y., Hao, F., Zhang, X., Chen, J., de Beurs, K., et al. (2023). Comparison of different machine learning algorithms for predicting maize grain yield using UAV-based hyperspectral images. *Int. J. Appl. Earth Observation Geoinformation* 124, 103528. doi: 10.1016/j.jag.2023.103528

Hao, Z., Huang, J., Li, X., Sun, H., and Fang, G. (2022). A multi-point aggregation trend of the outbreak of pine wilt disease in China over the past 20 years. *For. Ecol. Manage.* 505, 119890. doi: 10.1016/j.foreco.2021.119890

Ikegami, M., and Jenkins, T. A. R. (2018). Estimate global risks of a forest disease under current and future climates using species distribution model and simple thermal model – Pine Wilt disease as a model case. *For. Ecol. Manage.* 409, 343–352. doi: 10.1016/j.foreco.2017.11.005

Kim, S.-R., Lee, W.-K., Lim, C.-H., Kim, M., Kafatos, M., Lee, S.-H., et al. (2018). Hyperspectral analysis of pine wilt disease to determine an optimal detection index. *Forests* 9, 115. doi: 10.3390/f9030115

Li, J., Wang, X., Zhao, H., Hu, X., and Zhong, Y. (2022). Detecting pine wilt disease at the pixel level from high spatial and spectral resolution UAV-borne imagery in complex forest landscapes using deep one-class classification. *Int. J. Appl. Earth Observation Geoinformation* 112, 102947. doi: 10.1016/j.jag.2022.102947

Oide, A. H., Nagasaka, Y., and Tanaka, K. (2022). Performance of machine learning algorithms for detecting pine wilt disease infection using visible color imagery by UAV remote sensing. *Remote Sens. Applications: Soc. Environ.* 28, 100869. doi: 10.1016/j.rsase.2022.100869

Rahim, R., Shamsafar, F., and Zell, A. (2021). "Separable convolutions for optimizing 3D stereo networks," in *2021 IEEE International Conference on Image Processing (ICIP)* New York, USA*: (Ieee)*. doi: 10.1109/icip42928.2021.9506330

Rao, D., Zhang, D., Lu, H., Yang, Y., Qiu, Y., Ding, M., et al. (2023). Deep learning combined with Balance Mixup for the detection of pine wilt disease using multispectral imagery. *Comput. Electron. Agric.* 208, 107778. doi: 10.1016/j.compag.2023.107778

Sangaiah, A. K., Yu, F.-N., Lin, Y.-B., Shen, W.-C., and Sharma, A. (2024). UAV T-YOLO-rice: an enhanced tiny yolo networks for rice leaves diseases detection in paddy agronomy. *IEEE Trans. Network Sci. Eng.*, 1–16. doi: 10.1109/tnse.2024.3350640

Shan, C., Wang, G., Wang, H., Wu, L., Song, C., Hussain, M., et al. (2024). Assessing the efficiency of UAV for pesticide application in disease management of peanut crop. *Pest Manage. Sci.* 80, 4505–4515. doi: 10.1002/ps.8155

Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A. P., Bishop, R., et al. (2016). "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (New York, USA: Ieee). doi: 10.1109/cvpr.2016.207

Shi, Z., Chen, C., Xiong, Z., Liu, D., and Wu, F. (2018). "HSCNN+: advanced CNN-based hyperspectral recovery from RGB images," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (New York, USA: Ieee). doi: 10.1109/cvprw.2018.00139

Wu, B., Liang, A., Zhang, H., Zhu, T., Zou, Z., Yang, D., et al. (2021). Application of conventional UAV-based high-throughput object detection to the early diagnosis of pine wilt disease by deep learning. *For. Ecol. Manage.* 486, 118986. doi: 10.1016/j.foreco.2021.118986

Xiao, D., Pan, Y., Feng, J., Yin, J., Liu, Y., and He, L. (2022). Remote sensing detection algorithm for apple fire blight based on UAV multispectral image. *Comput. Electron. Agric.* 199, 107137. doi: 10.1016/j.compag.2022.107137

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

Yu, R., Luo, Y., Zhou, Q., Zhang, X., Wu, D., and Ren, L. (2021a). A machine learning algorithm to detect pine wilt disease using UAV-based hyperspectral imagery and LiDAR data at the tree level. *Int. J. Appl. Earth Observation Geoinformation* 101, 102363. doi: 10.1016/j.jag.2021.102363

Yu, R., Luo, Y., Zhou, Q., Zhang, X., Wu, D., and Ren, L. (2021b). Early detection of pine wilt disease using deep learning algorithms and UAV-based multispectral imagery. *For. Ecol. Manage.* 497, 119493. doi: 10.1016/j.foreco.2021.119493

Yu, R., Ren, L., and Luo, Y. (2021c). Early detection of pine wilt disease in Pinus tabuliformis in North China using a field portable spectrometer and UAV-based hyperspectral imagery. *For. Ecosyst.* 8, 44. doi: 10.1186/s40663-021-00328-6

Zeng, T., Diao, C., and Lu, D. (2021). U-net-based multispectral image generation from an RGB image. *IEEE Access* 9, 43387–43396. doi: 10.1109/access.2021.3066472

Zhao, J., Huang, J., Yan, J., and Fang, G. (2020a). Economic loss of pine wood nematode disease in mainland China from 1998 to 2017. *Forests* 11, 1042. doi: 10.3390/f11101042

Zhao, Y., Po, L.-M., Yan, Q., Liu, W., and Lin, T. (2020b). "Hierarchical regression network for spectral reconstruction from RGB images," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (New York, USA: Ieee). doi: 10.1109/cvprw50498.2020.00219

Zhao, G., Zhang, Y., Lan, Y., Deng, J., Zhang, Q., Zhang, Z., et al. (2023). Application progress of UAV-LARS in identification of crop diseases and pests. *Agronomy* 13, 2232. doi: 10.3390/agronomy13092232

frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 06 December 2024
DOI 10.3389/fpls.2024.1475282

OPEN ACCESS

EDITED BY
Chunlei Xia,
Chinese Academy of Sciences (CAS), China

REVIEWED BY
Yalin Wu,
Lushan Botanical Garden (CAS), China
Nisha Pillai,
Mississippi State University, United States

*CORRESPONDENCE
Linnan Yang
1985008@ynau.edu.cn

RECEIVED 03 August 2024
ACCEPTED 20 November 2024
PUBLISHED 06 December 2024

CITATION
Bai C, Zhang L, Gao L, Peng L, Li P and Yang L (2024) DINOV2-FCS: a model for fruit leaf disease classification and severity prediction. *Front. Plant Sci.* 15:1475282. doi: 10.3389/fpls.2024.1475282

DINOV2-FCS: a model for fruit leaf disease classification and severity prediction

Chunhui Bai[1,2,3], Lilian Zhang[1,2,3], Lutao Gao[1,2,3], Lin Peng[1,2,3], Peishan Li[1,2,3] and Linnan Yang[1,2,3]*

[1]College of Big Data, Yunnan Agricultural University, Kunming, China, [2]Yunnan Engineering Technology Research Center of Agricultural Big Data, Kunming, China, [3]Yunnan Engineering Research Center for Big Data Intelligent Information Processing of Green Agricultural Products, Kunming, China

Introduction: The assessment of the severity of fruit disease is crucial for the optimization of fruit production. By quantifying the percentage of leaf disease, an effective approach to determining the severity of the disease is available. However, the current prediction of disease degree by machine learning methods still faces challenges, including suboptimal accuracy and limited generalizability.

Methods: In light of the growing application of large model technology across a range of fields, this study draws upon the DINOV2 visual large vision model backbone network to construct the DINOV2-Fruit Leaf Classification and Segmentation Model (DINOV2-FCS), a model designed for the classification and severity prediction of diverse fruit leaf diseases. DINOV2-FCS employs the DINOv2-B (distilled) backbone feature extraction network to enhance the extraction of features from fruit disease leaf images. In fruit leaf disease classification, for the problem that leaf spots of different diseases have great similarity, we have proposed Class-Patch Feature Fusion Module (C-PFFM), which integrates the local detailed feature information of the spots and the global feature information of the class markers. For the problem that the model ignores the fine spots in the segmentation process, we propose Explicit Feature Fusion Architecture (EFFA) and Alterable Kernel Atrous Spatial Pyramid Pooling (AKASPP), which improve the segmentation effect of the model.

Results: To verify the accuracy and generalizability of the model, two sets of experiments were conducted. First, the labeled leaf disease dataset of five fruits was randomly divided. The trained model exhibited an accuracy of 99.67% in disease classification, an mIoU of 90.29%, and an accuracy of 95.68% in disease severity classification. In the generalizability experiment, four disease data sets were used for training and one for testing. The mIoU of the trained model reached 83.95%, and the accuracy of disease severity grading was 95.24%.

Discussion: The results demonstrate that the model exhibits superior performance compared to other state-of-the-art models and that the model has strong generalization capabilities. This study provides a new method for leaf disease classification and leaf disease severity prediction for a variety of fruits. Code is available at https://github.com/BaiChunhui2001/DINOV2-FCS.

KEYWORDS

DINOV2, deep learning, fruit disease recognition, semantic segmentation, smart agriculture

1 Introduction

In the contemporary globalized food supply chain, fruits occupy a pivotal position in the human diet. Fresh fruits, in particular, are highly esteemed for their alluring aroma and distinctive flavor (Wang et al., 2022). Fruit diseases represent a significant challenge for the fruit industry, accounting for significant economic losses annually. Timely identification of fruit diseases helps control infections and ensure optimal productivity (Khan et al., 2022). However, traditional fruit disease detection methods are susceptible to subjective judgement and experience differences of the inspector, leading to inconsistent and low accuracy of detection results (Khattak et al., 2021). Deep learning-based fruit disease detection methods not only significantly increase detection speed and accuracy, but also further optimise and enhance the ability of disease identification through continuous data accumulation and learning (Shoaib et al., 2023).

The development and implementation of autonomous plant disease detection has been made easier by the ongoing advancements in artificial intelligence technologies. A study (Atila et al., 2021) employed the EfficientNet model to identify diseases of plant leaves, with the objective of enhancing diagnostic accuracy and efficiency. By contrasting it with advanced convolutional neural network models, the study demonstrated that EfficientNet performs well in classifying plant leaf images, thereby validating its potential for automated diagnosis of plant diseases. The RIC-Net (Zhao et al., 2022) was developed on the foundation of the Inception and residual structure fusion models, with an enhanced Convolutional Block Attention Module (CBAM) integrated for the purpose of enhancing the efficacy of plant leaf disease classification. The DFN-PSAN (Dai et al., 2024) model demonstrated high performance in identifying diseases of plants through the application of weather data augmentation techniques on three datasets derived from real agricultural scenarios. The topic of plant disease identification has already reached a mature state of application for deep learning techniques.

Precisely determining the extent of plant diseases is vital from the standpoint of application. This is because the detection of disease severity assists farmers in making informed decisions to mitigate losses due to disease infection. A study (Zeng et al., 2020) created a HLB-infected citrus leaf image dataset, expanded the original training dataset with a deep convolutional generative adversarial network, and trained six different deep learning models to perform severity detection. A unique three-branch Swin Transformer classification network (TSTC) was designed in another study (Yang et al., 2023)to diagnose plant diseases and their severity independently and concurrently. However, these plant disease severity estimates are based on simple classification networks, which are less effective and weakly interpretable. In practice, calculating the percentage of leaf diseased area is a crucial step in assessing the severity of the disease (Madden et al., 2007). A study (Goncalves et al., 2021) trained six semantic segmentation models for the purpose of recognizing and estimating the severity of plant leaf diseases with an accuracy comparable to that of commercial software. This was achieved without the need to manually adjust the segmentation parameters or remove complex backgrounds from the images. Another study (Hu et al., 2021) employed a support vector machine to segment the lesion in order to better identify the disease and offered an elliptical restoration approach to fit and restore the whole size of the occluded or damaged tea leaves. Researchers presented a deep learning and fuzzy logic based approach to establish an automated technique for grapevine black measles disease identification and severity analysis (Ji and Wu, 2022). To address the problem of cucumber downy mildew, researchers proposed a two-stage segmentation framework to calculate the percentage of leaf disease area (Wang et al., 2021). The resulting accuracy of the disease severity classification was 92.85%. Nevertheless, all of these works have trained models just for a single plant disease, thus leading to limited generalization.

As computer vision technology advances, large vision models find extensive use in several domains. SAM (Kirillov et al., 2023), a powerful model designed for segmentation tasks, has been developed to achieve zero-sample migration to a variety of tasks through cueing engineering. It has demonstrated excellent performance on a range of image segmentation tasks, which has contributed to the advancement of the computer vision field. However, the considerable computational expense of SAM represents a significant obstacle to its broader deployment in industrial settings. FastSAM (Zhao et al., 2023), MobileSAM (Zhang et al., 2023a), and MobileSAMv2 (Zhang et al., 2023b) employ model parameter reduction and accelerate inference techniques to mitigate this challenge. DINO (Caron et al., 2021) employs a novel contrast learning method to enhance its visual generic representation. This method compares the features of the original image with those of a randomly cropped image, resulting in highly satisfactory outcomes. DINOv2 (Oquab et al., 2023) is a method for pre-training an image encoder on a large image dataset in order to obtain visual features with semantic meaning. These features can be employed for a diverse range of visual tasks without the necessity for further training to achieve performance levels comparable to those of supervised models. In the application of large vision models, MedSAM (Ma et al., 2024) was demonstrated to have significantly enhanced segmentation performance on medical images by fine-tuning SAM. SAMRS (Wang et al., 2024) dataset developed using SAM and existing remote sensing datasets. The powerful feature extraction capability of large vision models can better assist agricultural disease detection. Nevertheless, there hasn't been any information on the use of large vision models in plant disease detection, particularly for classification and severity estimate.

In this study, we constructed the model DINOV2-FCS for leaf disease classification and severity prediction of a variety of fruits based on the DINOV2 large vision model backbone network. The contributions of this study are as follows:

1. We constructed the model DINOV2-FCS for leaf disease classification and severity prediction of a variety of fruits based on the DINOV2 large vision model backbone network. This approach has been shown to have good generalization ability.
2. In order to enhance the training of the model, the leaf and lesion regions in the 2010 images were meticulously labeled.
3. An improvement to the MLP decoder has been proposed, namely Explicit Feature Fusion Architecture (EFFA), which fuses explicit feature information and multilevel feature information and improves the segmentation accuracy of the model.
4. We have proposed Alterable Kernel Atrous Spatial Pyramid Pooling (AKASPP), which fuses contextual and detailed edge information from different sensory fields in order to enhance adaptability to varying sizes and shapes of lesion targets and to align with the edge details of leaves and lesions.
5. We have proposed Class-Patch Feature Fusion Module (C-PFFM), which fuses local detailed feature information from patch tokens and global feature information from class token, resulting in improved classification accuracy of the model.

2 Materials and methods

2.1 Datasets

This study collected 2,010 images related to five different fruit foliar diseases: apple black rot, cedar apple rust, grape black measles, grape black rot, and strawberry leaf scorch. These images were obtained from the public PlantVillage dataset (Hughes and Salathé, 2015), which consists of images captured in an indoor laboratory setting and is widely used for crop and plant disease research. We increased the number of images to 8,040 using data augmentation techniques, and all images were accurately labeled. The precise number of images for each disease is presented in Table 1. The procedure for processing the dataset was as follows:

1. Uniform image size: The selected images were resized to 512×512 pixels, consistent with the input specifications of the model, by using the resize method of the Image class in the Pillow library (version 10.2.0).
2. Data labeling: The leaf and lesion areas in the images were manually labeled with high accuracy using LabelMe (version 3.16.7). Each image was categorized into three regions: background, leaf, and lesion, represented by black, green, and red, respectively. The labeled images serve as a benchmark for evaluating the accuracy of the segmentation model. Figure 1A shows a selection of images from the dataset, alongside their accurately labeled counterparts.
3. Data augmentation: To simulate various lighting conditions and disturbances, data augmentation was applied to the original images by introducing random noise, applying blurring operations, and adjusting brightness. Specifically, NumPy (version 1.24.4) was used to generate Gaussian-distributed noise, which was added to the images. Various blurring algorithms from the OpenCV library (version 4.9.0.80) were applied, and brightness was randomly adjusted using a factor generated by NumPy. This enhanced the diversity of the dataset. Figure 1B shows examples of the augmented images.
4. Data splitting: To train the model and evaluate its performance, the dataset was randomly divided into

TABLE 1 Statistics on the number of datasets.

	Apple black rot	Cedar apple rust	Grape black measles	Grape black rot	Strawberry leaf scorch
Original	441	417	419	404	329
Enhanced	1323	1251	1257	1212	987
Total	1764	1668	1676	1616	1316

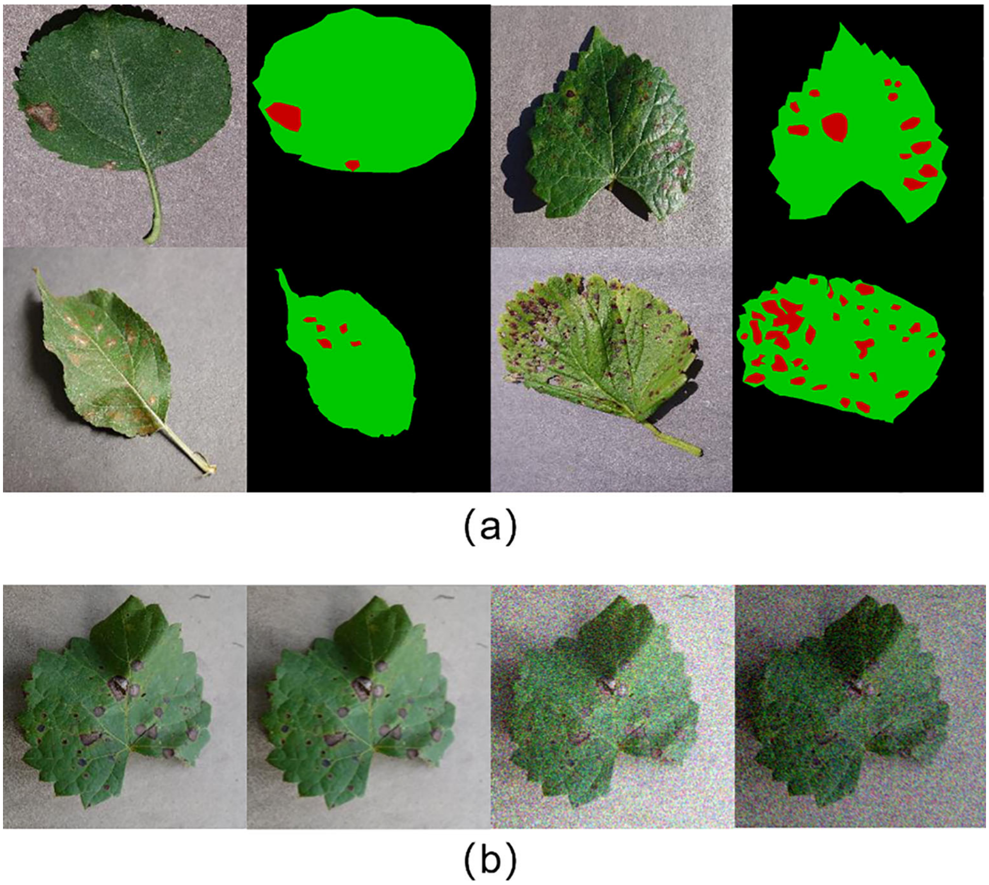


FIGURE 1
(A) Sample dataset annotation; **(B)** Sample data augmentation.

training and test sets with a 7:3 ratio. To ensure reproducibility, the random seed was set to 0.

In practice, calculating the percentage of leaf diseased area is a crucial step in assessing the severity of the disease. Nevertheless, there is as yet no uniform grading scale for the severity of disease. Guided by the experience of experts as well as references to the literature (Wang et al., 2021), this study graded the severity of leaf disease to facilitate a better assessment of model performance. illustrates the grading strategies employed to assess the severity of leaf disease. Table 2 illustrates the grading strategies employed to determine leaf disease severity.

TABLE 2 Grading strategies for the severity of leaf disease.

Disease grade	Proportion of disease spots in leaves P
Level 0	0
Level 1	0<P ≤ 10%
Level 2	10%<P ≤ 20%
Level 3	20%<P ≤ 40%
Level 4	40%<P ≤ 60%
Level 5	60%<P ≤ 100%

2.2 Model structure

In this study, a model, DINOV2-FCS, is constructed based on the DINOV2 large vision model for the purpose of classifying and segmenting diseased leaves of fruits. The DINOv2 model generates generalized visual features through pre-training on a large amount of well-curated data, which are effective across different image distributions and tasks without the need for fine-tuning. The DINOv2-FCS model uses the DINOv2-B (distilled) as the backbone. The DINOv2-B model adopts the ViT-B/14 architecture and consists of 12 consecutive Transformer Blocks. In this study, the classification and segmentation modules are designed separately to accomplish fruit leaf disease classification and severity prediction, respectively, using the features obtained from the backbone.

In the classification module, this study proposes Class-Patch Feature Fusion Module (C-PFFM) as a method of fusing patch tokens and class token for effective feature fusion. C-PFFM is demonstrated to more effectively utilise the features generated by the backbone for disease classification of fruit leaves, and to enhance the model's classification accuracy. In the segmentation module, the following methods are proposed: Explicit Feature Fusion Architecture (EFFA) and Alterable Kernel Atrous Spatial Pyramid Pooling (AKASPP). EFFA fuses explicit feature information and multilevel feature information. AKASPP fuses

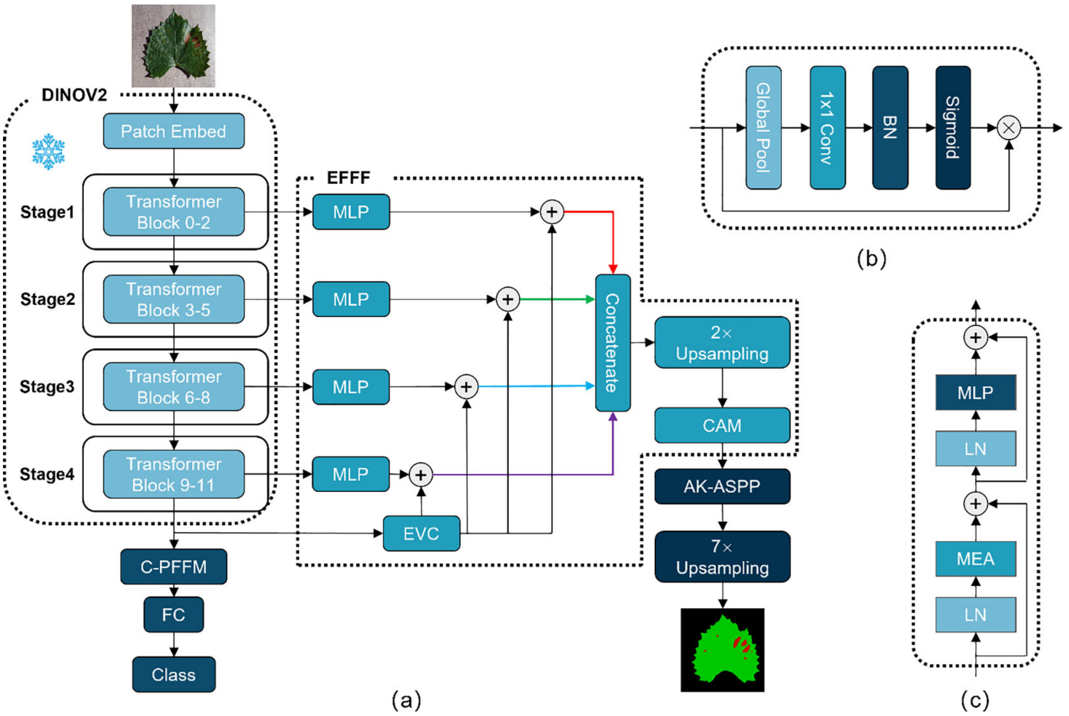


FIGURE 2
(A) Represents the overall structure of DINOV2-FCS; **(B)** represents the structure of CAM; **(C)** represents the structure of Transformer Block.

contextual information and detailed edge information from different sensory fields. These modules greatly enhance the segmentation performance. The overall model structure is shown in Figure 2.

2.3 Class-patch feature fusion module

In VIT (Dosovitskiy et al., 2020), the classifier typically inputs the class token to a fully connected layer, after which the classification result is obtained. The advantage of this approach is that the classifier is constructed in a straightforward manner, the number of parameters is minimal. However, utilising the class token as the sole input to the classifier will result in the omission of a substantial quantity of local, detailed feature information. To address this issue, Class-Patch Feature Fusion Module (C-PFFM) is proposed in this study. C-PFFM effectively fuses the local detail feature information of patch tokens and the global feature information of class token, thereby enhancing the model's classification accuracy. The operation procedure of C-PFFM is illustrated in Equation 1.

$$\begin{cases} \boldsymbol{H} = (\boldsymbol{1} - \boldsymbol{\alpha}) \cdot \boldsymbol{avgpoolX_p} + \boldsymbol{\alpha} \cdot \boldsymbol{X_c} \\ \boldsymbol{\alpha} = \boldsymbol{CBS}((\boldsymbol{avgpoolX_p} + \boldsymbol{X_c})) \end{cases} \tag{1}$$

X_p denotes patch tokens feature; X_c denotes class token feature; *avgpool* denotes global average pooling operation; *CBS* denotes Convolution + BN + Sigmoid; X denotes output feature map;

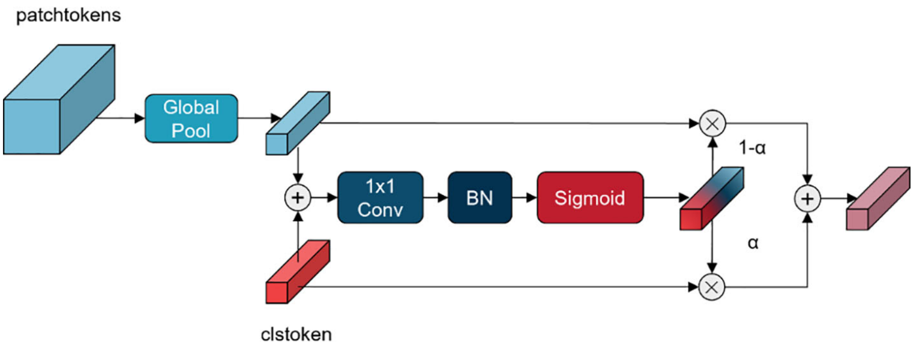


FIGURE 3
Structure of C-PFFM.

The final two layers of the backbone feature extraction network, patch tokens feature X_p and class token feature X_c, are initially identified. Feature W is obtained by performing a global average pooling operation on feature X_p and summing feature X_c element by element. The global average pooling operation is illustrated in Equation 2 The feature W is then subjected to convolution and BN operations to obtain the channel weights α via the Sigmoid operation. Feature X_p is subjected to element-by-element matrix dot-multiplication with the channel weights $(1 - \alpha)$ and the feature. The obtained features are subjected to element-by-element summing operation to obtain the patch tokens and class token fusion feature. The structure of C-PFFM is depicted in Figure 3.

$$\boldsymbol{X}_{avgpool} = \frac{\boldsymbol{1}}{\boldsymbol{H} \times \boldsymbol{W}} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{X}_{(i,j)} \tag{2}$$

X denotes the feature map; H denotes the height of the feature map; W denotes the width of the feature map; $X_{avgpool}$ denotes the feature after global average pooling.

Class token contains long-range global feature information and is often used as input features for classifiers. However, the rich local detailed feature information contained in patch tokens should not be ignored. In particular, in the task of classifying fruit leaf diseases, there is a great similarity between leaf spots of different diseases. If the detailed features are ignored and only the global features are focused on, it will lead to poor classification accuracy of the model. Local information typically encompasses fine structural and local features within an image, whereas global information encompasses the overall context and background knowledge. The effective fusion of the two enables the model to learn a complete and representative feature, thereby enhancing its ability to comprehend the input data and its classification performance.

2.4 Explicit feature fusion architecture

SegFormer (Xie et al., 2021) is a straightforward and effective semantic segmentation framework for Transformer. This approach avoids complex decoder design and fuses information from different layers. For semantic segmentation tasks, these feature information are multi-layered global feature information and lack explicit feature information, which makes it difficult to segment some tiny targets. CFPNet (Quan et al., 2023) proposes an Explicit Visual Center (EVC) that focuses on aggregating local corner-region features of an image to enhance the feature representation.In this study, Explicit Feature Fusion Architecture (EFFA) is proposed. The output features from each of the four stages of the DINOV2 backbone are input into the MLP layer to obtain global feature information at multiple levels. Subsequently, the features from the last layer of the DINOV2 backbone are inputted into the EVC to obtain explicit feature information. The explicit feature information is integrated into the global feature information of each layer through a summing operation with the global feature information of multiple layers. Finally, the multilevel feature information is spliced according to the channels and fused by a channel attention. The specific structure of EFFA is illustrated in Figure 2.

The image of leaf disease exhibits a multitude of spots of varying sizes. When the model performs segmentation, it is not uncommon that disease spots are incompletely segmented or subtle spots are directly ignored. EVC provides a powerful feature enhancement mechanism for the model. This mechanism enables semantic segmentation models to recognize and localize objects in an image with greater accuracy, particularly in the context of images comprising multiple segmented objects, such as those depicting leaf diseases. The EFFA proposed in this study fuses explicit feature information into global feature information at each level, subsequently fusing multilevel feature information. Multi-level fusion can exploit the complementarity between the underlying and higher-level features to enhance the feature representation. The lowest-level features typically comprise local details and texture information about the image, whereas the highest-level features encompass more abstract semantic information. These multilevel features integrate explicit feature information from EVC.

2.5 Alterable kernel atrous spatial pyramid pooling

In fruit leaf images, there are numerous spots with intricate shapes and varying sizes that can significantly impact the segmentation performance of the model. A Pyramid Pooling Module (PPM), comprising a set of pooling blocks with distinct scales, has been proposed in PSPNet (Zhao et al., 2017) based on the concept of pyramid pooling. The PPM provides a comprehensive global representation encompassing the interrelationships between diverse scales and subregions, thereby minimizing the loss of contextual information. DeepLabv2 (Chen et al., 2017a) proposed Atrous Spatial Pyramid Pooling (ASPP) to fuse multi-scale information. In light of this, DeepLabv3 (Chen et al., 2017b) and DeepLabv3+ (Chen et al., 2018) have enhanced the ASPP module, achieving notable outcomes. These modules employ diverse scales of receptive fields for fusion, addressing the issue of varying target sizes in images. However, in the context of fruit leaf disease images, the spot targets are also characterised by intricate shapes and indistinct edges. In this study, a novel approach, AKASPP, is proposed for the fusion of contextual and detailed edge information from different receptive fields. This approach is based on inflated convolution and AKConv (Zhang et al., 2023).

Expansion convolution offers the potential to provide a larger sensory field than conventional convolution. Conventional convolution permits the construction of a receptive field of size $K \times K$ when the convolution kernel size is K. In contrast, inflated convolution provides a receptive field as illustrated in Equation 3 Alterable Kernel Convolution (AKConv) is a new type of convolutional operation that allows convolution kernels to have an arbitrary number of parameters and an arbitrary sampling shape. In contrast to traditional convolution operations, which are typically constrained to fixed-size windows and fixed sample shapes, AKConv defines the initial position of an arbitrarily sized convolution kernel through a novel coordinate generation algorithm and introduces offsets to accommodate alterations in

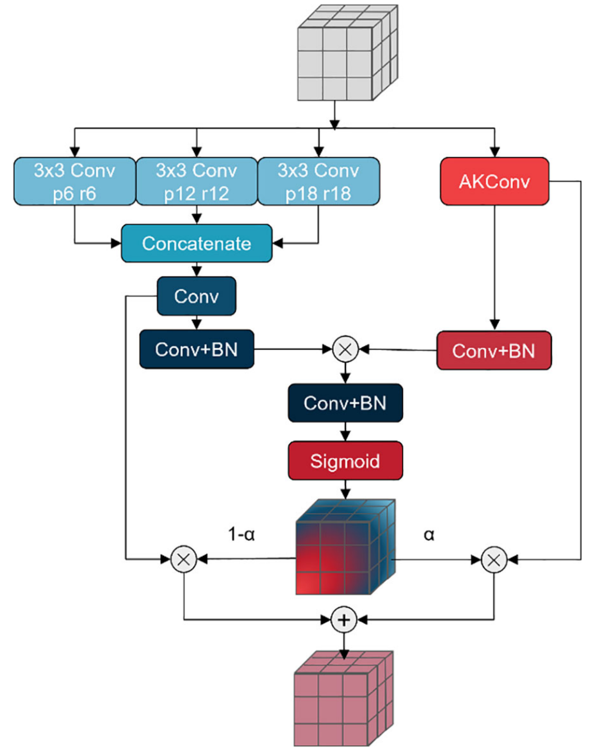


FIGURE 4
Structure of ASPP.

the target shape. In semantic segmentation tasks, AKConv can facilitate more precise local feature extraction and enhanced edge detail fitting, thereby enhancing the accuracy and detail of segmentation.

$$RF = ((r - 1)(K - 1) + K)^2 \tag{3}$$

RF denotes the receptive field of the convolution kernel;*r* denotes the expansion rate of the expansion convolution; *K* denotes the convolution kernel size;

In this study, AKASPP is proposed for fruit leaf disease images with complex spot shapes, blurred edges, and different sizes. Figure 4 illustrates the specific structure of AKASPP. AKASPP is capable of fusing contextual and detailed edge information from different receptive fields. In order to capture features under different receptive fields, expansion convolution with different expansion coefficients is employed. This enables the model to capture a sufficiently wide range of contextual information at different scales, thereby improving the recognition of targets of varying sizes. AKConv permits the convolutional kernel to have an arbitrary sampling shape, which differs from the traditional fixed square sampling shape. This flexibility allows the convolutional kernel to adapt more effectively to the varying shapes of spot targets, and to be sufficiently flexible to capture image features and fit the edge details of leaves and spots, thus improving performance. AKASPP effectively fuses this feature information to better segment different sizes and shapes of spot targets, and to better handle the edge portions of leaves and spots.

2.6 Loss functions

The cross-entropy loss function is used in this work as the loss function when the classification module is being trained. The cross-entropy loss function is shown in Equation 4. Figure 5A illustrates the variation of loss during the training of the classification model. The loss curve gradually becomes smooth after 5000 iterations.

$$L = -\frac{1}{N}\sum_{n=0}^{N-1} y\log(p) \tag{4}$$

L denotes the indicated cross-entropy loss; *y* denotes the true label of the pixel; *p* denotes the prediction result of the pixel; *N* denotes the number of difficult samples.

Unbalanced categories or a lack of challenging examples are common issues in semantic segmentation tasks, which can impair model performance. In the fruit leaf disease scene segmentation job, for instance, the disease spot category might only cover a minority of the space, but the leaf category might represent the majority. Insufficient performance in predicting other categories may result from the model's training primarily focusing on the leaf category. Online Hard Sample Mining (OHEM) can assist the model in

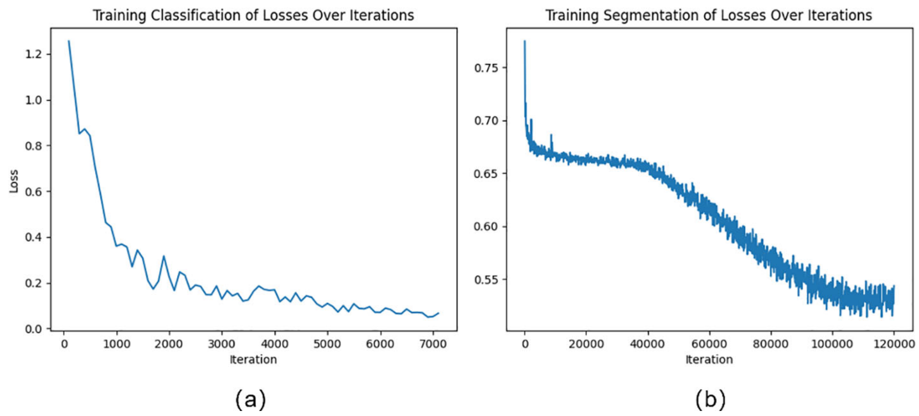


FIGURE 5
(A) Training classification of losses over iterations; **(B)** Training segmentation of losses over iterations.

focusing on difficult and rare samples, thereby improving overall performance (Shrivastava et al., 2016). In this study, the cross-entropy loss function of the semantic segmentation module includes OHEM. The loss function in this study is shown in Equations 5–7. Figure 5A illustrates the variation of loss during the training of the segmentation model. The loss curve gradually becomes smooth after 100000 iterations.

$$l_{CE} = -ylog(p) \tag{5}$$

$$l_{Hard} = l_{CE}, \ l_{CE} > 0.7 \tag{6}$$

$$L_{ohemCE} = \frac{1}{M}\sum_{m=0}^{M-1} l_{Hard} \tag{7}$$

l_{CE} denotes cross-entropy loss; y denotes the true label of the pixel; p denotes the prediction result of the pixel; l_{Hard} denotes the loss of difficult samples; L_{ohemCE} denotes the loss function in the OHEM combined with the cross-entropy loss function; M denotes the number of difficult samples.

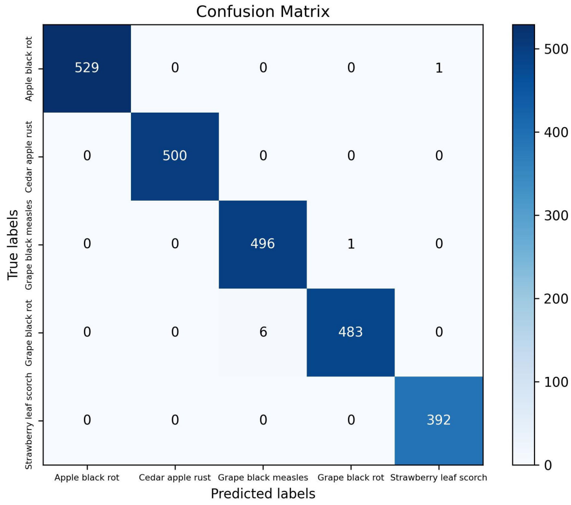


FIGURE 7
Confusion matrix for classification results.

3 Experimental results

3.1 Disease classification results

The classification module of the model proposed in this study achieved a ACC of 99.67% and a Macro F1 of 99.67% on the test set. Figure 6 presents the evaluation results of five distinct plant disease classification algorithms, including precision, recall, and F1 score. The diseases are presented from left to right in the following order: apple black rot, cedar apple rust, grape black measles, grape black rot, and strawberry leaf scorch. For each disease, the values of the three evaluation metrics are nearly identical, indicating that the model proposed in this study has high accuracy in recognizing these specific plant diseases. Figure 7 depicts a confusion matrix plot for the purpose of evaluating the performance of a classification model.

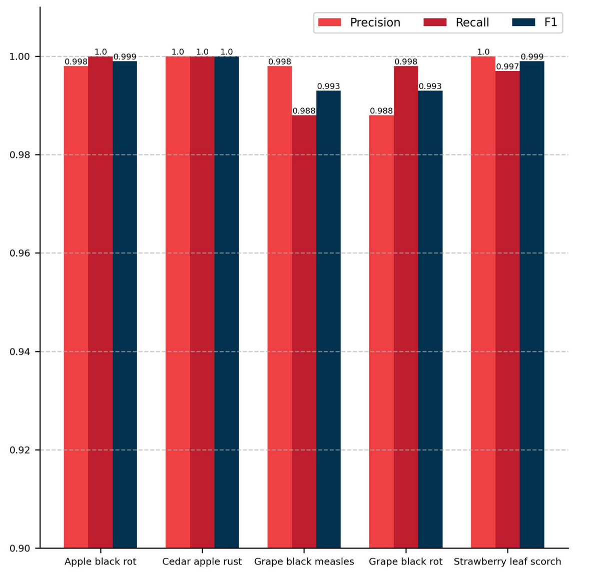


FIGURE 6
Histogram of classification results.

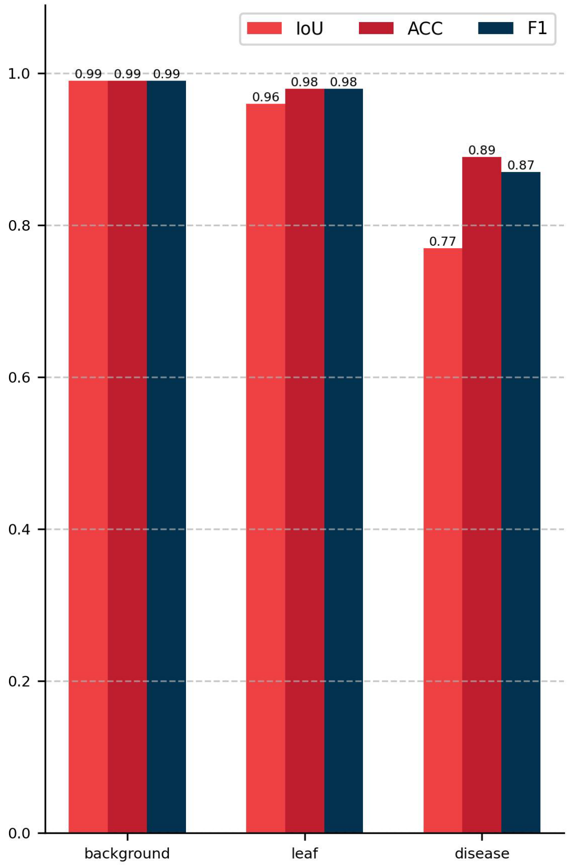


FIGURE 8
Histogram of semantic segmentation results.

The x-axis represents the predicted labels, the y-axis represents the true labels, the diagonal of the matrix represents the number of correct disease predictions, and the rest of the matrix represents misclassifications. As illustrated in the figure, the model exhibited a high degree of accuracy in classifying diseased leaves in the test set, correctly identifying the vast majority of samples. Only a small number of samples were misclassified. For instance, in the sample pertaining to apple black rot, there were 529 correctly classified samples, with only 1 misclassified as strawberry leaf scorch. Among the samples of grape black rot, 483 were correctly classified, while 6 were misclassified as grape black measles due to the high degree of similarity between the two grape diseases. Nevertheless, the model achieved satisfactory results. In conclusion, the DINOV2-FCS proposed in this study is an excellent tool for the classification of fruit leaf diseases.

3.2 Semantic segmentation results

The semantic segmentation module of the model proposed in this study achieved a mIoU of 90.29, a PA of 98.13%, and a Macro F1 of 94.61% on the test set. Figure 8 presents the outcomes of the evaluation of the semantic segmentation algorithm for three categories, including three evaluation metrics: IoU, PA, and F1. The IoU, PA, and F1 for the background category are 0.99, the leaf category is 0.96, 0.98, and 0.98, respectively, and the disease category is 0.77, 0.89, and 0.87, respectively. The data in Figure 8 indicates that the background category achieved the best evaluation results, the leaf category was the next best, and the disease category had the worst evaluation results. This phenomenon can be attributed to the fact that in images where the background and leaves tend to occupy the majority of pixels, the disease only occupies a small number of pixels. This results in a significant imbalance in the number of samples, which impedes the network's ability to learn sufficient information about the pixels in the disease category. As illustrated in Figure 9, the vast majority of pixels are correctly categorized, with only a small number of pixels not being correctly classified. The figure also demonstrates that the disease category has a relatively small number of pixels compared to the other categories. In conclusion, the DINOV2-FCS proposed in this study demonstrates satisfactory performance in the segmentation of leaf diseases.

TABLE 3 Classification performance of different models.

Model	ACC/%	Macro F1/%	Params
ResNet101	92.28	92.42	0.43×10^8
VIT(Base)	97.51	95.57	0.86×10^8
ConvNext(Base)	98.46	98.50	0.88×10^8
Swin(Base)	99.29	99.31	0.87×10^8
Ours	99.67	99.67	0.87×10^8

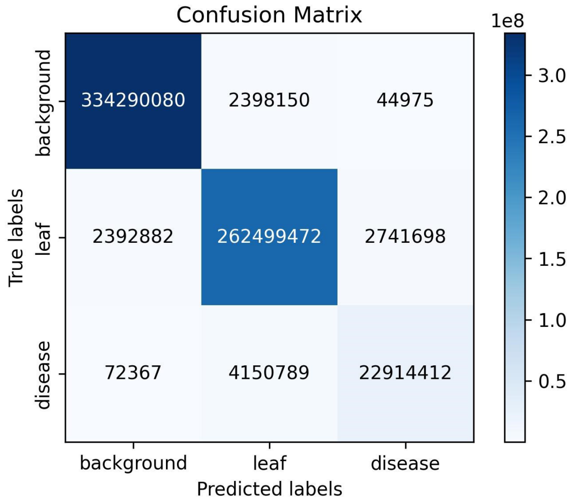


FIGURE 9
Confusion matrix of semantic segmentation results.

3.3 Results of leaf disease severity prediction

In this study, the fruit leaf disease severity was categorized into five classes. The model proposed in this work exhibited 95.68% accuracy in grading prediction on the test set. As illustrated in Figure 10, the model employed in this study demonstrated satisfactory performance in predicting the severity of fruit leaf disease. The proximity between the ratio of diseased spot area to total leaf area predicted by the model and the true label was high, with a difference of less than 0.40% observed even in individual samples where the prediction grading was erroneous. Consequently, the model in this study exhibited satisfactory capacity for the measurement of fruit leaf disease severity.

3.4 Comparison of other models

In order to evaluate the performance of the classification module of DINOV2-FCS proposed in this study, four state-of-the-art mainstream classification models, namely ResNet (He et al., 2016), VIT, ConvNext (Liu et al., 2022), and Swin (Liu et al., 2021), have been selected for comparison. The evaluation metrics chosen are ACC, Macro F1, and Params. It should be noted that these models freeze the backbone network during training as DINOV2-FCS.

Table 3 shows a comparison of the performance of different models on the fruit leaf disease classification task, where our model performs best with 99.67% ACC and Macro F1, and the same number of covariates is about 0.87×10^8. This indicates that the model proposed in this study achieves top level accuracy and F1 score while maintaining relatively compact parameter scales, outperforming all the benchmark models compared. Figure 11 shows scatter plots of the ACC and Params counts of the

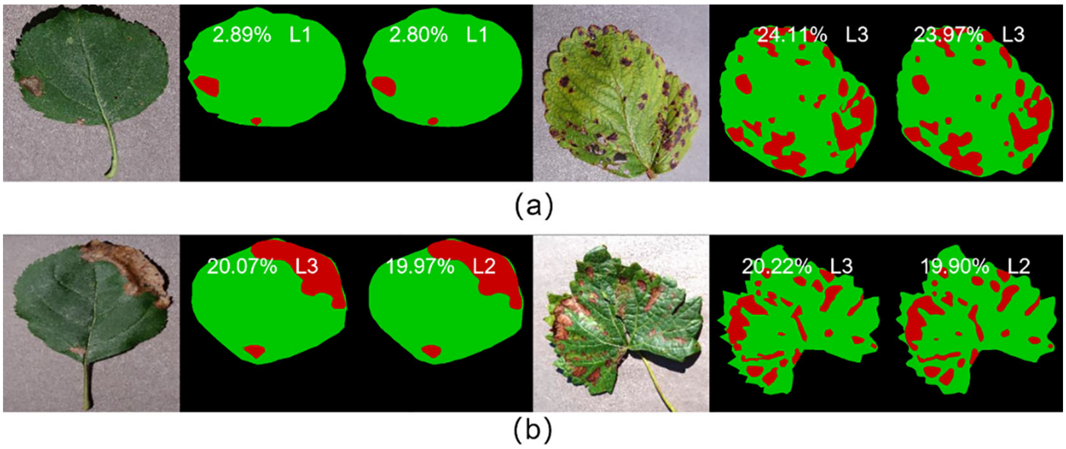


FIGURE 10
(A) Represents the samples with correct prediction of leaf disease severity grading; **(B)** represents the samples with incorrect prediction of leaf disease severity grading.

different models, with five points representing five different models. By observing the position of the points in the plot, we can see that our model performs very well in terms of Params and ACC, outperforming the other four models. In summary, the classification module of DINOV2-FCS proposed in this study is the most outstanding in terms of performance, not only achieving the highest accuracy and F1 score, but also comparable to the Swin base version in terms of model complexity, showing a very high level of efficiency and optimization.

In order to evaluate the performance of the semantic segmentation module for DINOV2-FCS proposed in this study, we selected seven advanced mainstream semantic segmentation models, namely FCN (Long et al., 2015), Deeplabv3+, SETR (Zheng et al., 2021), SegMenter (Strudel et al., 2021), SegFormer, MaskFormer (Cheng et al., 2021) and Mask2Former (Cheng et al., 2022). The comparison is performed. The evaluation metrics chosen are mIoU, PA, Macro F1 and Params. It should be noted that these models are trained with and without backbone network freezing, respectively, and DINOV2-FCS proposed in this study freezes the backbone network during training.

Table 4 shows the performance comparison of several semantic segmentation models on different evaluation metrics, where asterisks denote the freezing of the backbone network, and the model DINOV2-FCS proposed in this study, which leads in all metrics, with 90.29% of mIoU, 94.61% of Macro F1, 98.13% of PA, and 1.50 × 10^8 of Params, reflecting the effectiveness and progress of the model design. Figure 12 shows the scatter plots of mIoU and Params for different models, where each color represents one model. In the models, circles represent training without freezing the backbone network, triangles represent training with freezing the backbone network, and pentagram represents the model proposed in this study. By observing the position of the pentagram in the figure, we can see that our model outperforms the other models in terms of Params and mIoU. In the case of freezing the backbone network, all the other models show performance degradation, but the model proposed in this study still outperforms all the models in

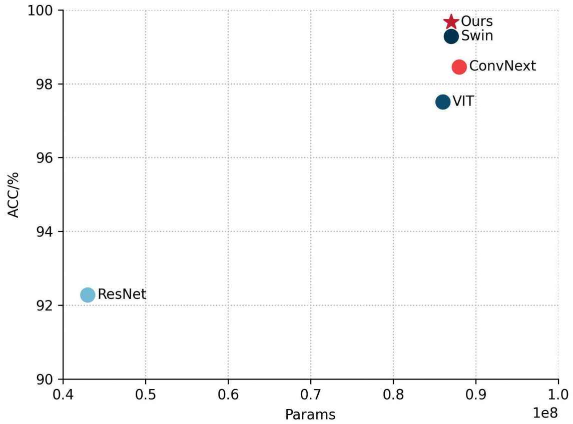


FIGURE 11
Scatterplot of ACC and Params for different models.

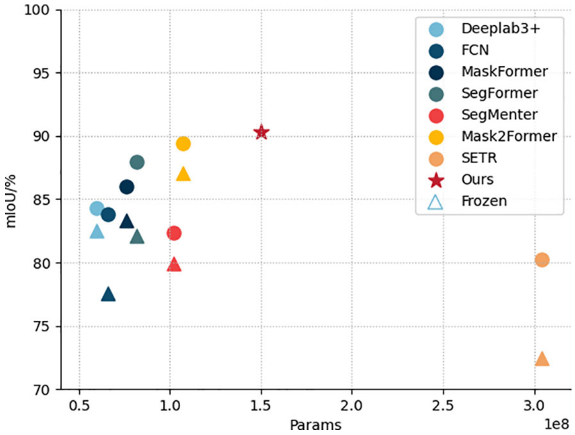


FIGURE 12
Scatterplot of mIoU and Params for different models.

TABLE 4 Segmentation performance of different models.

Model	mIoU/ %	Macro F1/%	PA/ %	Params
FCN(R101)	83.83	90.30	96.79	0.66×10^8
FCN(R101)*	77.53	85.34	95.46	0.66×10^8
Deeplabv3+(R101)	84.32	90.66	96.86	0.60×10^8
Deeplabv3+(R101)*	82.48	89.31	96.49	0.60×10^8
SETR(VIT-L)	80.28	87.60	96.06	3.04×10^8
SETR(VIT-L)*	72.42	80.47	94.55	3.04×10^8
SegMenter(VIT-B)	82.38	89.23	96.47	1.02×10^8
SegMenter(VIT-B)*	79.92	87.37	95.84	1.02×10^8
SegFormer(MIT-B5)	87.96	93.15	97.59	0.82×10^8
SegFormer(MIT-B5)*	82.11	89.01	96.46	0.82×10^8
MaskFormer(R152)	86.03	91.88	97.12	0.76×10^8
MaskFormer(R152)*	83.34	89.96	96.60	0.76×10^8
Mask2Former(SwinB)	89.39	94.07	97.81	1.07×10^8
Mask2Former (SwinB)*	87.10	92.60	97.34	1.07×10^8
Ours*	90.29	94.61	98.13	1.50×10^8

("*" indicates that the backbone network was frozen during model training.)

terms of performance in the case of freezing the backbone network. In summary, this study proposes that the semantic segmentation module of DINOV2-FCS has the best performance, not only achieving the highest mIoU, Macro F1 and PA. Meanwhile, the Params is smaller than that of SETR, which demonstrates its superiority in semantic segmentation tasks.

In Figure 13, the models Mask2Former, SegFormer, Maskforme, Deeplabv3+, and FCN, which exhibited superior performance on the dataset, are presented for comparison with the models in this study. It can be observed that although they also achieved satisfactory results, instances were identified where a considerable number of lesions were not entirely segmented, and even numerous fine lesions were not detected. In contrast, the model proposed in this study is not subject to the same limitations when segmenting fruit leaf disease images, and the overall segmentation effect is superior. This is due to the powerful feature extraction capability of DINOV2 and the improvement of the model by the characteristics of the disease spots in this study.

4 Discussions

4.1 Effectiveness of DINOV2 backbone network

In order to verify the feature extraction capability of the DINOV2 trunk feature extraction network, we performed principal component analysis (PCA) on the patch features extracted by the DINOV2 model. The features of the input image extracted by this model were subjected to PCA dimensionality reduction in order to map the high-dimensional features to the three-dimensional space. The background and foreground portions

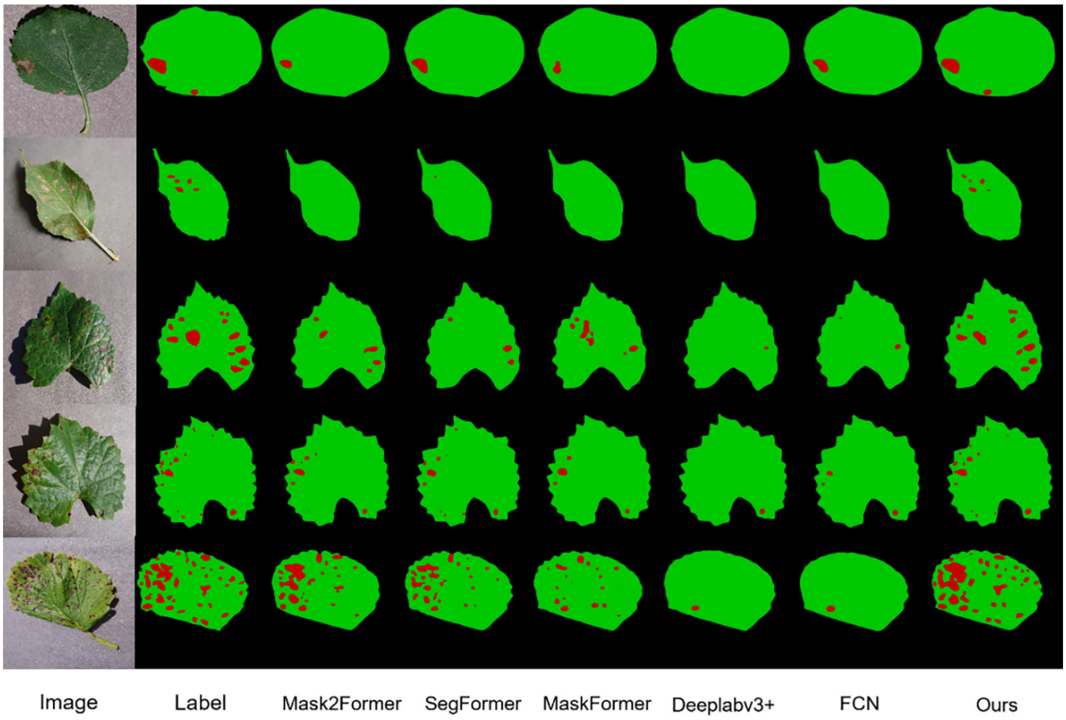


FIGURE 13
Segmentation effect of different models.

of the image were then judged based on the results of PCA, with the principal components of the foreground portion being renormalized in order to highlight them. The visualization facilitates comprehension of the feature extraction effect of the DINOv2 model on the image, as well as the structure and distribution in the feature space after dimensionality reduction by PCA. As illustrated in Figure 14, the DINOV2 model exhibits high performance in distinguishing between foreground and background regions in the image, and in delineating the boundaries of the main objects in the picture. Moreover, the DINOV2 backbone feature extraction network has not encountered these images prior to extraction, and the backbone feature extraction network remains fixed throughout the training process of this working model. This indicates that the DINOV2 backbone feature extraction network is well-suited for the extraction of features in images of fruit leaves affected by disease.

4.2 Effectiveness of C-PFFM

In order to verify the effectiveness of the C-PFFM proposed in this study, ablation experiments are designed to test the effectiveness of the C-PFFM. In the classification module, DINOV2 is used as the backbone feature extraction network in the first group, and one fully connected layer is used as the classifier. The second experimental group, which combined C-PFFM, was constituted on the basis of the first group. The evaluation metrics used are ACC, Macro F1, and Params. The results of the ablation experiments are presented in Table 5. We performed multiple replicated experiments on the proposed models. For the classification model, we selected one of the most important metrics, ACC, to conduct an ANOVA, and the results show that the p-value is 3.8×10^{-4}, and the difference is statistically significant.

As illustrated in the accompanying table, the C-PFFM proposed in this study has demonstrably enhanced the model's predictive capabilities. The benchmark model in the first group achieved an ACC of 97.80%, a Macro F1 of 97.86%, and a Params value of 0.86×10^8. In the second group, the C-PFFM was introduced, which represents an effective fusion of local detail feature information from the patch tokens and global feature information from the class token. This resulted in an enhancement of the classification accuracy of the model. The model achieved an ACC of 99.67%, a Macro F1 of 99.67% and 0.87×10^8 for the Params. The model's accuracy was significantly enhanced with the same number of parameters. This is due to the fact that in the initial set of experiments, only the class token was utilized as input to the fully connected layer, and the class token contains global feature information over long distances. In the context of classifying fruit leaf diseases, there is a notable similarity between the leaf spots of different diseases. This can result in suboptimal model classification accuracy if detailed features are overlooked and only global features are prioritized. The C-PFFM proposed in this study effectively integrates these features, leading to a notable performance improvement.

4.3 Effectiveness of segmentation modules

In order to ascertain the efficacy of the proposed enhancements to the segmentation module in this study, ablation experiments

TABLE 5 Classification module ablation experiment.

	C-PFFM	ACC/%	Macro F1/%	Params
First group	×	97.80	97.86	0.86×10^8
Second group	√	99.67	99.67	0.87×10^8

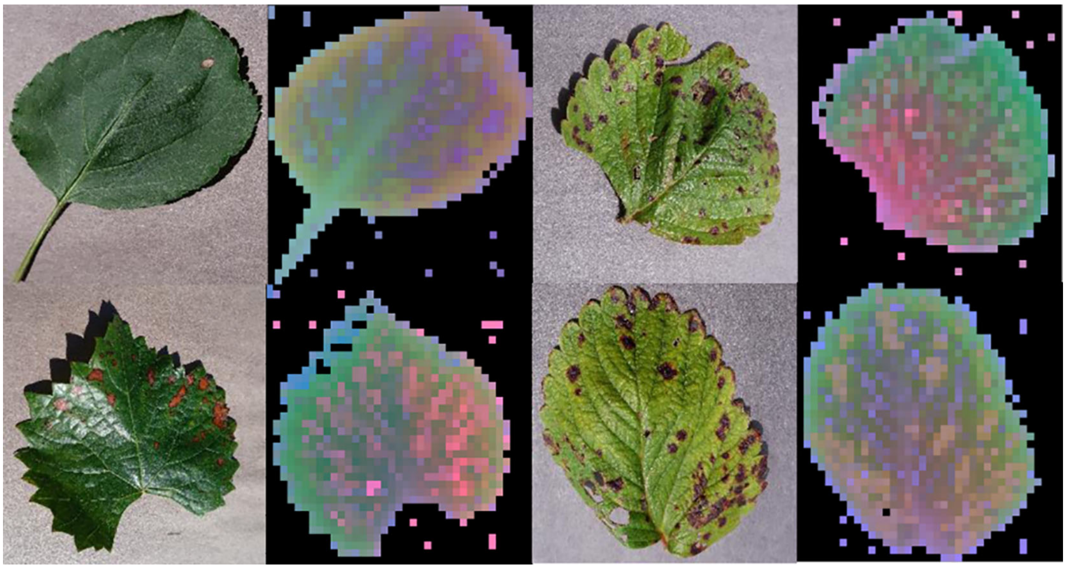


FIGURE 14
Visualization of principal component analysis of DINOV2 generated features.

TABLE 6 Segmentation module ablation experiment.

	EFFA	AKASPP	mIoU/%	Macro F1/%	PA/%	Params
First group	×	×	84.56	90.81	96.98	0.90×10^8
Second group	√	×	88.46	93.45	97.77	1.37×10^8
Third group	×	√	89.22	93.94	97.93	1.03×10^8
Fourth group	√	√	90.29	94.61	98.13	1.50×10^8

have been designed to assess the impact of these improvements. In the segmentation module, the DINOV2 network is employed as the backbone feature extraction network in the first group, resulting in the generation of a segmented image through up-sampling using the MLP decoder. The second experimental group, which combined EFFA, was constituted on the basis of the first group. The third experimental group, which combined AKASPP, was constituted on the basis of the first group. The fourth experimental group, which combined EFFA and AKASPP, was constituted on the basis of the first group. The evaluation indexes are mIoU, Macro F1, PA, and Params. The results of the ablation experiments are presented in Table 6. We performed multiple replicated experiments on the proposed models. For the semantic segmentation model, we selected one of the most important metrics, MIoU, for ANOVA, and the results showed that the p-value was 1.5×10-5, and the difference was statistically significant.

As illustrated in the accompanying table, the proposed enhancements to the segmentation module have demonstrably enhanced the model's performance. The mIoU of the benchmark model in the first group reached 84.56%, the Macro F1 reached 90.81%, the PA reached 96.98%, and the Params was 0.90×10^8. The incorporation of the EFFA into the second group, which fuses explicit feature information with multilevel feature information, resulted in an mIoU of 88.46%, a Macro F1 of 93.45%, and a PA of 97.77%. Additionally, the Params increased to 1.37×10^8. Despite an increase in the number of parameters, there was a notable improvement in accuracy, with an increase of 3.9% in the mIoU. This is attributed to the incorporation of explicit feature information from EVC into multilevel features, which enables the model to simultaneously consider the details and semantic information, thereby enhancing its ability to comprehend the image content. The addition of AKASPP to the third group enables the fusion of contextual and detail edge information from different sensory fields, resulting in an mIoU of 89.22%, a Macro F1 of 93.94%, and a PA of 97.93%, with a Params of 1.37×10^8. With a modest increase in the Params, the mIoU was enhanced by 4.66%, which can be attributed to the fact that the fruit leaf disease image spots exhibit complex shapes, fuzzy edges, and varying sizes. AKASPP effectively fuses contextual and detailed edge information from disparate sensory fields, enabling more precise segmentation of diverse spot targets of varying sizes and shapes, as well as enhanced processing of leaf and spot edge components. The fourth group incorporated both EFFA and AKASPP, based on the findings of the first group. This resulted in an mIoU of 90.29%, a Macro F1 of 94.61%, a PA of 98.13%, and a Params of 1.50×10^8, which achieved the optimal performance.

4.4 Validation of model generalization capabilities

In order to assess the model's ability to generalize, four of the five labeled fruit leaf disease datasets were used as the training set, with one dataset reserved for the test set. The training set includes

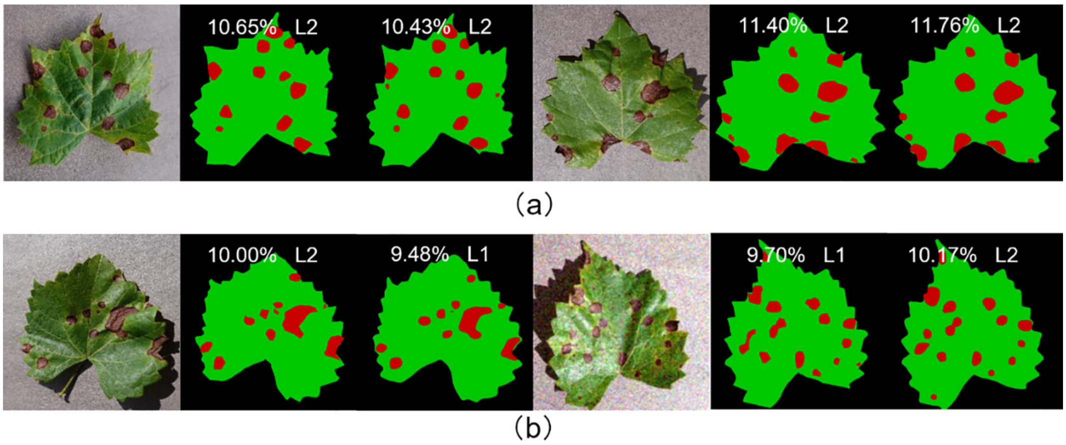


FIGURE 15
(A) Represents the samples with correct prediction of leaf disease severity grading; **(B)** represents the samples with incorrect prediction of leaf disease severity grading.

images of four diseases: apple black rot, cedar apple rust, grape black measles, and strawberry leaf scorch. The test set includes images of grape black rot. The semantic segmentation module achieved an mIoU of 83.95% and the fruit leaf disease severity reached the grading accuracy of 95.24%, thereby verifying the strong generalization ability of the model. As illustrated in Figure 15, the model exhibited strong generalization ability. The model demonstrated effective performance in segmenting diseases that had never been encountered before. The proximity between the ratio of diseased area to total leaf area predicted by the model and the true label was high, and the difference was minimal even in individual samples where the prediction was incorrectly graded.

5 Conclusion

In this study, we constructed the model DINOV2-FCS for leaf disease classification and severity prediction of a variety of fruits based on the DINOV2 large vision model backbone network. The model addresses the shortcomings of current models in disease severity prediction, namely their lack of accuracy and limited generalizability. DINOV2-FCS employs DINOv2-B (distilled) as the backbone feature extraction network to enhance the extraction of features from fruit diseased leaf images. In the context of fruit leaf disease classification, where the leaf spots of different diseases exhibit considerable similarity and the loss of detail information is a significant issue, we propose Class-Patch Feature Fusion Module (C-PFFM), which fuses the local detail feature information of patch tokens and the global feature information of class token. This results in an improvement in the classification accuracy of the model. In light of the fact that the model frequently fails to complete the segmentation of lesions, including those that are subtle, and that lesions are often ignored entirely, we have enhanced the MLP decoder and proposed EFFA, which fuses explicit feature information and multi-level feature information. This has led to an improvement in the segmentation accuracy of the model. Furthermore, we have proposed AKASPP, which fuses contextual information and detailed edge information from different sensory fields, thereby enabling better adaptation to the varying sizes and shapes of lesion targets and the edge details of leaves and lesions. To verify the accuracy and generalizability of the model, two sets of experiments were conducted. First, the labeled leaf disease dataset of five fruits was randomly divided. The trained model exhibited an accuracy of 99.67% in disease classification, an mIoU of 90.29%, and an accuracy of 95.68% in disease severity classification. These results demonstrate superior performance compared to other state-of-the-art models. In the generalizability experiment, four disease data sets were used for training and one for testing. The mIoU of the trained model reached 83.95%, and the accuracy of disease severity grading was 95.24%. The strong generalization ability of the model was verified. The subsequent stage of the process involves the augmentation of the dataset with respect to both species diversity and environmental diversity, thereby aligning it with more realistic scenarios. Furthermore, the model was tested on an NVIDIA GeForce RTX 3090 graphics card, achieving an inference speed of 21.56 frames per second (F/S). The next phase of the project will focus on refining the model to enable its deployment on mobile devices. This will support agricultural workers by assisting with disease identification in the field.

Data availability statement

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

Author contributions

CB: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization. LZ: Writing – review & editing, Conceptualization. LG: Writing – review & editing. LP: Writing – review & editing, Funding acquisition. PL: Writing – review & editing, Methodology, Conceptualization. LY: Writing – review & editing, Supervision, Project administration.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This study was supported by Yunnan Provincial Science and Technology Major Project(No. 202202AE090008) titled “Application and Demonstration of Digital Rural Governance Based on Big Data and Artificial Intelligence”.

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

Atila, Ü., Uçar, M., Akyol, K., and Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. *Ecol. Inf.* 61, 101182. doi: 10.1016/j.ecoinf.2020.101182

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., et al. (2021). Emerging properties in self-supervised vision transformers. *Proc. IEEE/CVF Int. Conf. Comput. Vision*, 9650–9660. doi: 10.1109/ICCV48922.2021.00951

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017a). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 834–848. doi: 10.1109/TPAMI.2017.2699184

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017b). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv* 1706, 5587. doi: 10.48550/arXiv.1706.05587

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *Proc. Eur. Conf. Comput. Vision (ECCV)*, 801–818.

Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, New York: IEEE 1290–1299.

Cheng, B., Schwing, A., and Kirillov, A. (2021). Per-pixel classification is not all you need for semantic segmentation. *Adv. Neural Inf. Process. Syst.* 34, 17864–17875.

Dai, G., Tian, Z., Fan, J., Sunil, C., and Dewi, C. (2024). DFN-PSAN: Multi-level deep information feature fusion extraction network for interpretable plant disease classification. *Comput. Electron. Agric.* 216, 108481. doi: 10.1016/j.compag.2023.108481

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv* 2010, 11929. doi: 10.48550/arXiv.2010.11929

Goncalves, J. P., Pinto, F. A., Queiroz, D. M., Villar, F. M., Barbedo, J. G., and Del Ponte, E. M. (2021). Deep learning architectures for semantic segmentation and automatic estimation of severity of foliar symptoms caused by diseases or pests. *Biosyst. Eng.* 210, 129–142. doi: 10.1016/j.biosystemseng.2021.08.011

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. *Proc. IEEE Conf. Comput. Vision Pattern recognition*, 770–778. doi: 10.1109/CVPR.2016.90

Hu, G., Wei, K., Zhang, Y., Bao, W., and Liang, D. (2021). Estimation of tea leaf blight severity in natural scene images. *Precis. Agric.* 22, 1239–1262. doi: 10.1007/s11119-020-09782-8

Hughes, D., and Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv* 1511, 8060. doi: 10.48550/arXiv.1511.08060

Ji, M., and Wu, Z. (2022). Automatic detection and severity analysis of grape black measles disease based on deep learning and fuzzy logic. *Comput. Electron. Agric.* 193, 106718. doi: 10.1016/j.compag.2022.106718

Khan, A. I., Quadri, S., Banday, S., and Shah, J. L. (2022). Deep diagnosis: A real-time apple leaf disease detection system based on deep learning. *Comput. Electron. Agric.* 198, 107093. doi: 10.1016/j.compag.2022.107093

Khattak, A., Asghar, M. U., Batool, U., Asghar, M. Z., Ullah, H., Al-Rakhami, M., et al. (2021). Automatic detection of citrus fruit and leaves diseases using deep neural network model. *IEEE Access* 9, 112942–112954. doi: 10.1109/ACCESS.2021.3096895

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., et al. (2023). "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, New York: IEEE 4015–4026.

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

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). "A convnet for the 2020s," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, New York: IEEE 11976–11986.

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*, New York: IEEE 3431–3440.

Ma, J., He, Y., Li, F., Han, L., You, C., and Wang, B. (2024). Segment anything in medical images. *Nat. Commun.* 15, 654. doi: 10.1038/s41467-024-44824-z

Madden, L. V., Hughes, G., and Van Den Bosch, F. (2007). *The study of plant disease epidemics* (St. Paul: APS Press), 421.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., et al. (2023). Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv* 2304, 7193. doi: 10.48550/arXiv.2304.07193

Quan, Y., Zhang, D., Zhang, L., and Tang, J. (2023). Centralized feature pyramid for object detection. *IEEE Trans. Image Process.* 32, 4341–4354. doi: 10.1109/TIP.2023.3297408

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

Shrivastava, A., Gupta, A., and Girshick, R. (2016). Training region-based object detectors with online hard example mining. *Proc. IEEE Conf. Comput. Vision Pattern recognition*, 761–769. doi: 10.1109/CVPR.2016.89

Strudel, R., Garcia, R., Laptev, I., and Schmid, C. (2021).). Segmenter: Transformer for semantic segmentation. *Proc. IEEE/CVF Int. Conf. Comput. Vision*, 7262–7272.

Wang, C., Du, P., Wu, H., Li, J., Zhao, C., and Zhu, H. (2021). A cucumber leaf disease severity classification method based on the fusion of DeepLabV3+ and U-Net. *Comput. Electron. Agric.* 189, 106373. doi: 10.1016/j.compag.2021.106373

Wang, C., Liu, S., Wang, Y., Xiong, J., Zhang, Z., Zhao, B., et al. (2022). Application of convolutional neural network-based detection methods in fresh fruit production: a comprehensive review. *Front. Plant Sci.* 13. doi: 10.3389/fpls.2022.868745

Wang, D., Zhang, J., Du, B., Xu, M., Liu, L., Tao, D., et al. (2024). Samrs: Scaling-up remote sensing segmentation dataset with segment anything model. *Adv. Neural Inf. Process. Syst.*, 36.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst* 34, 12077–12090.

Yang, B., Wang, Z., Guo, J., Guo, L., Liang, Q., Zeng, Q., et al. (2023). Identifying plant disease and severity from leaves: A deep multitask learning framework using triple-branch Swin Transformer and deep supervision. *Comput. Electron. Agric.* 209, 107809. doi: 10.1016/j.compag.2023.107809

Zeng, Q., Ma, X., Cheng, B., Zhou, E., and Pang, W. (2020). Gans-based data augmentation for citrus disease severity detection using deep learning. *IEEE Access* 8, 172882–172891. doi: 10.1109/ACCESS.2020.3025196

Zhang, C., Han, D., Qiao, Y., Kim, J. U., Bae, S.-H., Lee, S., et al. (2023a). Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv*, 2306.14289. doi: 10.48550/arXiv.2306.14289

Zhang, C., Han, D., Zheng, S., Choi, J., Kim, T.-H., and Hong, C. S. (2023b). Mobilesamv2: Faster segment anything to everything. *arXiv preprint arXiv*, 2312.09579. doi: 10.48550/arXiv.2312.09579

Zhang, X., Song, Y., Song, T., Yang, D., Ye, Y., Zhou, J., et al. (2023). AKConv: convolutional kernel with arbitrary sampled shapes and arbitrary number of parameters. *arXiv preprint arXiv*, 2311.11587. doi: 10.48550/arXiv.2311.11587

Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., et al. (2023). Fast segment anything. *arXiv preprint arXiv*, 2306.12156. doi: 10.48550/arXiv.2306.12156

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, New York: IEEE 2881–2890.

Zhao, Y., Sun, C., Xu, X., and Chen, J. (2022). RIC-Net: A plant disease classification model based on the fusion of Inception and residual structure and embedded attention mechanism. *Comput. Electron. Agric.* 193, 106644. doi: 10.1016/j.compag.2021.106644

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., et al. (2021). "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers." in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* 2021, 6881–6890.

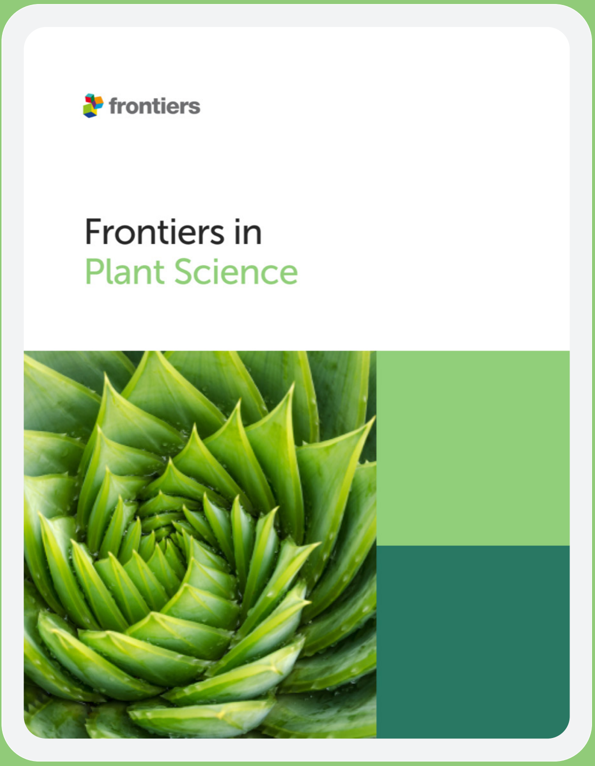
Frontiers in
Plant Science

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