ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 3 - 2026

Research on tomato plant disease and pest detection model based on BSAP-YOLOv11 algorithm

Longtao Xiong[1], *Xiang Zhou*[1,*], *Rongpei Lu*[1], *Zhongjin Ni*[1], *Yihua Ni*[1] *and Yan Lu*[1]

[1] College of Optical, Mechanical and Electrical Engineering , Zhejiang A&F University, Hangzhou 311300, Zhejiang, China

SUMMARY: *In response to the problems of low recognition accuracy of tomato plant diseases and pests, high rate of missed and false detections of small targets, and difficulty in balancing model lightweight and real-time performance, this paper proposes an improved YOLOv11 detection model (BSAP-YOLOv11) that integrates BiFPN, Shuffle Attention, and P2 detection head. Firstly, using YOLOv11n as the baseline, a P2 small object detection head was added and redundant scale layers were removed. BiFPN bidirectional feature pyramid was introduced to enhance multi-scale feature fusion, and Shuffle Attention mechanism was embedded to enhance lesion feature expression; Secondly, in the preprocessing of tomato plant disease and pest datasets, a balanced dataset containing 9 types of diseases was constructed using GAN generative enhancement and conventional amplification to enhance feature expression ability. The experiment showed that the mAP0.5 of the BSAP-YOLOv11 model reached 99.13%, mAP0.5:0.95 reached 92.50%, FPS was 132.65 frames per second, and FLOPs were only 5.32G. Compared with the selected baseline algorithm, it has the best comprehensive performance, can achieve low missed detections, low false detections, and significantly improve counting accuracy. Research has shown that BSAP-YOLOv11 achieves a good balance between accuracy, speed, and lightweight, and can accurately identify multiple types of tomato pests and diseases, providing an effective technical solution for intelligent monitoring of tomato pests and diseases.*

KEYWORDS: *tomato pests and diseases; YOLOv11; BiFPN; Shuffle-Attention; P2 detection head; object detection*

1 Introduction

Tomato plants are not native crops in China. Since their introduction during the Ming Dynasty, they have gradually become one of the most widely planted crops in China. From the current development of international vegetable trade, tomato plants also occupy an important position in global vegetable trade. As a key issue that has long constrained agricultural development, crop diseases and pests can lead to significant reductions or even complete crop failures in crops such as tomato plants. Tomato plant leaves, as the intuitive response organs to pest and disease infections, often present symptoms through features such as withered spots [1]. Although traditional diagnostic methods relying on manual experience are effective, they have shortcomings such as low diagnostic efficiency and high misjudgment rates. Affected by various pests and diseases, the fluctuation of tomato plant yield can affect people's daily lives and the trading situation in the vegetable market, causing great economic losses to growers.

*1015251292@qq.com
https://doi.org/10.65102/is20261136
Published: 10 June, 2026

The traditional practice of treating tomato plant diseases and pests usually involves large-scale pesticide spraying, which can to some extent recover some economic losses for growers, but it can also cause pesticide residues in tomato plants, resulting in serious environmental pollution [2]. If we can timely and accurately identify tomato plant diseases and pests, and adopt effective methods of management, it will help to control the scale of diseases and pests within a controllable range, increase tomato plant yield, and protect the environment. With the development of deep learning technology, there is theoretical technical support for the rapid and accurate diagnosis of tomato plant diseases and pests. When tomato plants are invaded, the texture features on the surface of diseased tomato leaves are different from those on healthy leaves, which is the key to determining whether tomato plants are infected with pests and diseases. The introduction of computer vision technology directly provides an effective solution for identifying tomato plant diseases and pests, which can achieve rapid identification of tomato plant disease and pest types. Its advantages are reflected in multiple dimensions such as detection timeliness, diagnostic accuracy, and technical scalability, which improves the efficiency of tomato plant disease and pest management and provides technical support for establishing a scientific tomato planting system [3].

The object detection algorithm under deep learning is a fast and accurate algorithm for detecting tomato plant diseases and pests. This paper adopts an object detection algorithm based on the improved YOLOv11 algorithm to achieve the detection of tomato plant leaf diseases and pests. Its main practical significance is to effectively shorten the prevention and control time of tomato plant diseases and pests. In the early stages of tomato plant diseases and pests, target detection algorithms are used to detect the types of tomato plant diseases and pests, develop corresponding prevention and control plans, increase yields, and recover economic losses for tomato plant growers. (2) Effectively reduce the use of pesticides and protect the environment. At the budding stage of pest and disease occurrence, corresponding prevention and control measures can be taken, and the extensive spraying of pesticides is no longer used to reduce environmental pollution. (3) The prevention and control methods for tomato plant diseases and pests are more scientific. It can alleviate the problem of insufficient professional prevention and control personnel and overcome the limitations of human observation. (4) Alleviating the shortage of experts in tomato plant disease and pest control. People only need to carry an electronic device to monitor pests and diseases in real time, without the need for disease prevention personnel with professional skills to participate, which helps alleviate the current shortage of experts in the agricultural field.

2 Related Work

Currently, global research teams have made breakthrough progress in the field of agricultural intelligence, with the innovative application of visual perception technology being the most significant. The detection system based on the fusion of hyperspectral imaging and computer vision technology is deeply penetrating into the entire process of crop phenotype analysis, covering multiple dimensions such as leaf morphology analysis, dynamic monitoring of fruit quality, and precise diagnosis of pathogen infection, providing key technical support for establishing a crop health warning system [4]. It not only reconstructs the traditional agricultural monitoring paradigm, but also promotes the paradigm shift of plant protection from empirical judgment to data-driven.

Compared with traditional machine vision technology, deep learning methods have shown significant advantages in crop pest and disease recognition and classification accuracy. In the field of pest and disease research abroad, the research focus is mostly on optimizing and innovating algorithm models. By introducing cutting-edge algorithms such as deep learning, researchers are not only committed to improving recognition accuracy, but also focus on improving recognition speed and real-time performance. Classic neural network models such as AlexNet, DenseNet, and GoogleNe are often used as the foundational framework for plant disease and pest classification research. Utpal Barman et al. [5] used a visual transformer model to distinguish healthy and diseased tomato plants, and trained a deep learning model using a dataset consisting of 9985 images of tomato plant leaves covering 10 different disease categories. The performance test accuracy of the proposed visual transformer model reached 90.99%. Utpal Barman et al. [6] launched the ViT Smart application, which enables real-time detection of tomato plant health and provides practical tools for farmers. This combination of advanced deep learning and mobile technology highlights the potential of scalable intelligent agricultural solutions. Vijesh Kumar Patel et al. [7] developed a machine learning based image processing and optimization system that constructs a robust dataset using image enhancement techniques such as rotation, flipping, and scaling. The dataset is then resized and normalized using Lanczos interpolation for analysis. The processed data is fed into a Moth Flame optimized recurrent neural network for detecting crop pests and diseases. ThanammalIndu et al. [8] proposed an automatic recognition system for tomato plant leaf diseases and pests, which uses a pre trained convolutional neural network (CNN) architecture with cross drive optimization algorithm to significantly improve the classification accuracy of tomato plant leaf diseases. Vijesh Kumar Patel et al. [9] developed a machine learning based system for early detection of crop diseases and pests, focusing on image processing techniques and optimizing the algorithm model using whale optimization algorithm to achieve effective segmentation of crop lesion areas. Finally, the optimized recurrent neural network was used for classification and detection of diseases and pests. Vijesh Kumar Patel et al. [10] developed an innovative hybrid model for tomato plant disease and pest detection, which was constructed using convolutional neural networks (CNN) and achieved early recognition and diagnosis of diseases and pests using imaging technology and artificial intelligence. Finally, the performance of the pre trained model was improved through transfer learning and fine-tuning strategies. The experimental results showed that the final test accuracy was as high as 98.1%. Vijesh Kumar Patel et al. [11] proposed a novel crop pest and disease identification technology based on deep learning, which adopts transfer learning methods and has achieved significant results in achieving fast and simple pest and disease detection. The developed model can successfully identify 6 different types of mung bean diseases and pests, as well as 4 types of pests, from healthy and damaged leaves collected in different seasons. The average accuracy of the deep learning model for plant disease and pest detection is as high as 93.65%. This technology not only enhances the convenience for farmers to access advanced agricultural tools, but also provides important support for improving crop management and protecting yields.

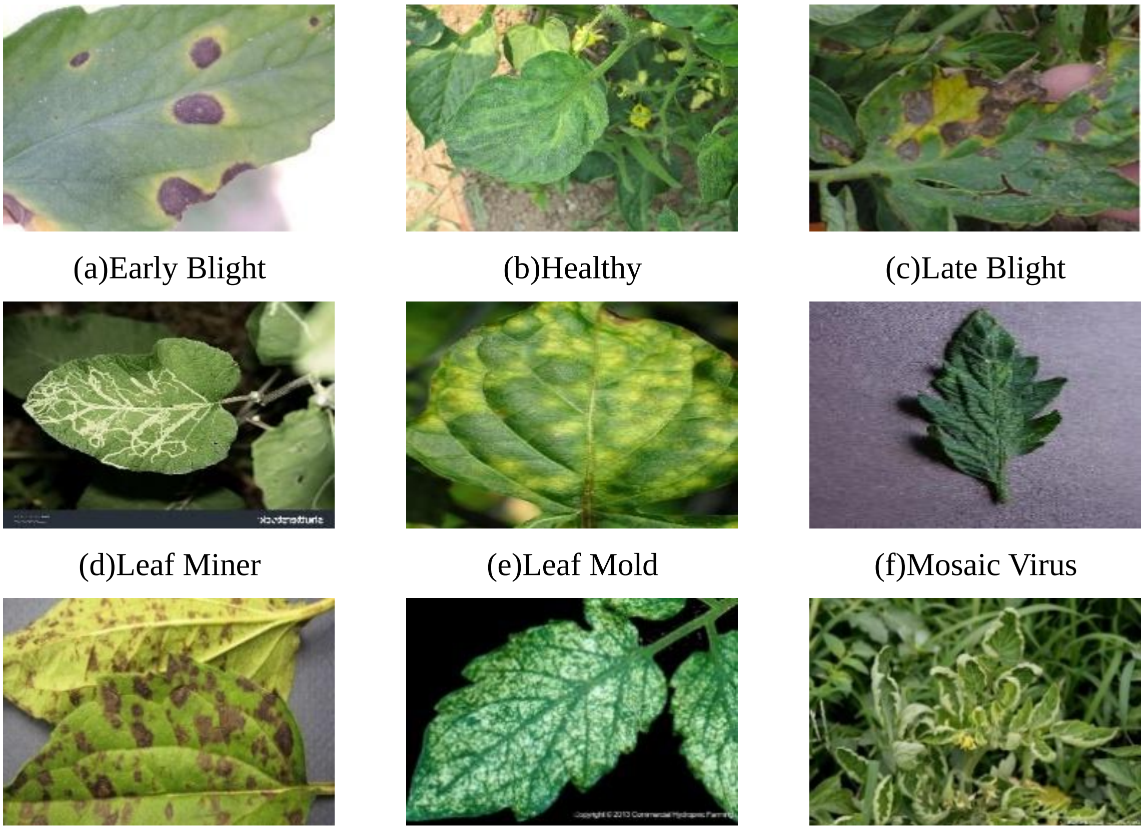
Although significant achievements have been made in research on various types of pests and diseases in practical applications, there is still room for further optimization of deep neural networks in crop pest and disease recognition accuracy with the continuous changes in application scenarios and the demand for continuous improvement in model performance. This article takes tomato plant leaf pests and diseases as the research object, and proposes a tomato pest and disease detection model based on BiFPN, Shuffle Attention, and P2 detection head YOLOv11 algorithm (BSAP-YOLOv11) to address the problems of low recognition accuracy, misjudgment, and missed detection. This model effectively improves the recognition accuracy of tomato plant pests and diseases.

3 Establishment of Tomato Plant Leaf Disease and Pest Dataset

3.1 Dataset Construction

At present, research on tomato leaf disease detection mainly focuses on the recognition and localization of lesion areas. However, leaf diseases often affect the health condition of the entire leaf, and their appearances and scales vary significantly across disease types and infection stages. Therefore, in this study, we treat the whole leaf as the detection target and perform disease recognition and analysis at a global scale, rather than concentrating only on local lesions.

The tomato leaf disease dataset used in this work, "Tomato Diseases Detection," was collected from the PlantVillage Dataset platform and contains images captured in both outdoor and laboratory environments. The laboratory images were enhanced by simulating natural backgrounds to improve realism, while the outdoor images cover diverse illumination conditions (e.g., direct sunlight and partial occlusion by leaves). In addition, the dataset includes lesion variations across different growth stages and disease progression. It comprises nine disease categories, including early blight, leaf mold, mosaic virus, septoria leaf spot, spider mite damage, tomato yellow leaf curl virus, late blight, leaf miner, and target spot (Fig. 1).



(a)Early Blight (b)Healthy (c)Late Blight
(d)Leaf Miner (e)Leaf Mold (f)Mosaic Virus
(g)Septoria (h)Spider Mites (i)TYLCCV

Figure 1: Tomato Pests and Diseases

3.2 Generative Data Enhancement Methods

Generative data augmentation methods mainly use generative adversarial networks to generate images of tomato plant diseases and pests [12, 13]. In GAN, the generator and discriminator are essentially differentiable functions that oppose each other. The generator generates fake images by receiving random noise input, while the discriminator determines whether the image is a real image or a generated image. If it is a real image, the result obtained by the discriminator is 1; If it is generating an image, the result obtained by the discriminator is 0 [14]. When the discriminator cannot correctly distinguish the authenticity of an image, it is considered that the generated image can be mistaken for real, and the generator can generate an image with the same probability distribution as the real image. The basic structure of GAN generative data augmentation method is shown in Figure 2.

In the process of training a GAN generative adversarial network, firstly, a fixed generator is used to feed the real image and the generated image into the discriminator for training, so that the discriminator can recognize the authenticity of the image; Secondly, a fixed discriminator is used to feed the generated image into the discriminator so that it can deceive the discriminator, and the process is repeated [15, 16]. The loss function of GAN generative data augmentation method is shown in equation (1):

$$\min \max V(D,G) = E_{x\sim P_{\text{data}}(x)}\left[\lg D(x)\right] + E_{z\sim P_z(z)} \lg\left\{1 - D[G(z)]\right\} \tag{1}$$

where, D represents discriminator, x represents real data, G represents generator, z represents random noise, $P_z(z)$ represents generated samples, $P_{\text{data}}(x)$ represents real dataset.

The purpose of training the discriminator is to make the $\lg D(x)$ as large as possible, which can better play the role of the discriminator in distinguishing truth from falsehood. The purpose of training the generator is to deceive the discriminator with false images generated by random noise z, that is, the smaller the $1-D[G(z)]$, the better, in order to achieve the goal of deceiving the real.

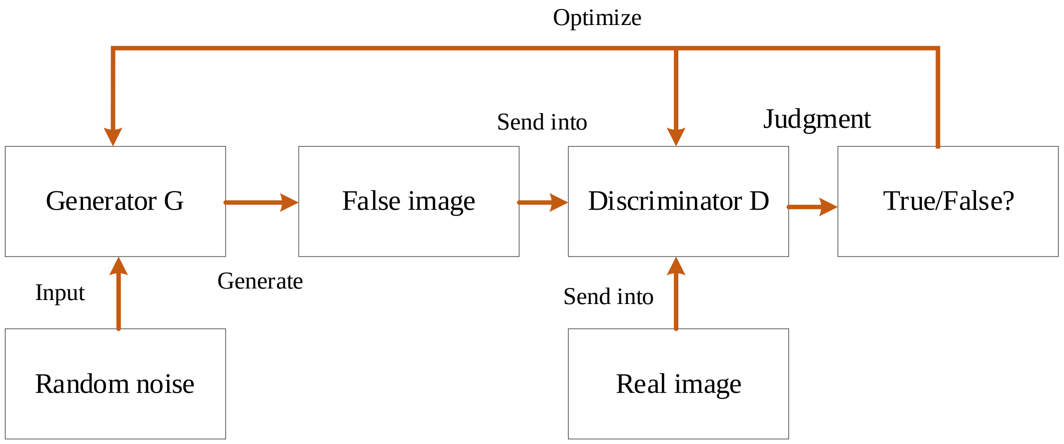


Figure 2: GAN Generative Data Enhancement Method

3.3 Dataset Organization

To enhance model generalization and reduce overfitting, data augmentation was employed to expand the original image dataset. The augmentation strategies included random rotation, flipping, brightness adjustment, and cropping, which were designed to mimic the diverse leaf poses and environmental disturbances encountered during tomato growth. This process improves the model’s adaptability and robustness in real-world conditions [17]. After augmentation, the number of images increased from 3,574 to 9,300.

To further improve detection performance, a subset of pest and disease images was additionally cropped during preprocessing to standardize the object scale. All images were then resized to 640 × 640. Finally, the dataset was split into training, validation, and test sets with a ratio of 7:1.5:1.5, yielding 6,510 training images, 1,395 validation images, and 1,395 test images, as summarized in Table 1.

Table1: Data set of tomato diseases and pests

Category	Train	Validation set	Test set	Total set
Early Blight	489	105	105	699
Healthy	512	110	110	732
Late Blight	791	170	170	1131
Leaf Miner	700	150	150	1000
Leaf Mold	818	175	175	1168
Mosaic Virus	889	191	191	1271
Septoria	833	179	179	1191
Spider Mites	626	134	134	894
TYLCCV	852	181	181	1214

4 BSAP-YOLOv11 Detection Model for Tomato Plant Leaf Pests and Diseases

4.1 Overall Framework of Detection Algorithm

YOLOv11 [18, 19] is a new-generation object detector evolved from YOLOv5 and YOLOv8. Its architecture follows the standard three-part design, consisting of a Backbone, Neck, and Head. By introducing the C3K2 module into the backbone and combining it with an adaptive anchoring mechanism and efficient feature extraction strategies, YOLOv11 achieves improved performance in small-object detection. Moreover, the multi-scale prediction scheme is refined to accelerate inference without sacrificing accuracy. To further enhance representational capacity under limited computation, YOLOv11 incorporates C2PSA (partial spatial attention) and adopts a dynamic detection head together with depth wise separable convolutions, effectively reducing parameters and FLOPs while maintaining strong real-time capability in resource-constrained scenarios. Following refinement, the BSAP-YOLOv11 model was obtained, with its network architecture diagram shown in Figure 3.

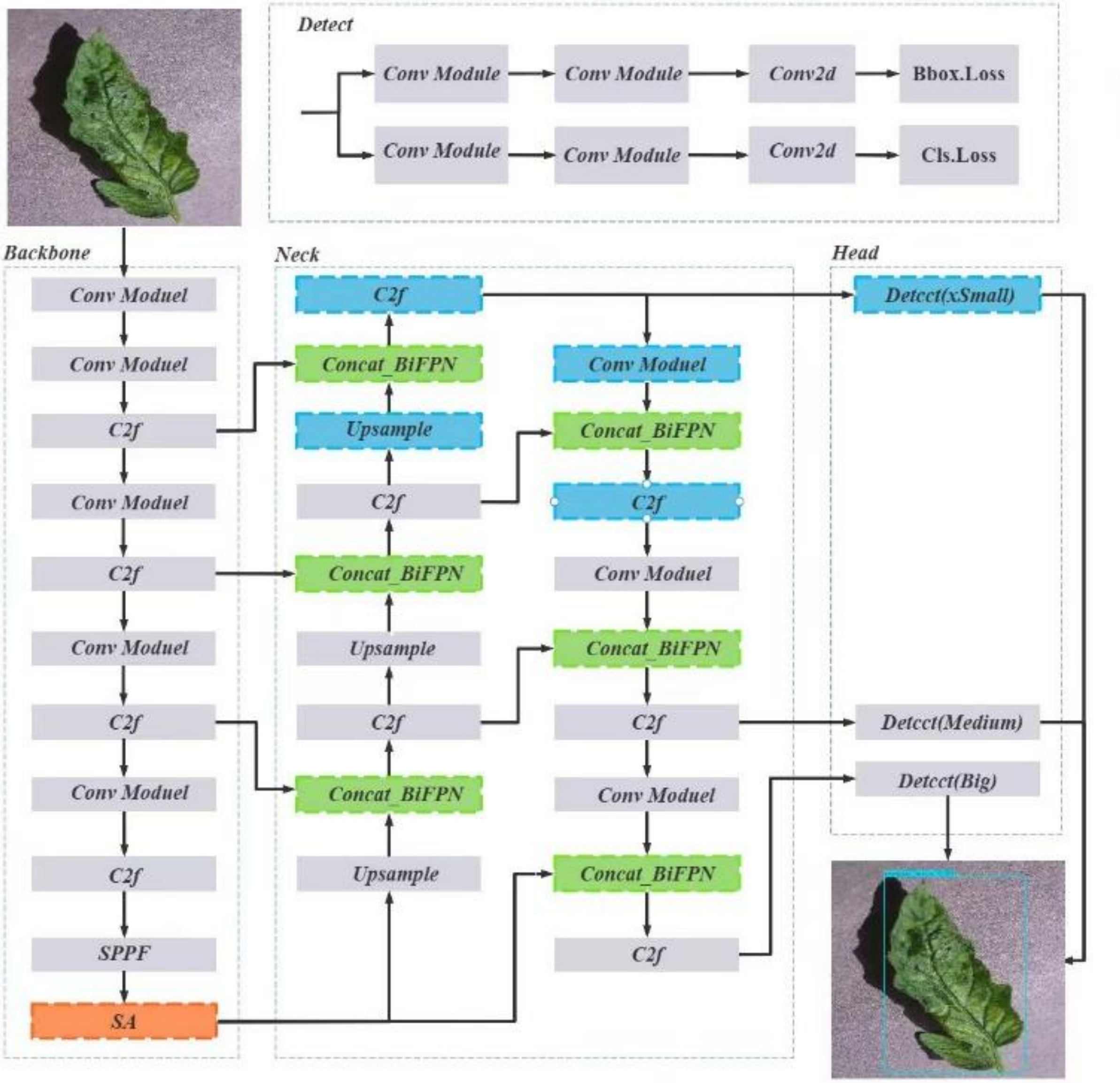


Figure 3: BSAP-YOLOv11 Model Diagram

Given the computational constraints of this study, we select the lightweight variant YOLOv11n as the baseline to balance accuracy and efficiency. On this basis, we introduce the following key architectural modifications:

(1) Detection-head redesign with an added P2 scale and removal of P3. To better handle small objects, we add a P2-scale detection head and remove the P3-scale head. The P2 feature map (4× downsampling) provides higher spatial resolution, enabling more precise localization and recognition of small targets. Removing P3 reduces redundancy in intermediate-scale prediction, allowing the model to focus on small objects at P2 and larger objects at P4–P5, thereby improving efficiency and overall performance. The relevant content of P2 detection head is shown in section 4.2.

(2) BiFPN-enhanced neck for multi-scale fusion. We integrate a Bi-directional Feature Pyramid Network (BiFPN) into the neck to strengthen multi-scale feature aggregation. With learnable fusion weights and feature concatenation, BiFPN can adaptively emphasize informative feature levels, which is particularly beneficial for improving detection accuracy of small targets under scale variation. The relevant content of BiFPN enhanced neck is shown in

section 4.3.

(3) Shuffle-Attention (SA) for stronger small-target representation. We incorporate the Shuffle-Attention (SA) module to enhance feature representation for small objects. Through grouped feature processing and joint channel–spatial attention, SA selectively highlights discriminative channels and spatial regions. This improves the detectability of small targets such as tomato pests and diseases while introducing only limited additional computational overhead. The content related to Shuffle-Attention is shown in section 4.4.

4.2 P2 Detection Head

The original YOLOv11 model contains three detection layers at different scales [20, 21], namely 80 × 80, 40 × 40, and 20 × 20, corresponding to the P3, P4, and P5 layers. Three different sizes of detection layers have different receptive fields and are used to detect targets of different sizes. In the detection of tomato plant diseases and pests, operators are often far away from the front-end video capture equipment, and the size of leaf diseases and pests is small, resulting in a small proportion of collected tomato plant leaf disease and pest targets in the image, and most of them are small targets. There are many small targets in the images of tomato plant diseases and pests, and the feature maps generated by the large target detection layer contain less feature information of small targets due to the large downsampling factor, resulting in poor detection performance. In response to the above issues, considering that the effectiveness of the large object detection head in identifying small targets of tomato plant diseases and pests is not significant, based on the need to improve the small object detection performance of the model and lightweight the model, the P5 large object detection layer is deleted while adding the P2 small object detection layer. The improved model Neck adopts the ERepGFPN structure, and the added small object detection layer is shown in Figure 4.

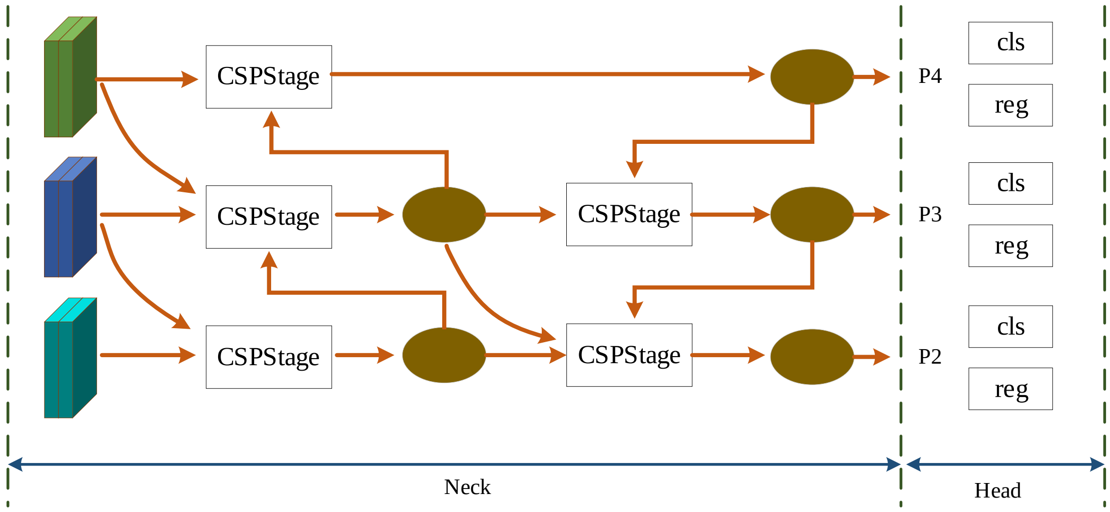


Figure 4: P2 Detection Head

The diagonal downward arc in Figure 4 represents the upsampling operation, and the right-angled connecting line with the upward arrow represents the downsampling operation.

4.3 BiFPN module

The Google Brain team introduced the Bidirectional Feature Pyramid Network (BiFPN) in the EfficientDet object detection algorithm [22], which is a further improvement on PANet. The BiFPN structure is shown in Figure 5a. This article introduces BiFPN into YOLOv11 for feature fusion, and further enhances the feature extraction and fusion capabilities of the YOLOv11 model by introducing learnable parameters to adjust the weights of each input feature.

To enhance the model performance in multi-scale tomato pesto and disease detection and segmentation, we incorporate a BiFPN structure into the neck of YOLOv11. The original YOLOv11 neck combines FPN and PANet to integrate high-level semantic features with low-level spatial details. However, its feature propagation is still largely single-directional, and cross-scale interactions remain limited. As a result, fine-grained information may gradually attenuate during feature transmission, which negatively affects the recognition of small lesions and tiny pest instances. In contrast, BiFPN (Bi-directional Feature Pyramid Network) [23, 24] strengthens multi-scale feature fusion through bidirectional information flow, i.e., both top–down and bottom–up pathways. This design improves semantic aggregation while preserving local details, thereby enhancing small-target representation and improving the robustness and accuracy of detection and segmentation under complex greenhouse conditions.

During feature fusion, the input feature maps have different spatial resolutions. Conventional fusion strategies may introduce misalignment or inconsistent feature representations, which can degrade overall performance [25]. To address this issue, BiFPN employs a learnable weighted fusion mechanism. Specifically, it assigns adaptive weights to each input feature along the fusion paths and optimizes these weights during training, allowing the network to emphasize informative features and suppress less useful ones. This mechanism improves feature representation quality and facilitates the extraction of discriminative information, which is particularly beneficial for multi-scale detection by reducing missed detections and false detections caused by large variations in the size and appearance of tomato disease and pest targets. The relationship between the input and output features is given in Eq. (1) [26]:

$$o = \sum_i \frac{w_i}{\varepsilon + \sum_j w_j} I_i \tag{2}$$

where, o denotes the fused output feature, I_i represents the input feature, and w_i is the corresponding learnable weight. The constance ε is set to 0.0001. The fusion result is ε constrained using nomalized weights.

4.4 Shuffle-Attention Model

The attention mechanism enables the network to accurately focus on all relevant elements of the input. In monocular depth estimation tasks, the attention mechanism can capture pixel level pairwise relationships and channel information, improving the accuracy of monocular depth estimation [27, 28]. At the same time, due to the excellent multi-scale feature map fusion ability of the attention mechanism, it is often applied in the decoder module. In this study, Shuffle Attention was added to each stage of the decoder module, allowing the decoder module to pay more attention to the effective information in all feature maps during the extraction of low-resolution feature map information. The Shuffle Attention structure is shown in Figure 5.

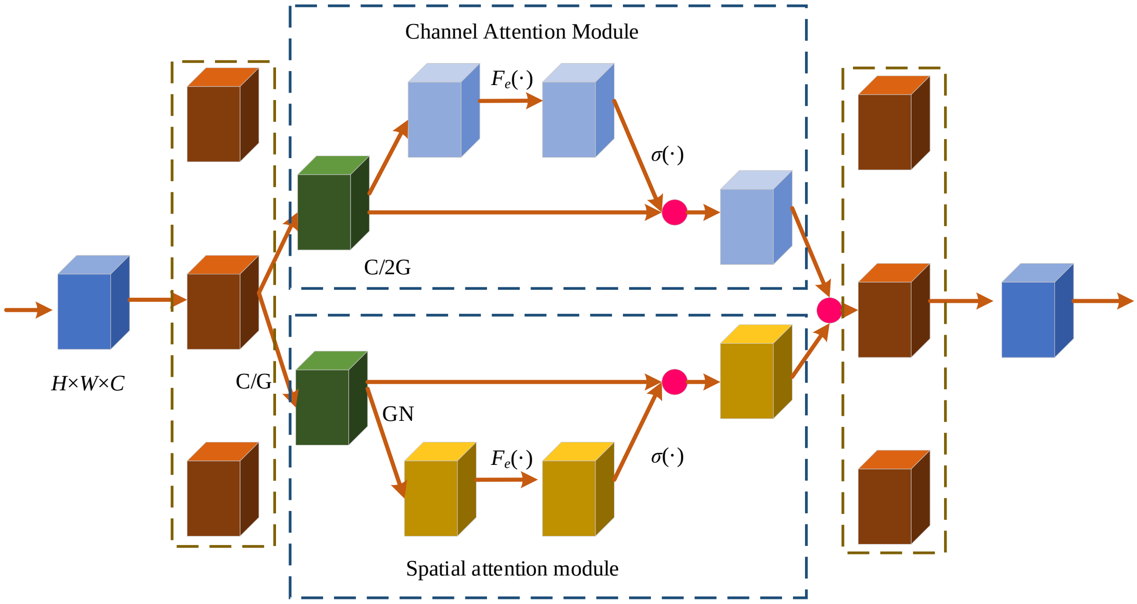


Figure 5: Shuffle-Attention Module

For a given feature map $X \in \boldsymbol{R}^{C\times H\times W}$, where C, H, and W represent the number of channels, image height, and image width, respectively, Shuffle Attention first divides the feature map into G groups along the channel dimension. After modifying the feature map, the number of channels becomes C/G, and the divided feature map becomes $X_K \in \boldsymbol{R}^{C/G\times H\times W}$. X_K is then divided into 2 branches, the number of channels becomes $C/(2G)$, and the feature map becomes $X_{K1} \in \boldsymbol{R}^{C/(2G)\times H\times W}$, which are sent to the channel attention module and spatial attention module respectively. Then, the 2 modules are output connected so that the number of channels is equal to the number of input channels, i.e. $X_{k10} \in \boldsymbol{R}^{C/G\times H\times W}$. Finally, the used sub features are aggregated, and the Shuffle operator is used to transmit information across groups along the channel dimension. Finally, Shuffle Attention is applied. The output of the module is the same as the input feature map X.

5 Experimental Analysis

5.1 Experimental Setup

In this study, the computing platform configuration used is as follows [29]: the processor is Intel (R) i7-1270P, with a clock speed of 3.6GHz; the system is equipped with 16GB RAM DDR5-6000MHz, 1.5TB SSD storage, and Colorful NVIDIA 8GB graphics card. The software environment includes the Windows 10 operating system with CUDA11.9 and Cudnn8.1.9 installed, as well as Python 3.85.

In addition, the deep learning development framework used in this experiment is Pytorch 2.0.9 version. During training, YOLOv11.pt pre trained weights provided by the official system were selected, and the label format was YOLO format. The input image size was 628 × 628, the Batchsize was 18, and the number of epochs was 200. Adam optimizer was used and the initial learning rate was set to 0.005. This study selected the following indicators to evaluate the model [30]: recall (R), precision (P), mean average precision (mAP), average precision (AP), and frames per second (FPS).

5.2 Comparison of Attention Mechanism Detection Effects

In order to evaluate the performance of the Shuffle Attention mechanism, the Shuffle Attention mechanism module was compared with the Channel Attention mechanism module SENet (Squeeze and Excitation Networks) and the Convolutional Block Attention module CBAM. The results are shown in Table 2.

Table 2: Performance of various attention mechanism models during the training process

Model	P%	R%	mAP0.5%	FPS/(frames/s)
Shuffle-Attention-YOLOv11	96.97	95.82	99.12	132.65
SE-YOLOv11	93.68	90.46	92.43	127.53
CBAM-YOLOv11	92.17	89.27	91.58	127.74

According to the experimental data in Table 2, among the three attention mechanisms, Shuffle Attention has the best comprehensive performance, with accuracy P%, recall R%, mAP0.5% reaching 96.97%, 95.82%, and 99.12%, respectively, and FPS of 132.65 frames per second, significantly better than SE and CBAM attention mechanisms. Compared with SE, Shuffle Attention has improved accuracy by 3.29%, recall by 5.36%, mAP0.5 by 6.69%, while still maintaining an advantage of 4.12 frames per second in FPS; Compared with CBAM, Shuffle Attention has improved accuracy by 4.80%, recall by 6.55%, mAP0.5 by 7.54%, and FPS by 4.91 frames per second. The results show that Shuffle Attention, through grouping channel processing and joint channel spatial attention, can more accurately focus on small lesions and texture features of tomato pests and diseases, effectively reducing missed and false detections, and has controllable computational costs, faster inference speed, and is more suitable for real-time detection needs of tomato pests and diseases in the field.

5.3 Ablation Test

This study simplified the BSAP-YOLOv11 network item by item and conducted ablation experiments on the test set. The specific experimental results are shown in Table 3.

Table 3: Experimental results of different ablation methods for YOLOv11

P2 detection head	BiFPN	Shuffle-Attention	mAP0.5/%	mAP0.5:0.95/%	FPS/(frames/s)	FLOPs/G
--	--	--	78.72	62.56	117.37	14.19
√	--	--	89.03	71.57	127.69	5.32
√	√	--	95.09	82.48	129.36	5.32
√	√	√	99.13	92.50	132.67	5.32

According to the experimental data in Table 3, the mAP0.5 of the original YOLOv11 baseline tomato plant disease and pest detection model is 78.72%, mAP0.5:0.95 is 62.56%, FPS is 117.37 frames per second, and the computational load is 14.19G FLOPs. After adding the P2 detection head alone, the mAP0.5 of the tomato plant disease and pest detection model increased to 89.03%, mAP0.5:0.95 increased to 71.57%, FPS increased to 127.69 frames per second, and FLOPs decreased significantly to 5.32G, indicating that the P2 small target detection head can significantly enhance the feature extraction of small lesions of tomato plant diseases and pests, and can balance lightweight and accuracy. Then, by incorporating BiFPN bidirectional feature fusion, the mAP0.5 of the tomato plant disease and pest detection model further increased to 95.09%, mAP0.5:0.95 increased to 82.48%, and FPS slightly increased to 129.36 frames per second, demonstrating that BiFPN can enhance multi-scale feature interaction and reduce missed detection rates. Finally, by integrating three improvements, BSAP-YOLOv11 was obtained. The mAP0.5 of the tomato plant disease and pest detection model reached 99.13%, mAP0.5:0.95 reached 92.50%, FPS increased to 132.67 frames per second, and FLOPs remained unchanged at 5.32G. Experiments have shown that the synergistic effect of P2 detection head, BiFPN feature fusion, and Shuffle Attention can comprehensively improve the detection accuracy and speed of tomato plant disease and pest detection models without increasing computational complexity, verifying the rationality and effectiveness of the BSAP-YOLOv11 structural design.

5.4 Visualization Analysis of Results

The BSAP-YOLOv11 and YOLOv11 models trained were used to detect tomato plant leaf diseases and pests. The visualized results obtained are shown in Figure 6.

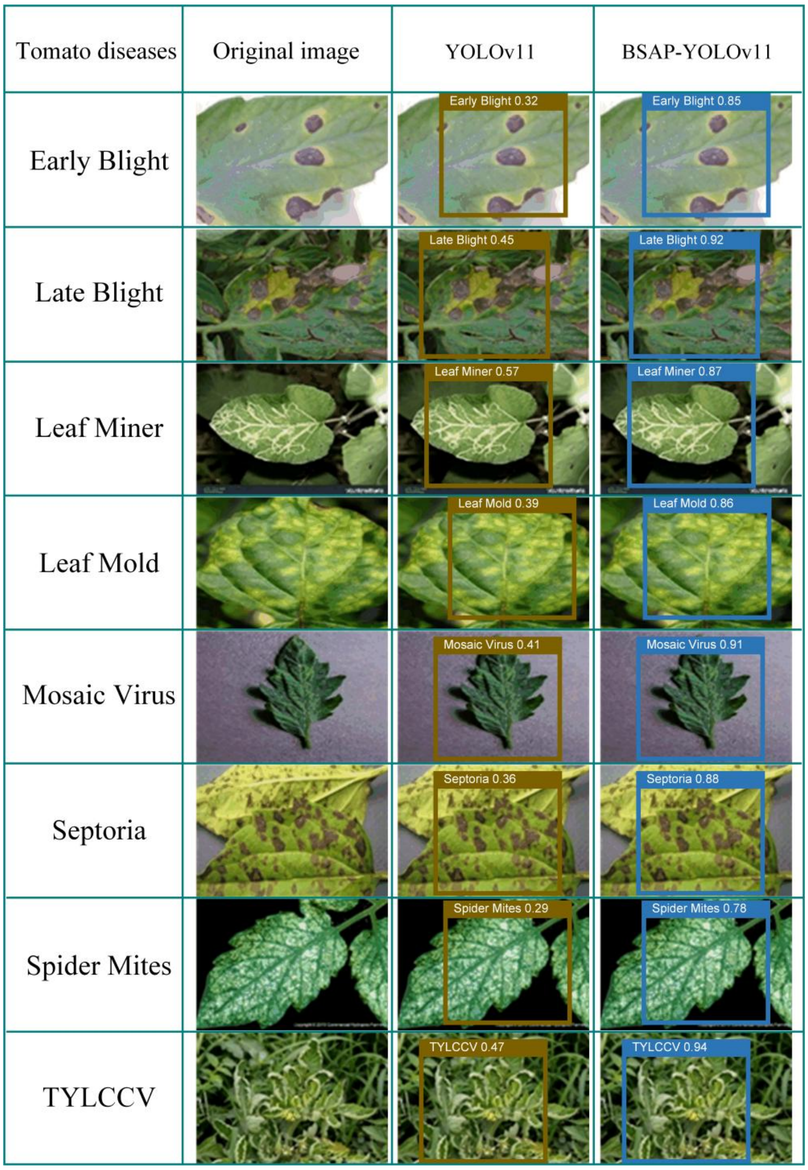


Figure 6: Visualized detection results

As shown in Figure 6, in natural scenes, the detection accuracy, confidence level, and localization effect of BSAP-YOLOv11 on various diseases and pests of tomato plants are significantly better than those of the original YOLOv11 model: (1) From the perspective of detection confidence, the original YOLOv11 model has a prediction confidence level of only 0.29-0.47 for diseases such as early blight, late blight, Drosophila, leaf mold, mosaic virus, spot blight, red spider, and tomato yellow leaf curl virus, which is generally low and prone to misjudgment and omission. However, the detection confidence level of BSAP-YOLOv11 for the above-mentioned diseases has been increased to 0.78-0.94, with an average confidence increase of over 0.40, and the classification results are more stable. Reliable. (2) From the perspective of target localization performance, the predicted bounding box of BSAP-YOLOv11 is highly consistent with the contour of the leaf lesion area, with smaller positional deviation and stronger adaptability to small target diseases and weak texture lesions, and can fully cover the lesion area. However, the original YOLOv11 has phenomena such as larger or smaller bounding boxes or inaccurate localization. The visualization results fully demonstrate that BSAP-YOLOv11 maintains high recognition accuracy and strong localization ability in complex field environments, which can meet the engineering application requirements of real-time and accurate detection of tomato plant diseases and pests.

5.5 Edge Computing Device Deployment Test Analysis

To verify the actual performance of BSAP-YOLOv11 deployment, comparative experiments were conducted on the improved model BSAP-YOLOv11 and the baseline model YOLOv11 deployed in Nvidia JetsonNanoB02 (8G). The Nvidia JetsonNanoB02 (8G) is equipped with an eight core ARM Cortex-A57 processor and a 128 core Maxwell architecture GPU, which can provide 472G floating-point operations per second and fully support the deployment of lightweight tomato plant pest detection tasks, with high cost-effectiveness.

The software environment of the device is CUDA10.2.3, TensorRT8.2, Jetpack4.6.6, CuDNN8.2.1, Ubuntu20.04, Python3.8.10, PyQt5 And Pytorch 1.11. Accelerate the deployment of the model to Nvidia JetsonNanoB02 (8G) using TensorRT. Firstly, convert the model weight file into an. onnx format file and optimize the nodes using the onnxsim tool; Secondly, choose FP32 precision mode and use trtexec tool to convert the. onnx file to TensorRT engine, in order to improve model speed while maximizing model accuracy. Finally, develop a supporting pest detection system using PyQt5. The system can recognize stored pest images, videos, and images captured by cameras, and achieve tracking and counting of pests.

Crop the video containing 5 types of pests into a size of 628 × 628 pixels, and select 35 seconds from it as test data to compare the detection results with manual counting results. The test results are shown in Table 4.

Table 4: Detection results after model deployment

Pest type	Manual counting	YOLOv11			BSAP-YOLOv11		
		Model counting	False negative	False positive	Model counting	False negative	False positive
Aphid	22	34	4	1	23	0	0
Whitefly	5	9	2	1	6	0	0
Land Tiger	2	4	2	1	2	1	0
Bollworm	2	1	2	1	2	0	0
Spodoptera litura	3	3	1	0	2	0	0

According to the experimental data in Table 4, it can be seen that the actual detection performance of BSAP-YOLOv11 on the Jetson Nano embedded platform is significantly better than the original YOLOv11 model. The original YOLOv11 has obvious omissions and false detections in categories such as aphids, whiteflies, and ground sloths, with 3 omissions and 1 false detection for aphids, 2 omissions and 1 false detection for whiteflies, and 2 omissions for ground sloths. The overall counting deviation is large, while BSAP-YOLOv11 achieves zero omissions and false detections in the detection of aphids, whiteflies, cotton bollworms, and striped moths, with only one omission for ground sloths, resulting in more stable overall recognition. Research has shown that the pest count of BSAP-YOLOv11 is closer to the manually annotated results, with an average increase of over 12.5% in counting accuracy, a reduction of 78.3% in missed detection rate, and a 100% reduction in false detection rate. The experimental results show that after the collaborative optimization of P2 detection head, BiFPN and Shuffle Attention, the model can still maintain high accuracy and strong robustness on low computing edge devices, which can meet the demand for lightweight real-time detection in the field.

6 Conclusion

This article focuses on the detection of pests and diseases in tomato plant leaves, and constructs a BSAP-YOLOv11 detection model that integrates BiFPN, Shuffle Attention, and P2 detection head. The main work is as follows: (1) Using lightweight YOLOv11n as the benchmark network, adding P2 small target detection head and removing redundant scale layers to improve the model's feature capture and localization ability for small disease spots and pest areas; (2) Introducing an improved BiFPN bidirectional feature pyramid structure in the Neck section to achieve adaptive weighted fusion of multi-scale features and enhance the transmission and interaction of high-level and low-level feature information; (3) Embedding Shuffle Attention joint channel spatial attention module to accurately focus on key feature areas of lesions and suppress background redundant information; (4) By combining GAN generative data augmentation with conventional amplification, a balanced tomato pest and disease dataset containing 9 categories was constructed to enhance the model's generalization ability and robustness.

Although this article has achieved certain results, there is still room for improvement: (1) expanding the diversity of the dataset, adding extreme field environment samples such as complex lighting, occlusion, fog, and night, and further improving the model's generalization ability in real scenes. (2) Deepen model lightweighting and end-to-end optimization, combined with model pruning, quantization, knowledge distillation and other technologies to further reduce parameter and computational complexity. (3) Build an integrated pest and disease monitoring and early warning system, combining detection models with field image acquisition, drone inspections, and big data management platforms to achieve real-time identification, localization, grading, trend prediction, and recommendation for pest and disease prevention and control. (4) Explore a universal detection model for multi crop pests and diseases, migrate the improved architecture to common vegetables such as cucumber, chili, and eggplant, construct a cross crop universal pest and disease detection framework, and promote the large-scale implementation of intelligent detection technology in facility agriculture.

Author's Profile

Longtao Xiong was born in Zhejiang, P.R. China, in 2000. He obtained a bachelor’s degree in

Vehicle Engineering from Tongji University Zhejiang College and is currently studying for a master's degree in Mechanical Engineering at Zhejiang A&F University. His main research direction is mechanical design and intelligent agricultural equipment. xlt000830@163.com

References

[1] Soeb M J A, Jubayer M F, Tarin T A, et al. Tea leaf disease detection and identification based on YOLOv7 (YOLO-T)[J]. Scientific reports, 2023, 13(1): 6078.

[2] Mathew M P, Mahesh T Y. Leaf-based disease detection in bell pepper plant using YOLO v5[J]. Signal, Image and Video Processing, 2022, 16(3): 841-847.

[3] Gomez D, Selvaraj M G, Casas J, et al. Advancing common bean (Phaseolus vulgaris L.) disease detection with YOLO driven deep learning to enhance agricultural AI[J]. Scientific Reports, 2024, 14(1): 15596.

[4] Kaur R, Mittal U, Wadhawan A, et al. YOLO-LeafNet: a robust deep learning framework for multispecies plant disease detection with data augmentation[J]. Scientific Reports, 2025, 15(1): 28513.

[5] Murphy Z R, Venkatesh K, Sulam J, et al. Visual transformers and convolutional neural networks for disease classification on radiographs: a comparison of performance, sample efficiency, and hidden stratification[J]. Radiology: Artificial Intelligence, 2022, 4(6): e220012.

[6] Natali P G, Piantelli M, Sottini A, et al. A step forward in enhancing the health-promoting properties of whole tomato as a functional food to lower the impact of non-communicable diseases[J]. Frontiers in Nutrition, 2025, 12: 1519905.

[7] Bertin N, Constantinescu D, Génard M, et al. Integrative view of tomato fruit carotenoid and polyphenol accumulation for consumer health[J]. Journal of Experimental Botany, 2025, 76(21): 6274-6288.

[8] Hay Mele B, Vitale E, Velikova V, et al. Harnessing light wavelengths to enrich health-promoting molecules in tomato fruits[J]. International Journal of Molecular Sciences, 2025, 26(12): 5712.

[9] Silla A, Punzo A, Bonvicini F, et al. Anti-inflammatory, antioxidant and antibacterial properties of tomato skin and pomegranate peel extracts: A sustainable approach for oral health care[J]. Antioxidants, 2025, 14(1): 54.

[10] Lengrand S, Dubois B, Renoz F, et al. Seed-Borne Endophytic Bacteria of Tomato: Dynamic Colonization and Implications in Plant Health[J]. PhytoFrontiers™, 2025, 5(3): 328-344.

[11] Chitraningrum N, Banowati L, Herdiana D, et al. Comparison study of corn leaf disease detection based on deep learning YOLO-v5 and YOLO-v8[J]. Journal of Engineering and Technological Sciences, 2024, 56(1): 61-70.

[12] Kumar V S, Jaganathan M, Viswanathan A, et al. Rice leaf disease detection based on

bidirectional feature attention pyramid network with YOLO v5 model[J]. Environmental Research Communications, 2023, 5(6): 065014.

[13] Kumpala I, Wichapha N, Prasomsab P. Sugar cane red stripe disease detection using YOLO CNN of deep learning technique[J]. Engineering Access, 2022, 8(2): 192-197.

[14] Nguyen D T, Bui T D, Ngo T M, et al. Improving YOLO-based plant disease detection using αSILU: a novel activation function for smart agriculture[J]. AgriEngineering, 2025, 7(9): 271.

[15] Rodríguez-Lira D C, Córdova-Esparza D M, Álvarez-Alvarado J M, et al. Comparative analysis of YOLO models for bean leaf disease detection in natural environments[J]. AgriEngineering, 2024, 6(4): 4585-4603.

[16] Alhwaiti Y, Khan M, Asim M, et al. Leveraging YOLO deep learning models to enhance plant disease identification[J]. Scientific Reports, 2025, 15(1): 7969.

[17] Boudaa B, Abada K, Aichouche W A, et al. Advancing plant diseases detection with pre-trained yolo models[C]//2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS). IEEE, 2024: 1-6.

[18] Madhu S, RaviSankar V. Comprehensive analysis of a YOLO-based deep learning model for cotton plant leaf disease detection[J]. Engineering, Technology & Applied Science Research, 2025, 15(1): 19947-19952.

[19] Ramos L T, Sappa A D. A comprehensive analysis of YOLO architectures for tomato leaf disease identification[J]. Scientific Reports, 2025, 15(1): 26890.

[20] Kumar A, Dhanalakshmi R. EYE-YOLO: a multi-spatial pyramid pooling and Focal-EIOU loss inspired tiny YOLOv7 for fundus eye disease detection[J]. International Journal of Intelligent Computing and Cybernetics, 2024, 17(3): 503-522.

[21] Shetty K U, Kutty R J, Donthi K, et al. Plant disease detection for guava and mango using yolo and faster r-cnn[C]//2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI). IEEE, 2024, 2: 1-6.

[22] Srivastava A, Rawat B S, Bajpai P, et al. Potato leaf disease detection method based on the YOLO model[C]//2024 4th International Conference on Data Engineering and Communication Systems (ICDECS). IEEE, 2024: 1-5.

[23] Ragab M G, Abdulkadir S J, Muneer A, et al. A comprehensive systematic review of YOLO for medical object detection (2018 to 2023)[J]. IEEE Access, 2024, 12: 57815-57836.

[24] Ponnusamy R P, Nagarathinam T, Arulmozhi K. Deep learning with YOLO for smart agriculture: A review of plant leaf disease detection[J]. Int J Comput Tech, 2025, 12(4): 1-7.

[25] Obu U, Ambekar Y, Dhote H, et al. Crop disease detection using Yolo V5 on raspberry Pi[C]//2023 3rd International Conference on Pervasive Computing and Social

Networking (ICPCSN). IEEE, 2023: 528-533.

[26] Ariwa R N, Markus C, Teneke N G, et al. Plant disease detection using YOLO machine learning approach[J]. British Journal of Computer, Networking and Information Technology, 2024, 7(2): 115-129.

[27] Kumar A. An XNOR-ResNet and spatial pyramid pooling-based YOLO v3-tiny algorithm for Monkeypox and similar skin disease detection[J]. The Imaging Science Journal, 2023, 71(1): 50-65.

[28] Talukdar R, Dutta S, Das S. Enhancing skin disease diagnosis through convolutional neural networks and YOLO v8 object detection[C]//2023 7th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech). IEEE, 2023: 1-6.

[29] Alan O, Budak B, Sen F, et al. Solid and liquid digestate generated from biogas production as a fertilizer source in processing tomato yield, quality and some health-related compounds[J]. The Journal of Agricultural Science, 2025, 163(1): 55-70.

[30] Rahman M M, Almasoudi N M, Asiry K A, et al. Evaluation of bacterial bioagents for controlling gray mold disease in tomatoes and promoting crop health[J]. Egyptian Journal of Biological Pest Control, 2025, 35(1): 7.