Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 12, No. 1, 2025

Cotton Pest Detection Method based on Improved YOLOv8

Xinyu Zhang *, Yaqi Li

Henan Polytechnic University, Jiaozuo Henan, 454150, China

*** Corresponding author:** Xinyu Zhang

Abstract: Accurate detection of cotton pests and diseases is critical for ensuring crop yield. Traditional methods relying on manual expertise suffer from inefficiency and poor robustness. This paper proposes an improved YOLOv8n model (RSS-YOLOv8n) by integrating a Triple Feature Encoding (TFE) module, Scale Sequence Feature Fusion (SSFF) module, Multi-level Feature Fusion (SDI) module, and SENet channel attention mechanism to enhance the detection capability for small targets in complex field environments. Experiments demonstrate that the improved model achieves a mean average precision (mAP) of 88.1% and a frame rate of 37.63 FPS on a self-built dataset (6,358 images), outperforming YOLOv8n by 1.6% mAP and surpassing Faster R-CNN by 8.5% mAP. This method provides an efficient solution for real-time pest and disease detection in agricultural scenarios.

Keywords: Object Detection; YOLOv8; Attention Mechanism; Multi-scale Feature Fusion; Agricultural Computer Vision.

1. Introduction

Cotton, as a crucial economic crop, plays a significant role in human production and national economies[1]. Among various challenges, cotton pests and diseases pose the most substantial threat to yield, causing severe economic losses. These pathogens and pests typically initiate infections from leaves before spreading throughout the entire plant, making timely and accurate identification of disease types from cotton leaves critical for effective prevention.

Traditional pest and disease control primarily relies on experienced growers or experts for visual inspection, which suffers from low recognition rates, visual fatigue-induced misjudgments, and other limitations. Recent advancements in computer vision and deep learning technologies have demonstrated remarkable performance of machine learning and deep learning models in crop disease detection systems[25]. Conventional machine learning methods automate disease detection by extracting texture, color, and shape features from images to construct classification models. While computationally efficient, these approaches inadequately represent spatial details of images and often neglect pixel-level information, resulting in sensitivity to local variations, poor robustness, and limited applicability in complex field environments.

Deep learning has shown significant potential in fine-grained plant phenotyping classification, yet challenges persist. For instance, Mohanty et al.[6] achieved 99% accuracy in classifying plant diseases on the PlantVillage dataset using GoogleNet and ResNet-101, but their models relied on high-quality images captured in controlled environments with limited adaptability to complex field conditions. Ferentinos[7] attained 99.53% accuracy with VGGNet for plant disease classification, yet the performance enhancement required extensive manually annotated data and failed to address class imbalance in small-sample categories. In object detection, Fuentes et al.[8] employed R-FCN with ResNet-50 to detect tomato diseases, achieving 85.98% mean average precision (mAP), but demonstrated insufficient sensitivity to occluded leaves and early-stage infections. Wang et al.[9] implemented disease region segmentation and quantification using Mask R-CNN, yet their approach's reliance on pixel-level annotations hindered large-scale deployment. Nagasubramanian et al.[10] combined genetic algorithms with SVM to select hyperspectral-sensitive bands, achieving 97% accuracy in early-stage soybean anthracnose detection, but the model's limited interpretability restricted its generalization to other crops.

Current research exhibits three primary limitations: 1) High annotation costs, particularly for multimodal data (e.g., hyperspectral and thermal imaging); 2) Weak model generalization with performance degradation across species and environments; 3) Insufficient recognition accuracy for subtle phenotypic features during early stress stages. Future directions should focus on few-shot learning, domain adaptation, and interpretable models to advance practical applications of plant phenotyping classification in complex field environments.

2. Materials and Methods

(1) Data Collection and Augmentation

Table 1. Types of cotton pests and diseases

Name	Description
Cotton Aphids	Body length <2 mm; secrete honeydew causing black sooty mold on leaves.
Bollworms	Larvae 40–50 mm long; initially bluish-gray, later yellowish or green.
Leaf Curl Disease	Leaf edges curl, veins thicken and protrude, forming cup-shaped lateral leaves.
Leaf Spot Disease	Dark red spots expand into irregular lesions with purple-red raised edges.
Red Leaf Stem Blight	Leaf margins turn yellow while veins remain green, or entire leaves turn reddish-brown.
Powdery Mildew	Thick white powdery fungal layer on leaves.
Healthy	Clear veins, vibrant green color, no curling.

Common pests and diseases during cotton cultivation include aphids, armyworms, leaf curl disease, leaf spot disease, red leaf stem blight, and powdery mildew. To comprehensively capture these conditions, images of cotton leaves infected with cotton aphids, bollworms, leaf curl disease, leaf spot disease, red leaf stem blight, powdery mildew, and healthy leaves were collected using multi-angle photography under varying weather and lighting conditions. A total of 1862 images were initially obtained. The characteristics of each pest/disease are detailed in Table 1.

To expand the dataset, 4,684 high-quality images of diseased and healthy cotton leaves were acquired via web crawlers. After deduplication and rigorous filtering, 6,358 images were retained as the experimental dataset. All images were resized to 640×640 pixels during preprocessing to mitigate resolution inconsistencies.

(2) Dataset Construction

To enhance model robustness, generalization, and resistance to label noise, the dataset was augmented through rotation, center cropping, Gaussian noise injection, and brightness adjustment. Manual annotation using LabelImg categorized the images into seven classes: Aphids, Armyworm, Leaf Curl, Leaf Spot, Wilt, Grey Mildew, and Health. Post-augmentation, the dataset expanded to 6,879 high-quality, resolution-uniform images. These were randomly split into training, validation, and test sets at an 8:1:1 ratio, completing the dataset construction.

3. Models and Training

(1) YOLOv8 Model Overview

The backbone of YOLOv8 adopts the CSPDarknet53 architecture, which includes multiple C2f modules influenced by the Cross Stage Partial (CSP) concept. The C2f module employs 3×3 convolutional kernels with a stride of 2 for feature extraction, effectively reducing the sampling rate. At the end of the backbone network, the Spatial Pyramid Pooling Fast (SPPF) module aggregates features from pooling windows of varying scales through concatenation, enhancing the network's perceptual capability. The neck incorporates a Feature Pyramid Network (FPN) combined with a Path Aggregation Network (PAN). The FPN enriches semantic information in low-level feature maps via a top-down pathway, while the PAN strengthens positional information in high-level feature maps through a bottom-up pathway. Their integration forms a bidirectional pyramid structure, improving multi-target detection capabilities. The head network generates feature maps corresponding to the number of target categories.

In cotton pest and disease detection tasks, YOLOv8 [7–9] achieves slightly higher accuracy compared to previous YOLO generations and the YOLOv10 model. YOLOv8 offers multiple model sizes (n, s, m, l, x), where the YOLOv8n model requires minimal memory footprint, making it more suitable for deployment on mobile devices for real-time pest detection. However, YOLOv8 still exhibits limitations in detecting small targets in complex environments, computational efficiency, feature extraction and fusion capabilities, and convergence speed, necessitating further improvements.

(2) Improved YOLOv8n-Based Cotton Pest and Disease Detection Model

To enhance recognition accuracy for cotton leaves in natural environments, this study modifies the YOLOv8n architecture as follows:

Triple Feature Encoding (TFE) Module: Introduces cross-scale fusion of large, medium, and small feature maps via concatenation to enhance the representation of microscopic targets.

Scale-Sequential Feature Fusion (SSFF) Module: Enhances multi-scale feature extraction by combining high-level semantic and low-level detailed information.

Multi-Level Feature Fusion (SDI) Module: Replaces the original Concat operation in the neck with a hierarchical feature integration mechanism.

SENet Channel Attention Mechanism: Integrated into the three detection heads (large, medium, small) to amplify critical channel features and suppress irrelevant ones, improving feature discriminability.

The modified architecture is illustrated in Figure 1.

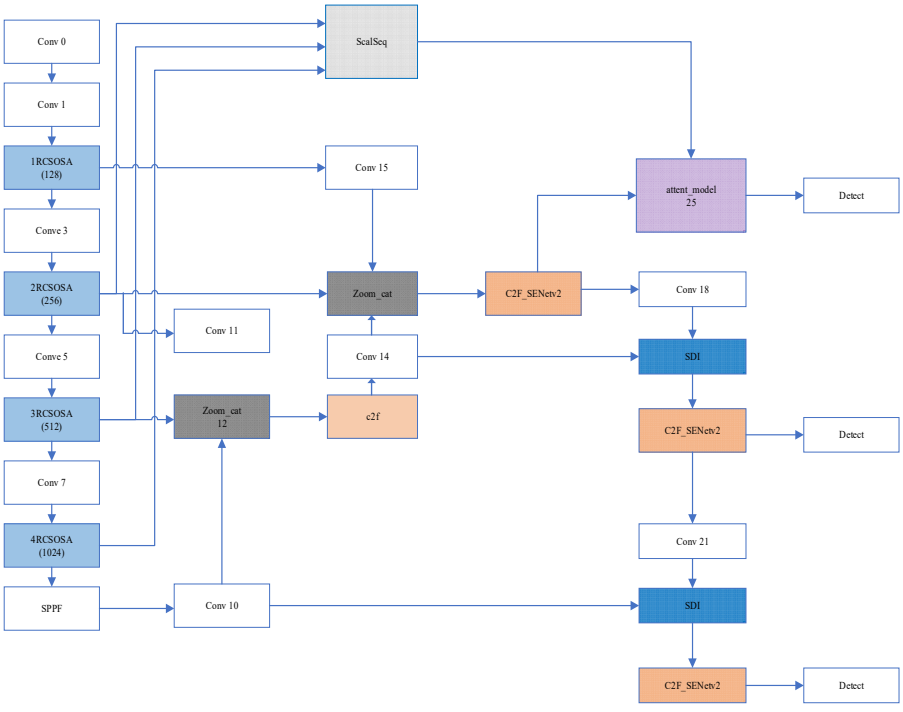

Fig 1. Improved Model Structure Diagram

Triple Feature Encoding (TFE):

For detecting dense small targets like cotton aphids, the original FPN's unidirectional top-down fusion path inadequately integrates high-resolution shallow features with fine-grained textures[13]. To address this, the TFE module divides backbone outputs into large, medium, and small spatial resolution groups. Through multi-granularity feature decoupling, it performs bidirectional interactions between high-resolution textures and deep semantic features via concatenation and cross-scale fusion. This process preserves edge sharpness and texture clarity for small targets. Before encoding, feature channels are adjusted to match the primary scale:

Large-scale features: Channel count reduced to 1c using hybrid max-average pooling downsampling.

Small-scale features: Channels adjusted via convolution and upsampled with nearest-neighbor interpolation Final outputs are concatenated channel-wise after convolution (Figure 2).

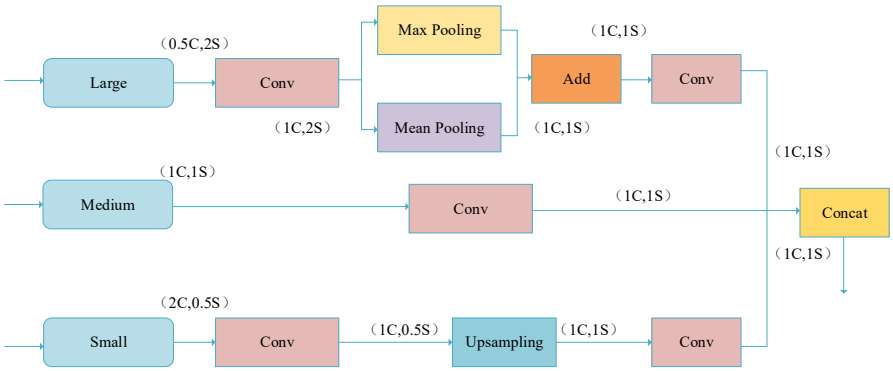

Fig 2. Triple Feature Encoding (TFE)

Scale-Sequential Feature Fusion (SSFF):

The original YOLOv8n model fuses pyramid features through simple concatenation or addition, which inadequately exploits correlations between multi-scale feature maps, leaving room for improvement in detecting cotton pests and diseases with inherent scale variations. The SSFF module (structure shown in Figure 3) addresses this by combining high-dimensional information from deep features with detailed information from shallow features:

Channel alignment: 1×1 convolutions adjust the channel numbers of P4 and P5 layers to match the P3 layer.

Resizing: Nearest-neighbor interpolation aligns feature map dimensions with P3.

Dimension expansion: The unsqueeze operation transforms 3D tensors [height, width, channels] into 4D tensors [depth, height, width, channels].

Depth-wise concatenation: 4D feature maps are concatenated along the depth dimension to form 3D feature maps.

Feature extraction: SiLU activation, 3D convolution, and 3D batch normalization complete the scale-aware feature fusion process.

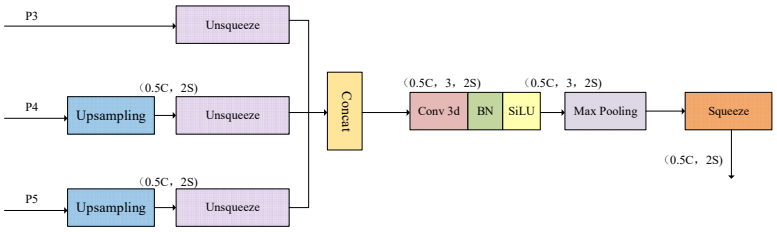

Fig 3. Scale-Sequential Feature Fusion (SSFF)

Multi-Level Feature Fusion (SDI):

The SDI module (structure shown in Figure 4) replaces the original Concat operation in the neck. Its core idea is to enhance semantic and detailed information by hierarchically integrating encoder-generated features. Together with the SSFF module, it forms the refined neck of the model[14].

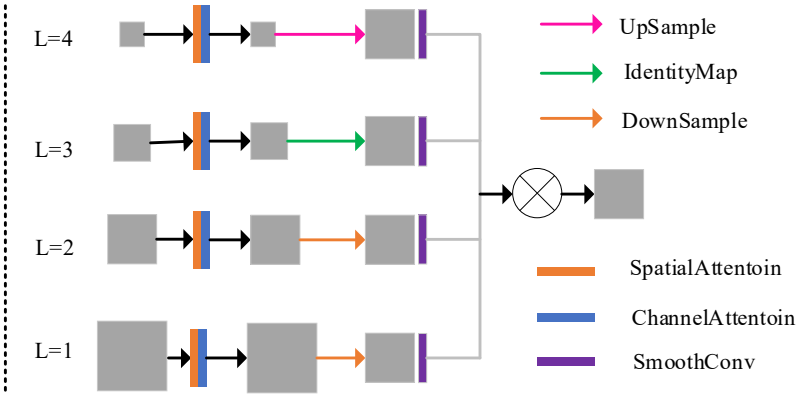

Fig 4. Multi-Level Feature Fusion (SDI)

SENet Channel Attention Mechanism (SENet):

When detecting small targets (e.g., cotton aphids, leaf spots) coexisting with larger ones, the original YOLOv8n model often suffers from missed detections due to insufficient extraction and filtering of critical features. To address this, the SENet module recalibrates channel weights by emphasizing important features and suppressing irrelevant ones.

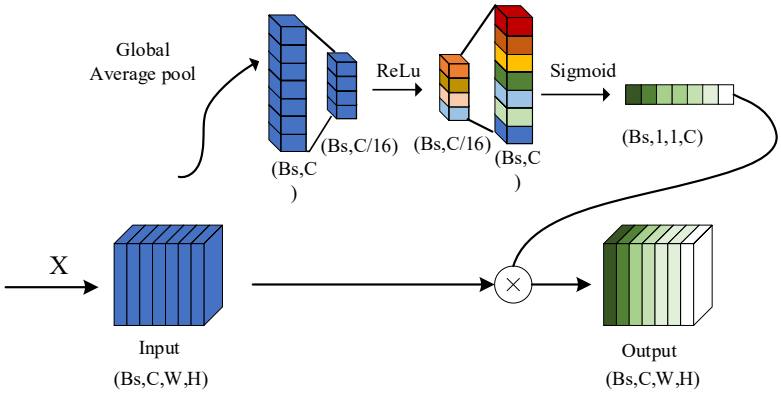

Fig 5. SENet Channel Attention Mechanism

As shown in Figure 5, SENet[15] operates in three stages:

Squeeze: Global average pooling compresses input features into a channel-wise vector of dimension *C*.

Excitation: Two fully connected layers capture channel dependencies, generating adaptive weights based on feature importance.

Recalibration: The original feature maps are scaled by the learned weights to amplify discriminative channels.

(3) Model Training and Evaluation Metrics

All experiments were conducted on a desktop with the configuration in Table 2.

Training Parameters: Input size: 640×640;Batch size: 24; Workers: 2; Initial learning rate: 0.01; Epochs: 1,000;Weight saving interval: 10 epochs (yielding 100 weight files); Pretraining: Initial 300 epochs without pretrained weights (due to modified backbone), followed by 1,000 epochs with trained weights.

Table 2. Experimental environment configuration

Configuration Name	Model/Version
System Environment	Ubuntu 20.04.6
Central Processing Unit (CPU)	Intel Core i5-12400 @4.4 GHz
Graphics Processing Unit (GPU)	NVIDIA GeForce RTX 3090 (24GB)
GPU Acceleration Library	CUDA 11.8
Random Access Memory (RAM)	32GB
Deep Learning Framework	PyTorch 2.0.1

Following Wang Xiangyu et al. [1],we adopt:

Precision (P, %): Ratio of correct predictions to total positive predictions:

$$P = \frac{T_P}{T_P + F_P} \times 100\% \quad (1)$$

Recall (R, %): Ratio of correctly detected positives to all actual positives:

$$R = \frac{T_P}{T_P + F_N} \times 100\% \quad (2)$$

Mean Average Precision (mAP, %): Average precision across all classes:

$$mAR = \frac{\sum_{i=0}^{N} \int_0^1 P(R) dR}{N} \times 100\% \quad (3)$$

Frames Per Second (FPS): Inference speed.

Here, Tp, Fp, and Fn denote true positives, false positives, and false negatives, respectively.N=7represents the number of disease/pest categories.

4. Results and Analysis

(1) Performance Comparison with Mainstream Models

The proposed RSS-YOLOv8n model is compared against mainstream object detection models, including YOLOv5, YOLOv7, YOLOv10, and Faster R-CNN. The results are summarized in Table 3.

Table 3. Comparison Between the Proposed Model and Mainstream Models

Model	Precision (P/%)	Recall (R/%)	mAP/%	Parameters (×10⁵)	FPS (frames/s)
Faster R-CNN	76.3	73.8	79.6	95.7	18.64
YOLOv5n	86.7	81.3	85.9	25.0	32.63
YOLOv7	84.6	82.8	86.0	36.7	36.88
YOLOv8n	85.8	83.7	86.5	31.5	36.74
YOLOv10n	86.0	81.5	86.1	26.7	35.45
Proposed Model	**88.8**	**83.2**	**88.1**	**33.9**	**37.63**

In comparisons with mainstream models, the proposed model achieves state-of-the-art performance in both precision and mean average precision (mAP). Specifically, it outperforms Faster R-CNN, YOLOv5n, YOLOv7, YOLOv8n, and YOLOv10n in precision by 12.5, 1.1, 4.2, 3.0, and 2.8 percentage points, respectively. For recall, the model exhibits improvements of 9.4, 1.9, 0.4, and 1.7 percentage points compared to Faster R-CNN, YOLOv5n, YOLOv7, and YOLOv10n, though it shows a slight 0.5 percentage point decrease relative to YOLOv8n. In terms of mAP, the proposed model surpasses Faster R-CNN, YOLOv5n, YOLOv7, YOLOv8n, and YOLOv10n by 8.5, 2.2, 2.1, 1.6, and 2.0 percentage points, respectively. Notably, the model significantly reduces parameter counts compared to Faster R-

CNN and YOLOv7-tiny while maintaining competitive complexity with other YOLO variants. Furthermore, it achieves superior inference speed, delivering 50.5%, 13.3%, 1.3%, 1.6%, and 5.0% improvements in frames per second (FPS) over Faster R-CNN, YOLOv5n, YOLOv7-tiny, YOLOv8n, and YOLOv10n, respectively. These results confirm that the proposed model effectively balances parameter efficiency with enhanced accuracy and real-time performance, addressing the dual challenges of computational overhead and detection precision in complex agricultural environments.

(2) Impact of Different Attention Mechanisms

To validate the effectiveness of the SENet channel attention mechanism, a comparative experiment is conducted with Deformable Attention (DAT), Convolutional Block Attention Module (CBAM), and Efficient Channel Attention (ECA). Results are shown in Table 4.

Table 4. Training Results of Models with Different Attention Mechanisms

Model	Precision (P/%)	Recall (R/%)	mAP/%	FPS (frames/s)
DAT	87.8	82.3	87.2	37.13
CBAM	86.4	81.9	86.4	34.72
ECA	87.5	82.7	87.5	36.79
Proposed	**88.8**	**83.2**	**88.1**	**37.36**

Experimental results demonstrate that SENet achieves superior comprehensive performance for cotton pest and disease detection:

Accuracy: SENet attains the highest precision (88.8%), recall (83.2%), and mAP (88.1%), outperforming DAT by 1.0%, 0.9%, and 0.9%, respectively. This validates the effectiveness of channel attention in enhancing small-target features.

Speed: SENet maintains real-time inference at 37.36 FPS. While CBAM suffers from computational overhead due to dual channel-spatial attention, ECA's lightweight design achieves 36.79 FPS but lags in mAP (-0.6%), reflecting its limited global modeling capability. DAT expands receptive fields through deformable attention but incurs multi-head computational complexity, constraining accuracy gains.

Overall, SENet optimally balances feature enhancement and computational efficiency, proving its suitability for multi-scale object detection tasks.

(3) Ablation Study

To evaluate the contribution of each proposed module, ablation experiments are conducted by incrementally adding TFE, SSFF, SDI, and SENet to the baseline YOLOv8n. Results are shown in **Table 5**.

Table 5. Ablation Study Results

Exp.	TFE	SSFF	SDI	SENet	mAP/%
1	—	—	—	—	86.5
2	√	—	—	—	87.1
3	√	√	—	—	87.3
4	√	√	√	—	87.5
5	√	√	√	√	88.1

Note: "√" indicates the use of this module;"—" indicates that this module is not used.

The ablation study results demonstrate that each module contributes progressively to model performance enhancement. The baseline model achieves a mAP of 86.5%. After introducing the TFE module, the mAP increases by 0.6% to 87.1%, validating the effectiveness of triple feature encoding in enhancing the representation of small targets through cross-scale concatenation. The subsequent addition of the SSFF module yields a marginal mAP improvement of 0.2%, indicating that standalone multi-scale feature fusion has limited impact on accuracy and requires synergistic integration with other modules. Replacing the standard Concat operation with the SDI module further boosts the mAP by 0.2% to 87.5%, proving that multi-level feature fusion optimizes cross-hierarchical information interaction. Finally, integrating SENet elevates the mAP to 88.1%, a 0.6% gain over Experiment 4, highlighting the critical role of the channel attention mechanism in feature weight recalibration.

Overall, TFE and SENet deliver the most significant gains, underscoring the importance of feature encoding and attention mechanisms for model optimization. The combined application of SSFF and SDI establishes a stable multi-scale feature enhancement pipeline. The nonlinear cumulative effects of module integration emphasize the necessity of functional complementarity among components during model refinement.

5. Conclusion

This paper addresses the challenge of small target detection in cotton pest and disease monitoring by proposing an improved YOLOv8n-based model. By integrating multi-scale feature fusion and channel attention mechanisms, the detection performance in complex field environments is significantly enhanced. Experimental results demonstrate that the improved model outperforms existing methods in precision (88.8%), recall (83.2%), and mean average precision (88.1%), while maintaining high real-time efficiency. Future work will focus on optimizing few-shot learning and model interpretability to further improve generalization and practical applicability.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] WANG Xiangyu, WEN Haojie, LI Xinxing, et al. Research progress analysis of mainly agricultural diseases detection and early warning technologies[J]. Transactions of the Chinese Society for Agricultural Machinery, 2016, 47(9): 266-277. (in Chinese with English abstract).

[2] REDMON J, FARHADI A. Yolov3: an incremental improvement [J]. arXiv preprint arXiv:1804.02767,2018.

[3] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934 ,2020.

[4] ZOU X, WW Z, ZHOU W, et al. YOLOX-PAI: an improved YOLOX, stronger and faster than YOLOv6[J]. arXiv preprintar Xiv: 2208.13040,2022.

[5] Li Z, Guo R, Li M, et al. A review of computer vision technologies for plant phenotyping[J]. Computers and Electronics in Agriculture, 2020, 176: 105672.

[6] Mohanty S P, Hughes D P, Salathé M. Using deep learning for image-based plant disease detection[J]. Frontiers in plant science, 2016, 7: 215232.

[7] Ferentinos K P. Deep learning models for plant disease detection and diagnosis[J]. Computers and electronics in agriculture, 2018, 145: 311-318.

[8] Fuentes A, Yoon S, Kim S C, et al. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition[J]. Sensors, 2017, 17(9): 2022.

[9] Jin X, Jie L, Wang S, et al. Classifying wheat hyperspectral pixels of healthy heads and Fusarium head blight disease using a deep neural network in the wild field[J]. Remote Sensing, 2018, 10(3): 395.

[10] Nagasubramanian K, Jones S, Sarkar S, et al. Hyperspectral band selection using genetic algorithm and support vector machines for early identification of charcoal rot disease in soybean stems[J]. Plant methods, 2018, 14: 1-13.

[11] Rehman Z U, Khan M A, Ahmed F, et al. Recognizing apple leaf diseases using a novel parallel real-time processing framework based on MASK RCNN and transfer learning: An application for smart agriculture[J]. IET Image Processing, 2021, 15(10): 2157-2168.

[12] Yang N, Qian Y, EL-Mesery H S, et al. Rapid detection of rice disease using microscopy image identification based on the synergistic judgment of texture and shape features and decision tree–confusion matrix method[J]. Journal of the Science of Food and Agriculture, 2019, 99(14): 6589-6600.

[13] Kang M, Ting C M, Ting F F, et al. ASF-YOLO: A novel YOLO model with attentional scale sequence fusion for cell instance segmentation[J]. Image and Vision Computing, 2024, 147: 105057.

[14] Peng, Yaopeng, Milan Sonka, and Danny Z. Chen. "U-net v2: Rethinking the skip connections of u-net for medical image segmentation." arxiv preprint arxiv:2311.17791 (2023).

[15] Narayanan, M. "SENetV2: Aggregated dense layer for channelwise and global representations. arxiv 2023."arxiv preprint arxiv:2311.10807..