Lightweight Visual Detection Framework for Complex Background Grape Leaf Disease Identification

Chenyu ZHAO
Inner Mongolia Minzu University
Mingyu BAI
Inner Mongolia Minzu University
Haoriqin WANG
Inner Mongolia Minzu University
Shanshan DANG
Inner Mongolia Minzu University
Chunyu PAN
Inner Mongolia Minzu University
Mingyue ZHANG
China United Network Communications Co., Ltd.
Shicheng QIAO
qiaosjo@imun.edu.cn
Inner Mongolia Minzu University

Research Article

Keywords: Disease Grape leaf, Grape leaf Image, recognition, YOLO11n, Lightweight

Posted Date: June 8th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9554582/v1

Additional Declarations: No competing interests reported.

Lightweight Visual Detection Framework for Complex Background Grape Leaf Disease Identification

Chenyu ZHAO[1], Mingyu BAI[1], Haoriqin WANG[1], Shanshan DANG[1], Chunyu PAN[1], Mingyue ZHANG[2], Shicheng QIAO[1*]

[1]School of Computer Science and Technology, Inner Mongolia Minzu University, Tongliao, 028043, China.
[2]China United Network Communications Co., Ltd., Tongliao, 028007, China.

*Corresponding author(s). E-mail(s): qiaosjo@imun.edu.cn;

Abstract

Accurate crop disease detection supports precision agriculture, but field-deployable identification remains hindered by complex backgrounds, varying illumination, and heavy deep learning models. This work presents a lightweight visual detection approach for grape leaf diseases under unconstrained field conditions. Built on the YOLO11n backbone, the method integrates three customized modules: C3k2-UltraLightBlock for efficient feature representation, LeafRepFusionStem for low-level feature enhancement, and RCSA-HSFPN for refined multi-scale fusion with residual channel-spatial attention. A dedicated dataset with complex backgrounds is constructed via augmentation and background replacement. Experiments show the model achieves 92.0% precision, 92.9% recall, and 93.0% mAP@0.5, with only 2.9 GFLOPs and 1.73 M parameters, representing 54.7% and 33.2% reductions over the baseline. Heatmap visualization confirms improved lesion focusing and background suppression, while cross-crop tests validate strong generalization. This framework provides an efficient solution for real-time, edge-deployable plant disease monitoring, balancing accuracy and computational efficiency for practical agricultural visual computing applications.The implementation code for this study is available at:https://github.com/aitizc/Lightweight-Visual-Detection-Framework-for-Complex-Background-Grape-Leaf-Disease-Identification.git

Keywords: Disease Grape leaf, Grape leaf Image, recognition, YOLO11n , Lightweight

1 Introduction

Grapes are an economically important fruit crop cultivated worldwide for fresh consumption, processing, and medicinal applications. In recent years, grape production in China has continued to increase, with an annual output exceeding 15 million tons and a stable annual consumption of approximately 10 million tons, making China one of the world's major grape producers and consumers [1]. As a leading grape-producing country [2], China has ranked first globally in grape production for many years [3]. Grape yield and quality are directly related to the stability of the industrial chain and the income of growers. However, grape leaf diseases, such as downy mildew, powdery mildew, and black rot, frequently occur during cultivation and remain a major biological constraint[4]. If not accurately detected and controlled in a timely manner, these diseases can rapidly spread throughout plants, disrupt photosynthesis, reduce yield, deteriorate fruit quality, and cause substantial economic losses [5].

Early studies on grape leaf disease detection mainly relied on traditional machine learning methods that extract handcrafted features, including color, texture, and shape, followed by classifiers such as support vector machines and random forests [6]. For example, PYDIPATI et al. utilized color and texture features for citrus disease identification, but the robustness of this approach was limited in complex environments [7]. Such methods are highly dependent on feature engineering, and when grape leaves are exposed to real-world field conditions—such as uneven illumination, occlusion by branches and fruits, soil background interference, and leaf overlap—the handcrafted features are easily corrupted by background noise. This results in reduced feature discriminability and a significant drop in detection accuracy, making these methods difficult to generalize to complex vineyard environments [8].

With the rapid development of deep learning, convolutional neural network (CNN)–based object detection methods have shown superior performance in plant disease detection tasks[9]. Representative approaches include two-stage detectors such as R-CNN [10], Fast R-CNN [11], and Faster R-CNN [12], as well as single-stage detectors such as YOLO and SSD [13]. Among them, the YOLO (You Only Look Once) series has attracted extensive attention due to its fast inference speed and strong real-time performance, and has been widely applied in plant disease detection scenarios [14]. Recent studies have attempted to enhance YOLO-based models by incorporating attention mechanisms, multi-scale feature fusion, and image reconstruction strategies. For instance, ZHU et al. introduced an efficient multi-scale attention module into the neck of YOLOv8, fully exploiting feature information across detection layers and achieving a grape leaf disease detection accuracy of 92.46% [15]. Zhang Linxuan et al. incorporated the BiFormer attention mechanism to sparsify attention computation through dual-layer routing, reducing computational complexity and improving detection accuracy by 1.8%, although the adaptability under complex disease scenarios remained limited [16]. BAO et al. proposed an improved YOLOv5-based method that combines RCAN-based super-resolution reconstruction with multi-scale RFB modules and the CBAM attention mechanism, achieving an mAP of 73.8%; however, this approach increases model complexity and computational cost, resulting in higher hardware requirements [17].

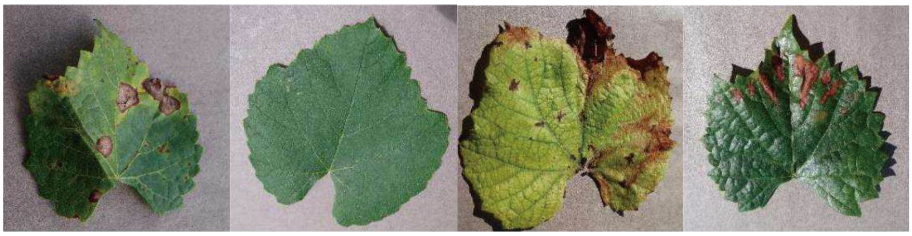

Fig. 1 The images in the PlantVillage dataset.

Despite the performance improvements achieved by these methods under specific conditions, several challenges remain for grape leaf disease detection in real-world vineyard environments. On the one hand, grape leaves are often distributed in complex natural backgrounds with occlusion from fruits and branches, soil interference, and severe leaf overlap, which can obscure disease features and lead to localization errors and missed detections. On the other hand, conventional YOLO models, such as YOLOv5 and YOLOv7, generally involve large parameter sizes and high computational complexity, making them difficult to deploy efficiently on mobile or embedded devices. Existing lightweight detection models often struggle to maintain satisfactory accuracy when disease features are partially occluded or visually ambiguous. Therefore, there remains a clear research gap in developing a lightweight yet robust detection framework that can achieve accurate and real-time grape leaf disease detection under complex field conditions.

To address these challenges, this study investigates grape leaf disease detection under complex backgrounds based on YOLO11n. By leveraging the inherent efficiency of YOLO11n and introducing task-oriented structural optimizations, the proposed method aims to improve detection accuracy and real-time performance while maintaining low computational cost. This work provides technical support for early disease monitoring and precision plant protection in vineyards, and offers valuable insights into the application of lightweight object detection models in plant disease detection.

2 Materials and Methods

2.1 Dataset construction

Common grape leaf diseases include black rot, leaf blight, downy mildew, anthracnose, rust, and esca. In this study, four types of leaf conditions—black rot [19], leaf blight, esca, and healthy leaves—were selected from the publicly available PlantVillage dataset [18]. The PlantVillage database contains grape leaf disease images covering different growth stages, multiple disease categories, and diverse scenarios [20], and is publicly accessible at https://github.com/spMohanty/PlantVillage-dataset.In addition, publicly available images were collected from agricultural research websites, plant disease repositories, and online plant health databases. These images were integrated to form a comprehensive dataset consisting of 3,854 images, with partial samples shown in Figure. 1.

Fig. 2 The images in the PlantVillage dataset.

As the collected images were primarily acquired under laboratory conditions and lacked the complex backgrounds typical of natural environments, background replacement was performed using Python to simulate grape leaf disease recognition under different weather conditions, viewing angles, and environmental settings, as illustrated in Figure. 2.Subsequently, data augmentation was applied to the background-replaced images using techniques such as flipping, mirroring, translation, and rotation. These operations effectively increased sample diversity, thereby enhancing the target detection algorithm's ability to learn annotated features and improving its generalization performance. Through multiple data augmentation strategies, a final dataset containing 10,000 grape leaf images was constructed, with 2,000 images per category, including black rot, downy mildew, leaf blight, esca, and healthy leaves.

The augmented images were precisely annotated using the Roboflow platform, and a complete labeling system was established based on the YOLO format [21]. The dataset includes five label categories corresponding to Black-rot, Downy-mildew, Leaf-blight, Esca, and Healthy. For dataset partitioning, a random allocation strategy was adopted, dividing the dataset into training, validation, and test sets at a ratio of 8:1:1, respectively.

2.2 YOLO11n Algorithm

YOLO11, released in September 2024, is a state-of-the-art general-purpose object detection algorithm [22]. Compared with its previous versions, YOLO11 incorporates several network and feature extraction optimizations, supporting multiple tasks including object detection, instance segmentation, pose estimation, and oriented bounding box detection [23–25]. The architecture introduces an improved C3k2 (Cross Stage Partial with kernel size 2) module to enhance feature propagation and gradient flow efficiency. In addition, the C2PSA (Convolution and Parallel Spatial Attention) module is integrated to improve the model's perception of small objects and complex backgrounds. The Neck of YOLO11 employs the SPPF (Spatial Pyramid Pooling–Fast) module to enable efficient multi-scale feature fusion.

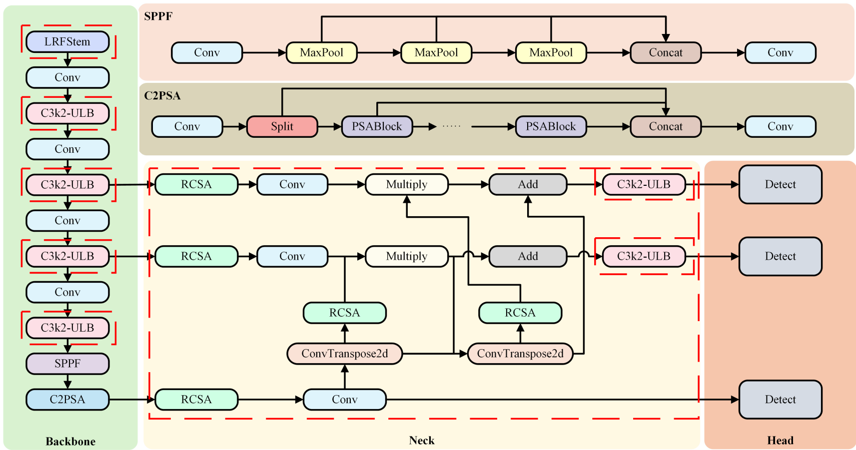

Fig. 3 Network structure diagram of the ULR-YOLO model.

YOLO11n is the lightweight variant of the YOLO11 series, designed specifically for resource-constrained or real-time detection scenarios. By inheriting the core advantages of YOLO11, YOLO11n further compresses the network depth and channel width, significantly reducing parameter count and computational cost while maintaining high detection accuracy. The model adopts a streamlined C3k2 backbone and a depth-wise separable convolution detection head. On the COCO dataset, YOLO11n achieves an mAP (50–95) of approximately 39.5%, with only 2.6 million parameters and 6.5 GFLOPs of computation. This balance between speed and accuracy makes YOLO11n particularly suitable for edge devices, mobile platforms, and real-time agricultural disease detection tasks [26–28].

2.3 ULR-YOLO Model

To further improve the lightweight performance and accuracy of grape leaf disease detection, this study proposes an enhanced network architecture based on YOLO11n, termed the ULR-YOLO model. First, an UltraLightBlock module was designed by integrating the C3k2 module to balance feature representation capability and model complexity. Second, a LeafRepFusionStem module was incorporated into the backbone network to exploit the expressive power of multi-branch convolutions for enhancing low-level feature representation, while ensuring efficient and lightweight inference. Finally, to address diseases with unique morphological characteristics, a RCSA-HSFPN pyramid network structure was added, strengthening the model's ability to detect irregularly shaped lesions. The overall structure of the ULR-YOLO model is illustrated in Figure. 3.

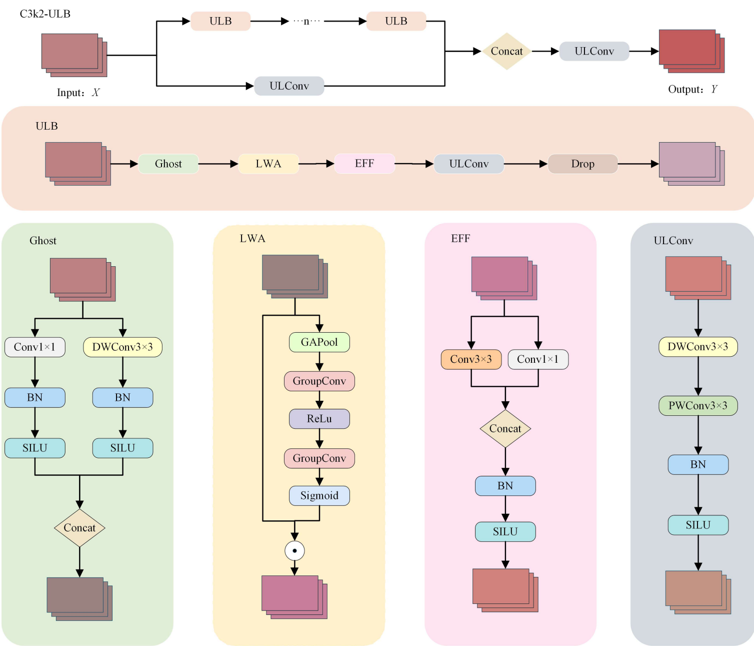

Fig. 4 Network Architecture Diagram of C3k2-UltraLightBlock.

2.4 C3k2-ULB Module

2.4.1 C3k2-UltraLightBlock (C3k2-ULB) Module

To further enhance the feature extraction capability and parameter efficiency of the YOLO11n model for grape leaf disease detection, a lightweight fusion module, termed C3k2-UltraLightBlock (C3k2-ULB), was designed based on the original C3k2 module. The C3k2-ULB module is built around three core design principles: efficient feature representation, lightweight computational structure, and multi-scale fusion. By integrating the Ghost feature generation mechanism, a lightweight attention mechanism, and multi-scale fusion structures, the module achieves a balance between feature expressiveness and model complexity.

The overall architecture of the C3k2-ULB module is illustrated in Figure. 4 and consists of three main components: (1) the UltraLightBlock unit, which performs feature enhancement and information compression; (2) a dual-path feature extraction structure, enabling parallel extraction of local and global features; and (3) a cross-layer fusion output layer, which implements residual connections and channel aggregation.

At the input, the module receives a feature mapX=RC×H×Wfrom the upper layer of the network. The input is processed through two lightweight convolutional paths. In one path, an ULConv (Ultra Lightweight Convolutional) module directly compresses the feature map along the CCC channel dimension. In the other path, the feature map passes through multiple UltraLightBlock layers for feature enhancement. The outputs of the two paths are then concatenated and fused using a 1×1 convolution, producing the output feature map Y.The overall computation of the C3k2-ULB module can be expressed as:

$$Y = \mathrm{ULConv}_{1\times1}\big([\mathrm{ULConv}_{1\times1}(X),\, \mathrm{ULB}(\mathrm{Conv}_{1\times1}(X))]\big) \tag{1}$$

Where ULB denotes the UltraLightBlock substructure.

To meet the real-time detection requirements of grape leaf diseases on mobile devices, this study proposes an Ultra Lightweight Convolution (ULConv) module based on standard depthwise separable convolutions. The ULConv module removes the bias term from convolutional layers entirely and leverages the affine parameters of batch normalization to achieve equivalent feature shifts. This design significantly reduces parameter count while improving model compressibility and deployment efficiency. In addition, the SiLU activation function replaces ReLU to enhance gradient propagation in lightweight networks.

ULConv adopts a minimal computational pipeline of depthwise convolution → pointwise convolution → batch normalization → SiLU activation, which reduces intermediate feature caching and memory access overhead. This design is well suited for the computational and memory constraints of mobile processors and provides an efficient feature extraction unit for real-time grape leaf disease detection under complex backgrounds.

The overall computation of ULConv can be expressed as:

$$Y = \mathrm{SiLU}\left(\gamma \odot \frac{W_P \cdot (W_d \cdot X) - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta\right) \tag{2}$$

In the equation, X denotes the input feature map;WP and Wd represent the bias-free depthwise and pointwise convolution kernels, respectively; $\odot$ denotes the convolution operation; μ and σ^2 are the batch normalization statistics; γ and β are the learnable affine parameters; and ε is the small constant for numerical stability in batch normalization.

2.4.2 UltraLightBlock Module

The UltraLightBlock is the core component of the C3k2-ULB module and incorporates four key mechanisms: redundant feature expansion, dynamic weight modulation, efficient feature fusion, and stochastic depth projection.

1. Redundant Feature Expansion: Inspired by GhostNet [29], a Ghost-like feature expansion is introduced to generate diverse features at minimal computational cost. The operation can be expressed as:

$$X_g = \mathrm{Concat}\big(\mathrm{SiLU}(X \cdot W_1),\, \mathrm{SiLU}(X \cdot W_2)\big) \tag{3}$$

   where X denotes the input feature map, W_1 is a 11 convolution kernel, W_2 is a depthwise convolution kernel, and X_g is the expanded feature map.
2. Dynamic Weight Modulation: To enhance the model's response to disease regions, a lightweight attention mechanism is applied for dynamic channel-wise feature weighting:

$$A = \sigma\big(W_3 \cdot \mathrm{ReLU}(W_3' \cdot \mathrm{GAP}(X_g))\big) \tag{4}$$

$$X_a = A \odot X_g \tag{5}$$

where A represents channel attention weights, GAP is global average pooling, W_3 denotes a grouped convolution kernel, σ is the Sigmoid function,$\odot$ denotes element-wise channel multiplication, and X_a is the attention-weighted feature map.

3. Efficient Feature Fusion: To capture both local and global receptive fields, a multi-scale fusion structure is designed using parallel 11 and 33 convolution branches. The fused features are then normalized and activated:

$$F = \psi\big([X_a \cdot W_{1\times1},\, X_a \cdot W_{3\times3}]\big) \tag{6}$$

where $W_{1\times1}$ and $W_{3\times3}$ are learnable convolution kernels, ψ denotes the combination of batch normalization and SiLU activation, and F is the fused feature map.

4. Output Mapping and Regularization: To further reduce computation while maintaining feature consistency, ULConv is applied at the output for dimensional mapping. Residual connections combined with stochastic depth (DropPath) [30] enhance model generalization:

$$Y_{\mathrm{ULB}} = X + \mathrm{DropPath}(\mathrm{ULConv}(F)) \tag{7}$$

where Y_{ULB} is the output feature map.

The C3k2-UltraLightBlock module leverages the Ghost+depthwise decomposition strategy and lightweight attention to effectively enhance the model's responsiveness to color anomalies and texture variations in diseased leaves. The efficient feature fusion structure enables the integration of multi-scale receptive fields, facilitating the recognition of lesions with diverse shapes and sizes.

2.5 LeafRepFusionStem Module

To improve the representation of low-level features in YOLO11n for grape leaf disease detection while maintaining lightweight and efficient inference, a novel feature extraction module, LeafRepFusionStem (LRFStem), is proposed, as illustrated in Figure. 5. This module employs multi-branch re-parameterized convolutions (RepBlock) to unify feature enhancement during training and lightweight inference during deployment.

The LRFStem module extracts spatial and channel information in parallel using 3×3 convolutions, 1×1 convolutions, and identity mapping branches, significantly enhancing low-level feature representation and improving sensitivity to fine-grained texture and edge variations of leaf lesions. During deployment, the branches are merged into a single $3{\times}3$ convolution via convolutional fusion, achieving substantial reductions in parameters and computation while preserving the feature representation learned during training.

To further enhance the edge information of lesion regions, the LRFStem module incorporates an edge enhancement operation based on Sobel convolution. The horizontal and vertical gradients of the feature maps are computed to inject local texture information into the network. These edge features are adaptively scaled using learnable parameters, effectively highlighting salient disease regions while ensuring Automatic

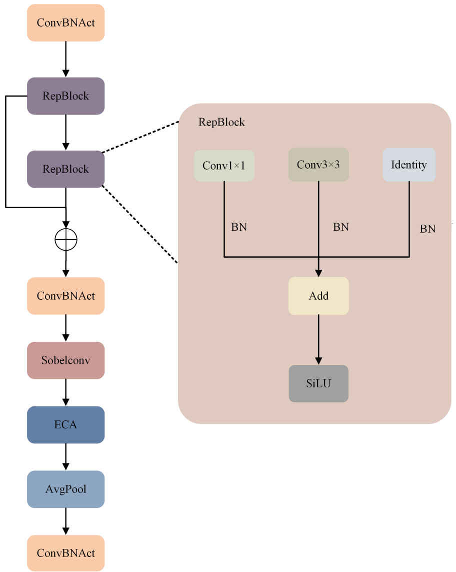

Fig. 5 LeafRepFusionStem Network Structure Diagram.

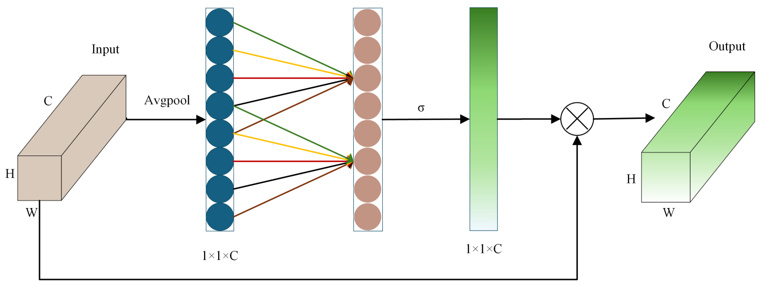

Fig. 6 LeafRepFusionStem Network Structure Diagram.

Mixed Precision (AMP) safety.Meanwhile, an efficient channel attention (ECA) mechanism [31] is embedded in the module to adaptively modulate the importance of each channel, enabling fine-grained cross-channel feature fusion. This allows the low-level features to retain spatial resolution while acquiring richer semantic representations, as illustrated in Figure. 6.

In addition, the LRFStem module introduces a shallow shortcut path, where the original input feature map is projected to the target output dimension through convolution and pooling operations. This design enables effective fusion of low-level spatial details with deeply enhanced semantic features, thereby preserving fine-grained structural information while improving discriminative representation capability. The overall computation process of the LRFStem module is formulated in Eqs.(8)–(14):

$$X_0 = \text{ConvBNAct}\,(I; W_{\text{init}}) \tag{8}$$

$$X_1 = \mathrm{RepBlock}_1\left(X_0\right) + \mathrm{RepBlock}_2\left(\mathrm{RepBlock}_1\left(X_0\right)\right) \tag{9}$$

$$X_2 = \mathrm{ConvBNAct}\left(X_1; W_{\mathrm{pw}}\right) \tag{10}$$

$$E = \alpha\sqrt{\left(X_2 \cdot K_x\right)^2 + \left(X_2 \cdot K_y\right)^2} \tag{11}$$

$$X_3 = X_2 \cdot \sigma\left(\mathrm{Conv}\left(\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} X_2(:,:,i,j)\right)\right) \tag{12}$$

$$X_4 = \mathrm{AvgPool}\left(X_3\right) \tag{13}$$

$$Y_{\mathrm{out}} = \mathrm{ConvBNAct}\left(X_4; W_{\mathrm{proj}}\right) + \mathrm{Shortcut}(I) \tag{14}$$

Here, I denotes the input image; K_x and K_y represent the horizontal and vertical Sobel convolution kernels, respectively. W_{init}, W_{pw} and W_{proj} denote the weight parameters of the current convolutional unit. α is a learnable edge-enhancement coefficient, and $\cdot$ indicates element-wise multiplication. $\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} X_2(:,:,i,j)$ denotes the channel descriptor obtained by global average pooling, σ represents the Sigmoid activation function, and Y_{out} is the output feature map.

Through this unified design for both training and deployment stages, the LRFStem module fully exploits the representational advantages of multi-branch convolutions while achieving efficient lightweight inference, thereby providing a robust and high-precision low-level feature foundation for grape leaf disease detection.

2.6 RCSA-HSFPN Module

CA-HSFPN[24] is a network architecture designed to efficiently address the challenges of multi-scale object detection. Its core modules achieve precise selection of feature maps at different scales through channel attention and dimension alignment mechanisms. In the feature selection stage, global average pooling and global max pooling are jointly employed with a weighting strategy to extract salient channel-wise information. In the feature fusion stage, a selective mechanism is adopted to integrate the filtered high-level and low-level features, where high-level features are resized via bilinear interpolation and transposed convolution to align with low-level features before fusion. This strategy significantly enhances feature representation capability and improves both the accuracy and efficiency of multi-scale detection.

Based on CA-HSFPN, a lightweight Neck network is constructed to maintain high detection accuracy while reducing the use of C2f modules, thereby effectively decreasing the number of parameters and computational cost. During model optimization, the original attention mechanism is replaced with a Residual Channel–Spatial Attention (RCSA) mechanism, which aims to adaptively emphasize channels and spatial locations that are more critical for disease region discrimination without disrupting the original feature distribution. As illustrated in Figure. 7, RCSA introduces a learnable scaling factor to flexibly regulate attention strength, thereby improving training stability and enhancing model generalization. The computation of the RCSA module is defined in Eqs. (15)–(18):

$$A_c = \sigma\big(f_c(\mathrm{GAP}(X)) + f_c(\mathrm{GMP}(X))\big), \tag{15}$$

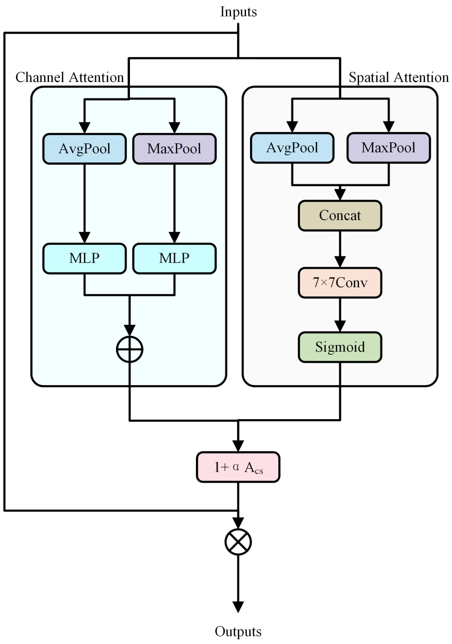

Fig. 7 RCSA Network Structure Diagram.

$$A_s = \sigma\big(f_s([\mathrm{mean}_c(X); f_s([\max_c(X)])])\big), \tag{16}$$

$$A_{cs} = A_c \cdot A_s, \tag{17}$$

$$Y = X \cdot (1 + \alpha A_{cs}), \tag{18}$$

In the above equations, X denotes the input feature map; GAP and GMP represent global average pooling and global max pooling, respectively; f_c denotes the channel mapping function; σ is the Sigmoid activation function; f_s denotes the spatial mapping function; $\cdot$ indicates element-wise multiplication; $mean_c$ and $\max_c$ denote channel-wise average and channel-wise maximum operators, respectively; and α is a learnable scaling coefficient.

In the channel dimension, global average pooling and global max pooling are jointly introduced to capture complementary statistical information, enabling the network to characterize the varying importance of different semantic channels for disease discrimination. In the spatial dimension, RCSA performs average and maximum aggregation along the channel axis to respectively model background distribution and salient disease-spot structures, thereby enhancing the network's sensitivity to the spatial locations of disease regions. The resulting joint channel–spatial attention response is incorporated through a residual scaling mechanism. This design ensures that, at the early stage of training, the network behavior remains close to an identity mapping, effectively preventing the attention module from introducing disruptive interference into the backbone feature extraction process. By balancing accuracy improvement and lightweight design requirements, the proposed RCSA mechanism is particularly well suited for grape leaf disease detection under complex background conditions

2.7 Evaluation Metrics

To comprehensively evaluate the performance of the proposed model in grape leaf disease detection and to verify the effectiveness of the introduced improvement strategies, multiple evaluation metrics are employed for systematic assessment. The primary metrics include Precision (P), Recall (R), Mean Average Precision (mAP), Giga Floating-point Operations Per Second (GFLOPs), and the number of model Parameters(Para). These metrics jointly reflect the model's overall capability in terms of detection accuracy and real-time computational efficiency. Among them, the specific calculation formulations of Precision, Recall, and Mean Average Precision are defined in Equations (19)–(21), respectively.

$$P = \frac{TP}{TP + FP} \times 100\% \tag{19}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{20}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} P_i(R)\, dR \times 100\% \tag{21}$$

where TP denotes the number of true positives, FP represents the number of false positives, TN is the number of true negatives, and FN indicates the number of false negatives. N denotes the total number of classes in the dataset.

3 Results

3.1 Experimental Environment

To ensure the rationality and fairness of the experiments, all tests were conducted under identical hardware, software, and configuration settings. The experimental hardware consisted of an Intel Xeon Platinum 8352V CPU @ 2.10 GHz and an NVIDIA RTX 4090 GPU with 24 GB memory. The software environment included Python 3.8.10, PyTorch 1.8.1, and CUDA 11.1.

The training parameters were set as follows: total number of training epochs=100, batch size=16, and number of worker threads=8, while all other parameters were kept at their default values. Figure. 8 illustrates the evolution of the loss functions over the 100 training epochs. As shown, all loss functions exhibit a clear decreasing trend with increasing epochs, indicating effective convergence of the training process.

3.2 Comparison of Attention Mechanisms

To validate the effectiveness of the proposed RCSA-HSFPN module, comparative experiments were conducted on the improved YOLO11n backbone. RCSA-HSFPN was compared with CA-HSFPN and CAA-HSFPN [32], and the results are summarized in Table 1.

The results indicate that RCSA-HSFPN achieves superior overall performance while maintaining computational complexity (2.9 GFLOPs) and parameter count

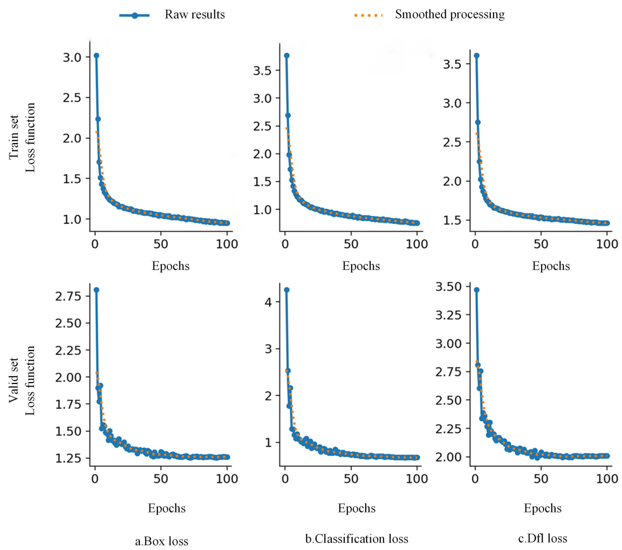

Fig. 8 Loss Function Variation Diagram.

Table 1 Comparison of attention mechanisms.

Model	P (%)	R (%)	mAP@0.5 (%)	FLOPs (G)	Params (M)
CA-HSFPN	91.8	92.4	92.0	2.9	1.64
CAA-HSFPN	91.7	92.2	92.7	3.1	1.83
RCSA-HSFPN	92.0	92.9	93.0	2.9	1.73

(1.7 M) comparable to CA-HSFPN. Notably, RCSA-HSFPN outperforms the heavier CAA-HSFPN in precision, recall, and mAP, highlighting its efficiency and enhanced multi-scale feature aggregation capability, which is beneficial for lightweight, high-precision disease detection in real agricultural scenarios.

3.3 Ablation Experiments

To evaluate the contribution of each proposed module, ablation studies were performed on YOLO11n using the augmented dataset. Results are summarized in Table 2.

The ablation results demonstrate that adding C3k2-ULB improves recall by 1.9% while reducing GFLOPs and parameters by 10.9% and 14.3%, respectively. Incorporating LRFStem further balances accuracy and efficiency, increasing mAP to 92.2% and reducing computational load by 53.1%. The full integration of RCSA-HSFPN achieves optimal overall performance: Precision = 92.0%, Recall = 92.9%, mAP = 93.0%, with GFLOPs and parameters reduced to 2.9 and 1.73 M, respectively, representing decreases of 54.7% and 33.2% compared to baseline YOLO11n. These results

Table 2 Ablation study.

C3k2-ULB	LRFStem	RCSA-HSFPN	P (%)	R (%)	mAP@0.5 (%)	FLOPs (G)	Params (M)
–	–	–	91.1	89.6	91.8	6.4	2.59
√	–	–	91.6	91.5	91.8	5.7	2.22
√	√	–	91.0	91.7	92.2	3.0	2.23
√	√	√	92.0	92.9	93.0	2.9	1.73

highlight the synergistic effect of the three modules in enhancing feature extraction, computation efficiency, and multi-scale feature fusion.

3.4 Model Comparison

To verify the effectiveness of the improved algorithm, comparisons were conducted with classical object detection models under the same experimental settings. Results are summarized in Table 3.

Table 3 Model comparison experiment.

Model	P (%)	R (%)	mAP@0.5 (%)	FLOPs (G)	Params (M)
SSD	91.6	90.0	90.2	61.0	23.88
Fast R-CNN	77.4	94.5	91.8	370.2	137.1
YOLOv5	87.5	88.7	89.8	16.0	7.03
YOLOv7	88.7	91.1	89.6	105.2	3.65
YOLOv8	90.6	92.1	92.2	8.2	3.01
YOLOv9-t	91.4	88.4	90.8	11.7	2.80
YOLOv10-n	90.9	90.7	91.9	8.4	2.70
YOLO11-n	91.1	89.6	91.8	6.4	2.59
ULR-YOLO	92.0	92.9	93.0	2.9	1.73

ULR-YOLO demonstrates significant advantages in both detection accuracy and computational efficiency. It achieves 92.0% precision, 92.9% recall, and 93.0% mAP, outperforming all comparison models. Notably, mAP is 0.8 percentage points higher than the second-best YOLOv8 (92.2%) and 1.2 percentage points higher than the baseline YOLO11n (91.8%). Computationally, ULR-YOLO requires only 2.9 GFLOPs and 1.73 M parameters, achieving reductions of up to 65.5% and 54.7% in GFLOPs and significantly lowering memory demands compared to other lightweight models. Traditional two-stage detectors, such as Fast R-CNN, show high recall but are computationally expensive and unsuitable for real-time deployment. Single-stage detectors, including SSD and YOLOv5/7/8/9t, optimize efficiency but fail to achieve the accuracy-efficiency balance of ULR-YOLO. These results highlight the model's superiority for real-time agricultural disease detection on mobile and edge devices.

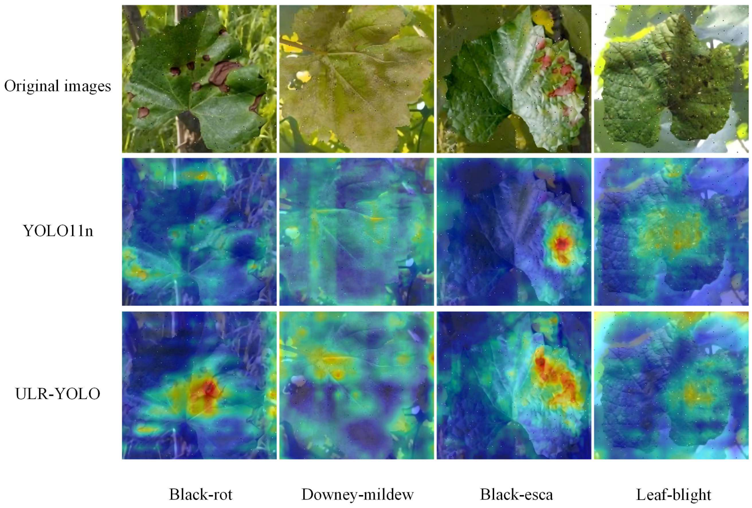

Fig. 9 Diagram of detection comparison results.

3.5 Heatmap Analysis

Figure. 9 illustrates the attention response comparison between YOLO11n and ULR-YOLO for four disease types: Black-rot, Downy-mildew, Esca, and Leaf-blight. The first row shows the original images under natural light and complex backgrounds, where lesion regions exhibit diverse shapes, blurred boundaries, and strong interference from healthy leaf textures and background elements. YOLO11n heatmaps (second row) show dispersed responses, with non-lesion areas also activated, indicating susceptibility to background noise. ULR-YOLO heatmaps (third row) display more concentrated attention, focusing on true lesion locations while suppressing non-target areas, providing clearer lesion contours. These observations confirm ULR-YOLO's ability to enhance discriminative lesion features and reduce background interference, improving localization consistency and robustness across disease types.

3.6 Model Generalization Test

To verify the detection accuracy of the ULR-YOLO model and the effectiveness of its improvements, five common crops—corn, potato, cherry, apple, and strawberry—were selected from multiple public dataset platforms such as PlantVillage and Kaggle. A total of 11,139 images were included, with shooting conditions covering different growth stages, lighting conditions, and multiple perspectives. This setup more realistically simulates the complexity and diversity of agricultural production environments. The corn dataset includes four categories: gray leaf spot, rust, blight, and healthy leaves. The potato dataset includes three categories: early blight, late blight, and healthy leaves. The cherry dataset includes two categories: powdery mildew and healthy leaves. The apple dataset includes four categories: black rot, rust, leaf spot, and healthy leaves. The strawberry dataset includes two categories: angular leaf spot and healthy leaves. The comparison results are shown in the table 4.

Table 4 Disease detection results on different crop leaf datasets.

Crop type	Model	P (%)	R (%)	mAP@0.5 (%)
Maize	YOLO11n	88.4	92.8	91.6
	ULR-YOLO	90.4	89.6	92.9
Potato	YOLO11n	88.8	90.9	88.4
	ULR-YOLO	90.1	92.2	88.9
Cherry	YOLO11n	85.3	86.0	84.4
	ULR-YOLO	87.9	88.9	86.8
Apple	YOLO11n	91.9	88.1	94.3
	ULR-YOLO	89.4	89.7	94.3
Strawberry	YOLO11n	91.9	94.2	91.3
	ULR-YOLO	92.3	93.8	91.4

The results demonstrate that ULR-YOLO consistently outperforms the baseline YOLO11n in most crops. For maize, mAP increases by 1.3 points, and for potato and cherry, both recall and mAP exceed the baseline. Although precision for apple is slightly lower, recall and mAP remain comparable or higher, showing robust cross-crop generalization. These findings confirm the proposed model's strong feature representation and generalization capability for complex, multi-crop agricultural disease detection.

4 Conclusion

This study constructed a grape leaf disease dataset with complex backgrounds and proposed the lightweight ULR-YOLO detection model based on the YOLO11n framework. The model addresses limitations of traditional methods in adapting to complex field conditions and achieving lightweight deployment. By integrating C3k2-ULB, LeafRep-FusionStem, and RCSA-HSFPN modules, ULR-YOLO balances detection accuracy and computational efficiency, achieving 93.0% mAP with only 1.73 M parameters and 2.9 GFLOPs, demonstrating strong feature extraction and resistance to background interference. Heatmap and cross-crop experiments confirm its precise lesion localization and robust generalization.

Compared with existing YOLO models, ULR-YOLO maintains high precision while significantly reducing computational load, making it suitable for deployment on mobile and edge devices. Future work could explore integrating multispectral data or dynamic adaptive mechanisms to enhance robustness under extreme lighting or occlusion conditions in real-world agricultural environments.

Declarations

- The authors have no relevant financial or non-financial interests to disclose.
- The authors have no competing interests to declare that are relevant to the content of this article.

- All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.
- The authors have no financial or proprietary interests in any material discussed in this article.

Ethics statement

This study does not involve human participants or animals. The research focuses on grape leaf disease detection using image data and deep learning methods; therefore, ethical approval was not required.

References

[1] Wu Canghai, Gu Xingxiang, Xiong Huanliang, and Huang Huixin. Fine-grained recognition of grape leaf diseases based on transfer learning and convolutional block attention module. *Applied Soft Computing*, 172:112896, 2025.

[2] Li Min, Yan Chenggong, Ma Shuai, Tan Haochao, Kang Yunyou, and Xu Liming. Current status and countermeasures of grape plant protection mechanization technology in china. *China Agricultural Mechanization Journal.*, 45(01):76–82, 2024.

[3] Li Xiaohong, Li Yunjing, Ma Xiaoqing, Guo Jun, Liu Haijiao, Zheng Guoqing, and Tao Jianmin. Developmentstatus and prospect of grape industry in china. *South ChinaFruits*, 50(05):161–166, 2021.

[4] Zhang Huanshuo and Ren Guobiao. Intelligent leaf disease diagnosis: image algorithms using Swin Transformer and federated learning. *The Visual Computer*, 41(7):4815–4838, 2025.

[5] Lin Wenqing and Li Shuqin. Grape leaf disease detection method based on improved rt-detr. *Transactions of the Chinese Society of Agricultural Engineering*, 41(15):145–154, 2025.

[6] He Dongjian, Wang Peng, Niu Tong, Mao Yanru, and Zhao Yanru. Field grape downy mildew severity grading model based on improved residual network. *Transactions of the Chinese Society of Agricultural Machinery*, 53(01):235–243, 2022.

[7] R Pydipati, TF Burks, and WS Lee. Identification of citrus disease using color texture features and discriminant analysis. *Computers and electronics in agriculture*, 52(1-2):49–59, 2006.

[8] Zhai Zhaoyu, Cao Yifei, Xu Huanliang, Yuan Peisen, and Wang Haoyun. Review on key technologies of crop pest and disease identification. *Transactions of the Chinese Society of Agricultural Machinery*, 52(07):1–18, 2021.

[9] Lu Huimin, Dong Bingwang, Zhu Bingxue, Ma Songzhe, Zhang Zexing, Peng Jianzhong, and Song Kaishan. A survey on deep learning-based object detection for crop monitoring: pest, yield, weed, and growth applications. *The Visual Computer*, 41(12):10069–10094, 2025.

[10] Carina Albuquerque, Roberto Henriques, and Mauro Castelli. Deep learning-based object detection algorithms in medical imaging: Systematic review. *Heliyon*, 11(1), 2025.

[11] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[12] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[13] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[14] Qiao Shicheng, Zhao Chenyu, Bai Mingyu, Dang Shanshan, Pan Chunyu, and Zhan Mingyue. Research progress on crop disease detection based on yolo algorithm. *Journal of Inner Mongolia Agricultural University (Natural Science Edition)*, 46(06):86–93, 2025.

[15] Jiajun Zhu, Jinlin Qiu, Shouwen Chen, Shuxi Chen, and Haifei Zhang. An application of yolov8 integrated with attention mechanisms for detection of grape leaf black rot spots. *PLoS One*, 20(4):e0321788, 2025.

[16] Zhang Linxuan, Bayintana, and Zeng Qingsong. Early disease detection method for grape leaves based on stylegan2-ada and improved yolo v7. *Transactions of the Chinese Society of Agricultural Machinery, volume = 55, number = 01, pages = 241-252, year = 2024, issn = 1000-1298,.*

[17] Wenxia Bao, Ziqiang Zhu, Gensheng Hu, Xingen Zhou, Dongyan Zhang, and Xianjun Yang. Uav remote sensing detection of tea leaf blight based on ddma-yolo. *Computers and Electronics in Agriculture*, 205:107637, 2023.

[18] Habiba N Ngugi, Absalom E Ezugwu, Andronicus A Akinyelu, and Laith Abualigah. Revolutionizing crop disease detection with computational deep learning: a comprehensive review. *Environmental Monitoring and Assessment*, 196(3):302, 2024.

[19] Zhe Tang, Jialing Yang, Zhe Li, and Fang Qi. Grape disease image classification based on lightweight convolution neural networks and channelwise attention. *Computers and Electronics in Agriculture*, 178:105735, 2020.

[20] Zhen Yuanyuan, Liu Ping, Jin Yuchun, and Yang Guohu. Grape leaf disease detection and recognition based on improved yolov8. *Transactions of the Chinese Society of Agricultural Engineering*, 41(14):148–154, 2025.

[21] Dang Shanshan, Qiao Shicheng, Bai Mingyu, Zhang Mingyue, Zhao Chenyu, Pan Chunyu, and Wang Guochen. Small object detection method for maize leaf diseases based on dcc-yolov10. *Smart Agriculture*, 7(05):124–135, 2025.

[22] Ranjan Sapkota, Marco Flores-Calero, Rizwan Qureshi, Chetan Badgujar, Upesh Nepal, Alwin Poulose, Peter Zeno, Uday Bhanu Prakash Vaddevolu, Sheheryar Khan, Maged Shoman, et al. Yolo advances to its genesis: a decadal and comprehensive review of the you only look once (yolo) series. *Artificial Intelligence Review*, 58(9):274, 2025.

[23] Enrico Mendez, Jesús Arturo Escobedo Cabello, Alfonso Gómez-Espinosa, Jose Antonio Cantoral-Ceballos, and Oscar Ochoa. Capsicum counting algorithm using infrared imaging and yolo11. *Agriculture*, 15(24):2574, 2025.

[24] Yifei Chen, Chenyan Zhang, Ben Chen, Yiyu Huang, Yifei Sun, Changmiao Wang, Xianjun Fu, Yuxing Dai, Feiwei Qin, Yong Peng, et al. Accurate leukocyte detection based on deformable-detr and multi-level feature fusion for aiding diagnosis of blood diseases. *Computers in biology and medicine*, 170:107917, 2024.

[25] Wenjing Xu, Jiachi Yao, Yanxue Wang, Chao Liu, Xinyu Liu, and Dongxiang Jiang. Wind turbine blade damage identification using lightweight yolo11 with multi-scale dilated attention. *Applied Soft Computing*, 185:114042, 2025.

[26] Tan Suiyan, Zhong Lei, Liu Changjiang, Wang Jie, Huang Junming, Hu Xihong, Wang Yuwei, Zheng Huiwen, Chen Xueshen, and Ma Xu. Design and experiment of a leafy vegetable plug seedling seeding performance detection system based on yolo11n. *Transactions of the Chinese Society of Agricultural Engineering*, pages 1–12.

[27] Zhang Zhe, Yu Helong, Yang Xinting, Luo Na, Li Shanshan, and Sun Chuanheng. Keypoint detection method for carapace of chinese mitten crab based on yolo-fmc-pose. *Transactions of the Chinese Society of Agricultural Engineering*, pages 1–12.

[28] R Yuvaraj, Sunil Arjun Bhalerao, Krishnan Murugesan, Suresh Vellaiyan, Nguyen Van Minh, et al. Real-time fire detection and suppression system using yolo11n and raspberry pi for thermal safety applications. *Case Studies in Thermal Engineering*, page 107159, 2025.

[29] Mercedes E Paoletti, Juan M Haut, Nuno S Pereira, Javier Plaza, and Antonio Plaza. Ghostnet for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(12):10378–10393, 2021.

[30] Jiajiao Chang, Honghui Li, Xueliang Fu, and Yuanyuan Jiao. Enet-caem: a field strawberry disease identification model based on improved efficientnetb0 and multiscale attention mechanism. *Frontiers in Plant Science*, 16:1701740, 2025.

[31] Panpan Gao, Yu Song, Minhui Song, Ping Qian, and Yanjing Su. Extract nanoporous gold ligaments from sem images by combining fully convolutional network and sobel operator edge detection algorithm. *Scripta Materialia*, 213:114627, 2022.

[32] Guo Jingbo and Huang Xiaohui. Mfdf-yolo: A lightweight cotton detection algorithm for complex scenarios. *Lightweight Cotton Detection Algorithm for Complex Scenarios*, pages 1–13.