LIGHTWEIGHT DETECTION OF VISIBLE CUCUMBER DOWNY MILDEW AND POWDERY MILDEW LESIONS UNDER GREENHOUSE CONDITIONS USING AN IMPROVED YOLOv11n

/

温室条件下基于改进 YOLOv11n 的黄瓜霜霉病与白粉病可见病斑轻量化检测

Qinyou SUN[1,2,3], Xingyu GAO[1,2,3], Fengyu LI[1,2,3], Xianyong MENG[1,2,3], Jun YAN[1,2,3, *], Pingzeng LIU[1,2,3, *]

[1] School of Information Science and Engineering, Shandong Agricultural University, Taian 271018, China;

[2] Key Laboratory of Huanghuaihai Smart Agricultural Technology, Ministry of Agriculture and Rural Affairs, Taian 271018, China;

[3] Agricultural Big Data Research Center, Shandong Agricultural University, Taian 271018, China

** Corresponding authors: E-mail: yanj2016@sdau.edu.cn; pzliu@sdau.edu.cn*

DOI: https://doi.org/10.35633/inmateh-78-112

Keywords: *cucumber, downy mildew, powdery mildew, lesion detection, greenhouse monitoring, YOLOv11n, edge deployment*

ABSTRACT

Downy mildew and powdery mildew pose significant threats to greenhouse cucumber production; however, accurate lesion detection on edge devices remains challenging due to the small size, high density, and frequent occlusion of lesions. This study proposes HSLG-YOLO, a lightweight detector based on YOLOv11n, which integrates CAA-HGNet, MPE-FPN, and TPL-Head for greenhouse cucumber lesion detection and edge deployment. Experimental results on a self-collected dataset show that the model achieves a precision of 93.58%, recall of 91.88%, mAP@0.5 of 96.49%, and mAP@0.5:0.95 of 67.89%. The model size is reduced from 5.3 MB to 4.6 MB, and it achieves 9.777 FPS on the Jetson Orin Nano platform, enabling near-real-time greenhouse disease monitoring.

摘要

霜霉病和白粉病威胁温室黄瓜生产，但由于病斑小、密集且易受遮挡影响，边端设备上的精准检测仍较困难。本文提出一种基于 YOLOv11n 的轻量化检测模型 HSLG-YOLO，融合 CAA-HGNet、MPE-FPN 和 TPL-Head，用于温室黄瓜病斑检测与边端部署。在自建数据集上，该模型的 Precision、Recall、mAP@0.5 和 mAP@0.5:0.95 分别达到 93.58%、91.88%、96.49% 和 67.89%。模型大小由 5.3 MB 降至 4.6 MB，并在 Jetson Orin Nano 上达到 9.777 FPS，可支持近实时温室病害巡检。

INTRODUCTION

Cucumber (*Cucumis sativus L.*) is an important crop in protected agriculture, and its stable production is closely related to effective disease management in greenhouse systems. Among cucumber foliar diseases, downy mildew and powdery mildew are two of the most common and economically important diseases because they spread rapidly and can seriously reduce leaf photosynthetic capacity, yield, and fruit quality (*Raufer et al., 2025*). In practical greenhouse production, timely and reliable detection of these two diseases is therefore essential for routine scouting and precision management. However, lesion detection under greenhouse conditions remains challenging because visible symptoms are often small, densely distributed, partially occluded, and affected by uneven illumination and complex backgrounds. Traditional diagnosis still relies heavily on manual observation and expert experience, which is labor-intensive, time-consuming, and easily influenced by subjectivity (*Liu and Wang, 2021*). Therefore, automatic and accurate lesion detection has important practical value for intelligent greenhouse disease monitoring (*Upadhyay et al., 2025*).

Deep learning has shown strong capability in agricultural disease analysis, especially for image-based recognition under complex visual conditions (*Abade et al., 2021; George et al., 2025; Zhao et al., 2025*). Among existing object detectors, the YOLO series has been widely adopted in agriculture because it provides a favorable trade-off between speed and accuracy in real-time applications (*Gupta et al., 2025; Wang et al., 2023; Wang, Yeh and Liao, 2024; Jocher et al., 2024*). Nevertheless, agricultural vision tasks still require robustness to occlusion, background interference, and domain variation, particularly when models are

expected to operate in real production environments rather than in simplified laboratory settings (*Rana et al., 2026*).

In cucumber disease analysis, recent studies have reported encouraging progress. *Li et al. (2022)* improved disease localization in natural scenes using YOLOv5, while *Cao et al. (2023; 2025)* explored multimodal and self-supervised strategies for small-sample cucumber disease recognition. *Qiao et al. (2025)* enhanced YOLOv5s for cucumber downy mildew spore detection, *Xie et al. (2025)* improved YOLOv8 for cucumber leaf pest and disease detection in natural backgrounds, and *Wang et al. (2024)* strengthened small-target perception in cucumber disease images. In addition, studies on lightweight agricultural vision models have shown that efficient backbones, attention mechanisms, and multi-scale fusion are helpful for improving detection performance while maintaining deployment efficiency (*Ma et al., 2024; Zeng et al., 2023; Hou et al., 2021; Qin et al., 2024*). These studies provide a useful basis for cucumber disease detection, but a gap still remains between high detection performance and lightweight deployment under real greenhouse conditions.

To address this gap, this study proposes HSLG-YOLO, an improved lightweight detector based on YOLOv11n for visible lesion detection of cucumber downy mildew and powdery mildew in greenhouse environments. Instead of emphasizing only accuracy improvement, the proposed method aims to balance detection performance, model compactness, and deployment potential for practical greenhouse scouting. Specifically, a CAA-HGNet backbone is introduced to improve feature extraction efficiency while reducing redundancy, an MPE-FPN neck is designed to strengthen multi-scale feature fusion for small lesions, and a TPL-Head is adopted to enhance localization of dense lesion targets. The proposed method was evaluated through comparative experiments, ablation studies, cross-dataset validation, and edge-device inference testing. This study aims to provide a practical and lightweight technical solution for automated greenhouse cucumber disease monitoring.

MATERIALS AND METHODS

Data Collection and Dataset Construction

To evaluate the proposed HSLG-YOLO model for cucumber disease detection under greenhouse conditions, a self-collected image dataset was constructed in this study. The dataset construction process included field image acquisition, data augmentation, and manual annotation. Particular attention was paid to preserving disease characteristics under practical greenhouse conditions, so that the collected images could reflect the complexity of real agricultural environments, including variations in viewpoint, illumination, and lesion appearance.

Data Acquisition

The samples used in this study were collected from cucumber planting bases in Shouguang City, Shandong Province, China. Following the recommendations of agronomic experts, cucumber disease images were acquired on the mornings of July 11, July 12, and December 14, 2025. To improve the diversity and representativeness of the collected data, image acquisition was conducted under different natural lighting conditions.

An iPhone 13 Pro Max was used as the image acquisition device. To maintain consistency during data collection, unified device settings were adopted, and both exposure and white balance were kept in automatic mode. A random shooting strategy was employed to simulate practical field inspection conditions. Therefore, no strict constraints were imposed on shooting direction or illumination during image collection. The shooting distance was controlled within a range of 0.2 to 1.0 m, and the viewing angles included top-down, eye-level, and bottom-up perspectives. This design helped the dataset cover diverse image characteristics that may be encountered in real greenhouse applications.

According to the disease characteristics observed in the planting bases, this study focused on two common cucumber foliar diseases: downy mildew (*Pseudoperonospora cubensis*) and powdery mildew (*Podosphaera xanthii*). Downy mildew lesions usually appeared as pale green or pale yellow angular spots, and a purplish-gray mold layer could often be observed on the abaxial leaf surface. Powdery mildew was characterized by white mildew spots on the leaf surface, which gradually expanded into powdery plaques covering larger areas of the leaf. These two diseases were selected because they are agriculturally important and highly relevant to practical greenhouse disease monitoring. These disease categories were identified on the basis of typical visible symptoms and confirmed during image collection with the assistance of agronomic personnel familiar with greenhouse cucumber disease scouting.

A total of cucumber leaf images were collected in this study. After removing blurred and duplicate images, 852 images were retained for subsequent experiments, including 532 images of downy mildew and

320 images of powdery mildew. All retained images were acquired under natural-light conditions, which helped preserve the authenticity and practical relevance of the dataset. Representative examples of the collected cucumber disease images are shown in Fig. 1.

Fig. 1 - Representative images of cucumber downy mildew and powdery mildew collected under greenhouse conditions

Dataset Partitioning and Data Augmentation

After the original images were collected and screened, dataset partitioning was first performed at the image level to reduce the risk of cross-subset data leakage. Specifically, the 852 retained original images were divided into training, validation, and test subsets at a ratio of 7:2:1, resulting in 596 training images, 170 validation images, and 86 test images. Data augmentation was then conducted independently within each subset rather than before dataset partitioning. In this way, all augmented variants derived from the same original image were retained within the same subset, thereby preventing overlap among the training, validation, and test sets.

To enhance sample diversity and improve model robustness under practical greenhouse conditions, several augmentation operations were applied, including horizontal flipping, vertical flipping, diagonal flipping, 90° rotation, 270° rotation, brightness adjustment, contrast adjustment, and Gaussian noise injection. These operations were designed to simulate variations in viewpoint, illumination, and imaging disturbance commonly encountered in greenhouse environments. After augmentation and secondary quality control, the training, validation, and test sets contained 5,744, 1,641, and 820 images, respectively, yielding a total of 8,205 images. This augmentation strategy increased the diversity of image appearances while preserving the original subset boundaries, thereby supporting subsequent model training and evaluation under a more robust and reproducible data organization scheme. The effects of different augmentation operations are illustrated in Fig. 2.

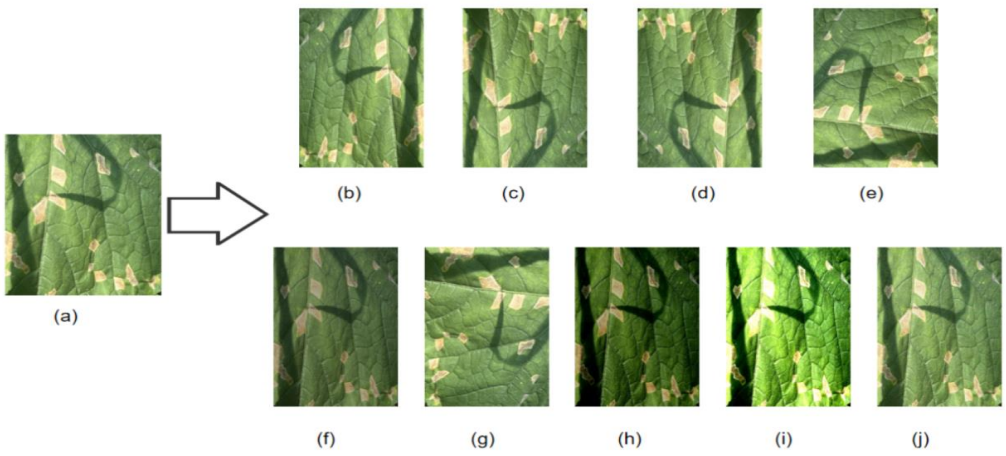


Fig. 2 - Data augmentation effects: (a) original image; (b) horizontal flip; (c) vertical flip; (d) diagonal flip; (e) 90° rotation; (f) random contrast adjustment; (g) 270° rotation; (h) random brightness adjustment; (i) contrast enhancement; (j) Gaussian noise.

Dataset Preparation

The collected cucumber disease images were manually annotated on the Roboflow platform using rectangular bounding boxes, and the annotations were exported in YOLO format for subsequent training and evaluation. In this study, image counts and annotation counts are explicitly distinguished to avoid ambiguity. At the image level, the dataset consisted of 852 screened original images, which increased to 8,205 images after within-subset data augmentation.

At the annotation level, each image may contain multiple lesion regions. Therefore, the total number of annotated instances was substantially larger than the number of images. As shown in Table 1, the dataset contains 247,986 annotated lesion instances in total, including 146,692 downy mildew instances and 101,294 powdery mildew instances. Specifically, the training, validation, and test sets contain 170,907, 49,515, and 27,564 annotations, respectively. For downy mildew, the numbers of annotated instances in the training, validation, and test sets are 101,101, 29,303, and 16,288, respectively. For powdery mildew, the corresponding numbers are 69,806, 20,212, and 11,276, respectively.

Table 1

Number of annotated disease instances in the dataset.

Disease Type	Training Set	Validation Set	Test Set	Total
Downy mildew	101,101	29,303	16,288	146,692
Powdery mildew	69,806	20,212	11,276	101,294
Total	170,907	49,515	27,564	247,986

Network Architecture and Lightweight Improvement Methods

To improve cucumber disease detection performance under greenhouse conditions while maintaining model compactness, this study develops HSLG-YOLO based on YOLOv11n. The overall framework follows the conventional backbone–neck–head architecture, while introducing targeted modifications for lightweight feature extraction, multi-scale feature fusion, and small-lesion localization. Specifically, the proposed model incorporates a CAA-HGNet lightweight backbone, an MPE-FPN neck, and a redesigned detection head. The detailed design of each component is described in the following subsections.

YOLOv11n Network Modeling

YOLOv11n is a lightweight variant of the YOLO11 family and serves as the baseline model in this study. Inheriting the real-time and efficient advantages of the YOLO series, it achieves improvements in both accuracy and computational efficiency through optimized network structure and module design. The algorithm mainly consists of four core parts: the input stage, the backbone network, the neck network, and the detection head.

The input stage is responsible for image reception and preprocessing, which improves the robustness and generalization ability of the model through size normalization, pixel normalization, and data augmentation. The backbone network introduces the improved C3k2 module and the C2PSA attention mechanism, combined with the optimized SPPF module, to enhance multi-scale feature extraction under a lightweight design. The neck adopts an improved PAN-FPN architecture to efficiently aggregate features from different levels and improve target localization accuracy. The detection head adopts a decoupled multi-task optimization strategy and integrates depthwise separable convolution and the DFL mechanism into the classification branch, thereby balancing detection accuracy and inference efficiency while facilitating deployment across different hardware platforms.

Although YOLOv11n already provides good efficiency, its direct application to cucumber disease detection in greenhouse environments still faces several limitations. Lesions are often small, densely distributed, and irregularly shaped, and they are easily affected by illumination variation, leaf overlap, and complex background textures. Therefore, task-oriented improvements are required to further enhance the model's suitability for cucumber disease detection.

CAA_HSFPN Attention Mechanism

Conventional large-kernel convolutions are effective for global context modeling, but their parameter cost and computational burden increase rapidly with kernel size, which limits lightweight deployment. To address

this issue, this study introduces CAA_HSFPN, a lightweight attention module based on the Cross-Axis Attention (CAA) framework. Its main idea is to decompose the two-dimensional receptive field of large-kernel convolution into horizontal and vertical one-dimensional modeling processes, thereby reducing complexity while preserving long-range dependency modeling.

For an input feature map $X \in \mathbb{R}^{B\times C\times H\times W}$, CAA_HSFPN first applies 7×7 average pooling, followed by a 1×1 convolution, horizontal and vertical depthwise strip convolutions, and a final 1×1 convolution with sigmoid activation to generate the attention map. The reweighted output is formulated as

$$\mathrm{Y} = \sigma\left(\mathrm{Conv}_{1\times1}\left(\mathrm{DWConv}_{\mathrm{k_v}\times1}\left(\mathrm{DWConv}_{1\times\mathrm{k_h}}\left(\mathrm{Conv}_{1\times1}(\mathrm{AvgPool}_{7\times7}(\mathrm{X}))\right)\right)\right)\right)\odot \mathrm{X} \quad (1)$$

where $\mathrm{AvgPool}_{7\times7}(\cdot)$ denotes average pooling, $\mathrm{Conv}_{1\times1}(\cdot)$ denotes pointwise convolution, $\mathrm{DWConv}_{1\times\mathrm{k_h}}$ and $\mathrm{DWConv}_{\mathrm{k_v}\times1}$ denote horizontal and vertical depthwise strip convolutions, $\sigma(\cdot)$ denotes the sigmoid function, and $\odot$ denotes element-wise multiplication. For simplicity, batch normalization and the SiLU activation function are omitted from Equation (1).

The convolutional parameter count of the module can be expressed as

$$\mathrm{P_{conv}} = 2\mathrm{C}^2 + \mathrm{Ck_h} + \mathrm{Ck_v} \quad (2)$$

where $2C^2$ corresponds to the two 1×1 convolutions, and $Ck_h + Ck_v$ corresponds to the two cross-axis depthwise strip convolutions. Compared with standard large-kernel convolution, this design reduces parameter cost and improves inference efficiency. Because it models long-range dependencies in both horizontal and vertical directions, CAA_HSFPN is well suited to capturing the directional boundaries of downy mildew lesions and the dense small-target characteristics of powdery mildew lesions. The overall structure is shown in Fig. 3.

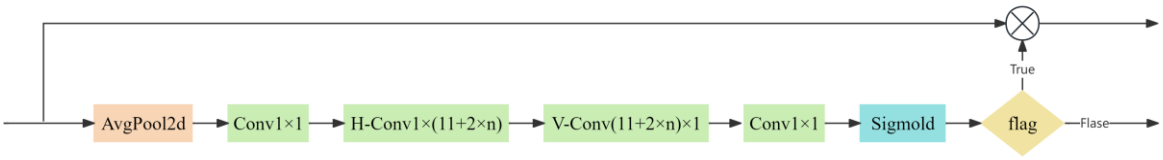


Fig. 3 - CAA_HSFPN attention mechanism.

Lightweight Cucumber Disease Detection Method Based on HSLG-YOLO

Based on YOLOv11n, this study develops HSLG-YOLO as a lightweight detector for visible lesion detection of cucumber downy mildew and powdery mildew under greenhouse conditions. As shown in Fig. 4, the model is modified at the backbone, neck, and detection head levels to address three practical difficulties: insufficient feature extraction under lightweight constraints, inadequate cross-scale fusion of small lesions, and inaccurate localization of dense lesion targets.

Specifically, the original backbone is replaced by CAA-HGNet to improve feature extraction efficiency and reduce structural redundancy. MPE-FPN is introduced to enhance the transmission and fusion of lesion features at different scales. TPL-Head is designed to improve localization sensitivity by using complementary P3-level features. Therefore, Fig. 4 not only presents the overall architecture, but also reflects the task-oriented optimization of HSLG-YOLO for small, densely distributed lesions under complex greenhouse backgrounds. Compared with the original full-scale detection strategy of YOLOv11n, HSLG-YOLO places greater emphasis on dense small-lesion detection while maintaining model compactness and deployment suitability.

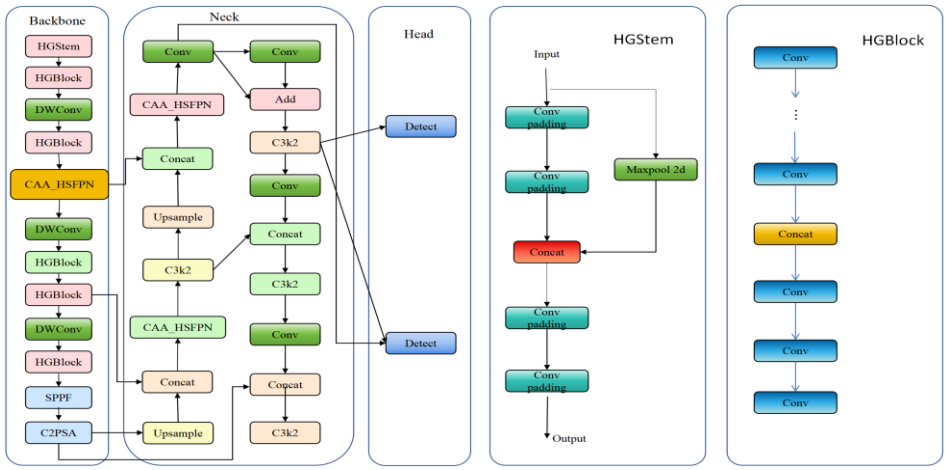


Fig. 4 - Enhanced YOLOv11n network architecture.

CAA-HGNet

To balance feature extraction capability and model lightweightness, this study designs the CAA-HGNet lightweight backbone network.

Based on the HGNetv2 architecture, this network introduces the CAA_HSFPN coordinate attention module to enhance spatial feature expression and integrates a global attention mechanism at the end to improve semantic understanding capability, with its overall architecture illustrated in Fig. 5.

CAA-HGNet adopts a non-uniform depth configuration, deploying 4, 4, 7, and 3 HGBlock modules at the P2, P3, P4, and P5 scales, respectively, to adapt to the detection requirements of targets at different scales. The P3 layer, which is a key stage for small-target detection, embeds the CAA_HSFPN module after four HGBlocks, separating attention calculations in the horizontal and vertical directions to preserve spatial information. The P4 layer enhances medium-scale features through grouped deployment of "4 + 3" HGBlocks. The P5 layer integrates SPPF and C2PSA after three HGBlocks to expand the receptive field and improve global semantic understanding.

The total parameter count of CAA-HGNet is calculated as shown in Equation:

$$P_{CAA-HGNet} = \sum_{i=2}^{5}(N_i \times P_{HGBlock}) + P_{CAA} + P_{SPPF} + P_{C2PSA} \quad (3)$$

In this formula, N2=4, N3=4, N4=7, N5=3, and the remaining terms denote the parameter quantities of the corresponding modules. This design enables the backbone to balance feature representation and model compactness, making it suitable for lightweight cucumber disease detection.

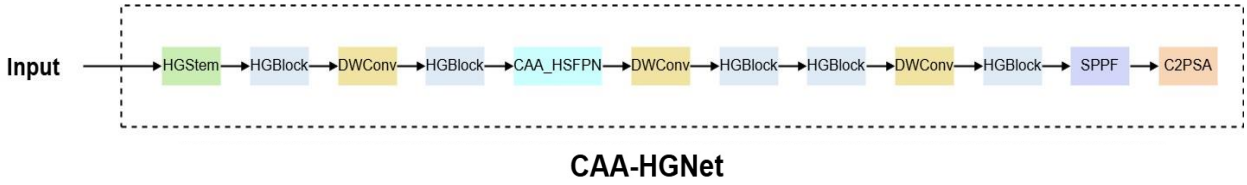


Fig. 5 - Overall architecture of CAA-HGNet

MPE-FPN Multi-Scale Feature Pyramid

This study constructs the MPE-FPN neck network to improve the multi-scale feature fusion capability, targeting the significant size variation of cucumber leaf disease spots. The module adopts a bidirectional feature pyramid design to achieve cross-scale feature interaction through top-down and bottom-up information flows, with its overall architecture shown in Fig. 6.

In the top-down pathway, the MPE-FPN integrates CAA_HSFPN attention modules at the P3 and P4 layers to optimize the attention weighting of fused features. The feature fusion process of the top-down pathway can be expressed as:

$$\begin{cases} T_4 = CAA(Fuse(F_4, Up(F_5))) \\ T_3 = CAA(Fuse(F_3, Up(T_4))) \end{cases} \quad (4)$$

where:

$Up(\cdot)$ denotes the upsampling operation, and $Fuse(\cdot)$ denotes the feature fusion operation.

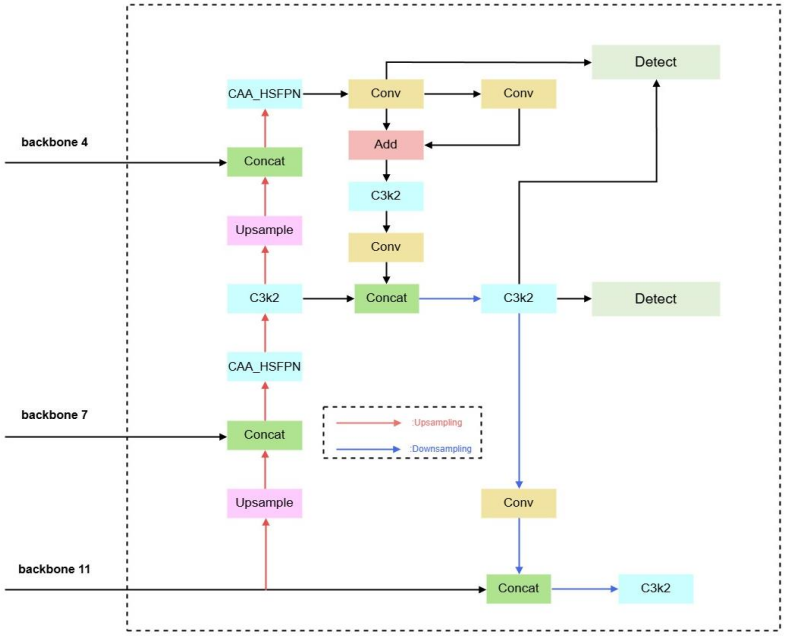
After attention enhancement, the P4 layer is processed twice by the C3k2 module, which is stronger than the single iteration used in the standard FPN. To meet the requirement of small-target powdery mildew detection, this study designs a dedicated enhancement module at the P3 layer. Specifically, a 3×3 convolution with 256 channels is first used to obtain local contextual information. Then, a 1×1 convolution with 256 channels is applied for feature fusion. After that, a residual connection is introduced to preserve the original feature information. Finally, three successive C3k2 modules are used to obtain the refined P3 feature, which can be expressed as:

$$P_3^* = C3k2^3\left(Conv_{1\times1}\left(Conv_{3\times3}(T_3)\right) + T_3\right) \quad (5)$$

where:

$C3k2^3$ denotes three consecutive stacked C3k2 modules, Conv3×3 and Conv1×1 denote 3×3 and 1×1 convolution operations respectively, and +T3 represents the residual connection.

In the bottom-up pathway, a 3 × 3 convolution with a stride of 2 is used for downsampling, and the C3k2 module is applied twice at both the P4 and P5 layers for feature aggregation. This design strengthens feature interaction across different scales and improves the representation of dense small lesion regions.



MPE-FPN

Fig. 6 - Architecture of MPE-FPN multi-scale feature pyramid

TPL-Head Two-Level P3 Detection Head

To improve localization accuracy for dense small lesions, this study redesigns the original detection head as TPL-Head (Two-Level P3 Detection Head). Unlike the conventional full-scale deployment strategy that places detection branches on the P3, P4, and P5 feature maps simultaneously, TPL-Head focuses on the P3-related feature hierarchy. This design is motivated by the practical characteristics of cucumber disease detection, in which lesion targets are mainly small, densely distributed, and highly sensitive to fine-grained spatial details. A statistical analysis of the annotated lesion instances revealed that approximately 66.3% of the bounding boxes belong to the small-object range, indicating that most disease targets in the dataset are small objects. This distribution provides quantitative support for the P3-focused design of the proposed TPL-Head.

TPL-Head employs two detection branches to construct complementary feature hierarchies. The first branch directly uses the final output of the P3 enhancement block, which has undergone complete feature enhancement and fusion and therefore contains stronger semantic representation capability. The second branch jointly uses the intermediate output of the P3 enhancement block and the final enhanced P3 output. In the original feature flow, the intermediate output refers to the feature after the 3×3 convolution and 1×1 convolution, but before the residual connection and the subsequent C3k2 refinement. Because this intermediate feature preserves richer spatial detail information, its combination with the final enhanced feature helps form a more complementary representation for disease lesion detection.

Let P_{3m} denote the intermediate P3 feature and P_{3e} denote the final enhanced P3 feature. The two-branch prediction process can be written as:

$$Y_1 = \mathrm{Det}_1(P_{3e}), \quad Y_2 = \mathrm{Det}_2(\mathrm{Fuse}(P_{3m}, P_{3e})) \tag{6}$$

where: $\mathrm{Det}_1(\cdot)$ and $\mathrm{Det}_2(\cdot)$ denote the two detection branches, and $\mathrm{Fuse}(\cdot)$ denotes feature fusion. The final prediction result is obtained by merging the outputs of the two branches and applying non-maximum suppression.

Training Environment and Parameter Settings

The HSLG-YOLO model was trained and evaluated under a unified hardware and software environment. The hardware platform consisted of an AMD EPYC 9754 128-Core Processor, 384 GB DDR5-4800 MHz memory, and an NVIDIA GeForce RTX 4060 GPU with 8 GB of video memory. The software environment included Ubuntu 22.04 LTS, Python 3.10.14, PyTorch 2.2.0, and CUDA 11.8.

The main hyperparameter settings used in the experiments are listed in Table 2. The optimizer was set to SGD, the batch size was 16, the number of training epochs was 400, the input image size was 640 × 640, the number of workers was 4, and close_mosaic was set to 0. Unless otherwise stated, the same data split, training schedule, and evaluation settings were used for all comparative experiments to ensure fair comparison.

Table 2

Main hyperparameter settings used in model training and evaluation

Parameter	Value
Optimizer	SGD
Batch size	16
Epoch	400
Image size	640×640
Workers	4
Close_mosaic	0

Model Performance Evaluation Indicators

To evaluate the performance of HSLG-YOLO for cucumber disease detection, Precision, Recall, mean Average Precision (mAP), model size, and GFLOPs were used as the main evaluation indicators. Precision, Recall, mAP@0.5, and mAP@0.5:0.95 were adopted to assess detection performance, while model size and GFLOPs were used to reflect model complexity and deployment efficiency.

Precision serves to evaluate the accuracy of the model's positive prediction outcomes, essentially characterizing the model's capability to reduce false positive detections. Specifically, it denotes the ratio of correctly identified diseased samples to all samples predicted as diseased. Its calculation formula is presented in Equation (7):

$$\text{Precision}(\%) = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \tag{7}$$

Here, TP (True Positive) represents the count of diseased samples detected accurately; FP (False Positive) indicates the number of normal samples mistakenly classified as diseased.

Recall centers on assessing the integrity of the model's detection for diseased samples, primarily reflecting the model's capability to reduce missed detections. Specifically, it represents the ratio of correctly identified diseased samples to all actually diseased samples. Its definition is presented in Equation (8):

$$\text{Recall}(\%) = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \tag{8}$$

where FN (False Negative) denotes the number of actual diseased samples not detected.

Mean Average Precision (mAP) is a core indicator for comprehensively measuring the model's detection effectiveness for different types of cucumber diseases. It is calculated by the weighted average of the Average Precision (AP) of each type of disease under multiple Intersection over Union (IoU) thresholds. Its calculation method is shown in Equation (9):

$$\text{mAP}(\%) = \frac{1}{\text{N}} \times \sum_{\text{i}=1}^{\text{N}} \text{AP}_{\text{i}} \times 100\% \tag{9}$$

Here N denotes the total count of cucumber disease categories; APi refers to the Average Precision of the i-th type of cucumber disease, serving to quantify the model's recognition capability for this specific disease type.

Model size, expressed in megabytes (MB), was used to measure storage cost and lightweight deployment potential. In addition, GFLOPs was used as an auxiliary indicator of computational complexity to reflect the inference cost of different detection models.

RESULTS

Comparison with Different Detection Models

To evaluate the performance of the proposed model, HSLG-YOLO and several representative detection models were trained and tested on the self-collected cucumber disease dataset under the same experimental conditions. The comparison results are shown in Table 3.

Table 3

Performance comparison of HSLG-YOLO and representative detection models on the self-collected cucumber disease dataset

Models	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Model Size (MB)	GFLOPs
HSLG-YOLO	93.58	91.88	96.49	67.89	4.6	7.3
YOLOv11n	90.47	90.07	94.50	64.40	5.3	6.3
YOLOv3	85.96	82.43	88.85	56.36	18.3	14.3
YOLOv5	88.44	88.32	92.89	58.84	4.5	5.8
YOLOv6	85.32	84.32	89.27	52.72	8.2	11.5
YOLOv8	88.45	87.54	93.05	59.91	5.4	6.8
YOLOv9t	87.34	85.33	90.95	54.99	4.0	6.4
YOLOv10	89.22	90.74	94.89	60.86	5.5	6.5
Hyper-YOLO	92.32	90.35	95.77	67.04	7.3	7.3
RT-DETR-X	75.34	74.77	79.06	38.31	129.1	222.5

As shown in Table 3, HSLG-YOLO achieved the best overall performance among the compared models, with a precision of 93.58%, a recall of 91.88%, an mAP@0.5 of 96.49%, and an mAP@0.5:0.95 of 67.89%. Compared with the YOLOv11n baseline, HSLG-YOLO improved precision by 3.11 percentage points, recall by 1.81 percentage points, mAP@0.5 by 1.99 percentage points, and mAP@0.5:0.95 by 3.49 percentage points, while reducing the model size from 5.3 MB to 4.6 MB.

Compared with YOLOv3, YOLOv5, YOLOv6, YOLOv8, YOLOv9t, and YOLOv10, HSLG-YOLO consistently achieved higher detection accuracy, especially in terms of mAP@0.5:0.95, indicating better localization performance under stricter IoU thresholds. Among the comparison models, Hyper-YOLO showed the closest performance to HSLG-YOLO, but the proposed model still obtained higher precision, recall, mAP@0.5, and mAP@0.5:0.95 while using a smaller model size.

In terms of computational complexity, HSLG-YOLO required 7.3 GFLOPs, which remained much lower than RT-DETR-X and YOLOv3 while maintaining superior detection performance. Although the GFLOPs of HSLG-YOLO were slightly higher than those of YOLOv11n, YOLOv5, YOLOv8, YOLOv9t, and YOLOv10, the proposed model achieved a more favorable balance between detection accuracy and lightweight deployment.

Overall, the comparison results demonstrate that HSLG-YOLO provides superior comprehensive performance among the evaluated detectors and is well suited for cucumber mildew lesion detection under greenhouse conditions.

Comparison of Backbone Networks

Table 4

Comparison of CAA-HGNet with representative backbone networks.

Backbone	Precision	Recall	mAP@0.5 (%)	Model Size (MB)
RevCol	88.77	86.47	92.47	4.5
C3K2-PPA	90.45	89.32	94.27	5.7
C3K2-EMA	88.22	86.77	92.18	5.7
Timm	89.39	89.24	93.70	25.2
C3K2-IDWB	88.41	87.23	82.14	5.1
Starnet	83.43	83.09	87.98	4.0
C3K2-SHSA	88.24	88.37	92.90	5.1
CAA-HGNet	89.51	87.83	93.29	3.8

To further evaluate the proposed backbone, CAA-HGNet was compared with several representative backbones in terms of Precision, Recall, mAP@0.5, and model size, as shown in Table 4. CAA-HGNet achieved 89.51% precision, 87.83% recall, 93.29% mAP@0.5, and a model size of 3.8 MB. Although some backbones obtained slightly higher accuracy, CAA-HGNet achieved the smallest model size while maintaining competitive performance. These results indicate that the proposed backbone provides a favorable balance between detection accuracy and lightweight deployment.

Ablation Experiments

Table 5

Ablation experiments of HSLG-YOLO for cucumber disease detection.

CAA-HGNet	MPE-FPN	TPL-Head	Precision (%)	Recall (%)	Model Size (MB)	mAP@0.5 (%)
✗	✗	✗	90.47	90.07	5.3	94.50
✓	✗	✗	89.51	87.83	3.8	93.29
✗	✓	✗	92.06	91.22	6.4	95.69
✗	✗	✓	91.61	87.76	5.2	93.56
✓	✓	✗	92.03	90.59	4.8	95.27
✓	✓	✓	93.58	91.88	4.6	96.49

Note: ✓ denotes that the module is adopted; ✗ denotes that the module is not adopted; bold values indicate the optimal performance.

To verify the contribution of each proposed component, ablation experiments were conducted using YOLOv11n as the baseline, and the results are shown in Table 5. Replacing the original backbone with CAA-HGNet reduced the model size from 5.3 MB to 3.8 MB, while precision, recall, and mAP@0.5 changed from 90.47%, 90.07%, and 94.50% to 89.51%, 87.83%, and 93.29%, respectively. Introducing MPE-FPN alone improved precision, recall, and mAP@0.5 to 92.06%, 91.22%, and 95.69%, respectively, but increased model size to 6.4 MB. When only TPL-Head was adopted, precision increased, but recall and mAP@0.5 decreased. After combining all three modules, the final HSLG-YOLO achieved the best performance, with 93.58% precision, 91.88% recall, 96.49% mAP@0.5, and a model size of 4.6 MB. This result indicates that the final improvement is achieved by the complementary effect of the backbone, neck, and head.

Visualization of Test Results

To further evaluate the practical performance of HSLG-YOLO for cucumber disease detection, representative test samples of downy mildew and powdery mildew were selected for visualization analysis in this section. Fig. 7 presents the comparison of downy mildew detection results, Fig. 8 presents the comparison of powdery mildew detection results, and Fig. 9 shows the normalized confusion matrix of the proposed model on the test set.

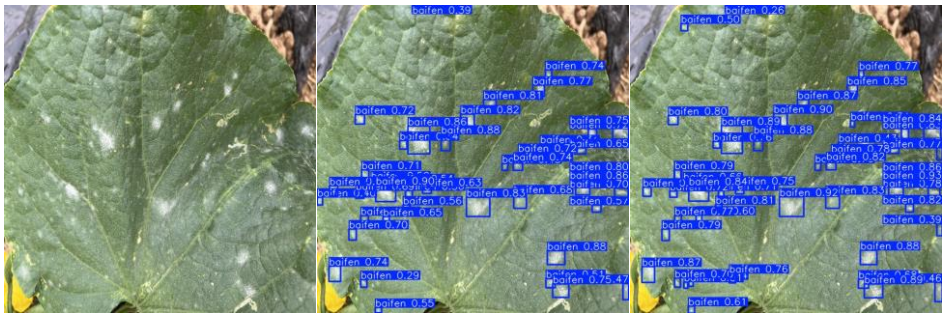
(a) (b) (c)

Fig. 7 - Comparison of Downy Mildew Detection Results

(a) Original image; (b) Detection results of YOLOv11n; (c) Detection results of HSLG-YOLO

For downy mildew, the lesions usually appear as small angular spots constrained by leaf veins and are often affected by complex textures, local reflections, and partial occlusion. As shown in Fig. 7, although the YOLOv11n baseline can identify part of the lesion regions, missed detections and localization deviations still occur in dense or low-contrast areas. In contrast, HSLG-YOLO detects more lesion targets on the same leaf and provides more complete lesion localization, indicating improved sensitivity and robustness for dense small lesions under greenhouse conditions.

For powdery mildew, the lesions are typically distributed densely on the leaf surface and often exhibit low contrast against the background. As shown in Fig. 8, the YOLOv11n baseline can detect some lesion regions, but several weak and small lesions are still missed, particularly in densely distributed areas. By comparison, HSLG-YOLO identifies more powdery mildew lesions and shows stronger responses in dense lesion regions, suggesting better detection completeness and stronger recognition ability for weak lesion targets.



(a) (b) (c)

Fig. 8 - Comparison of Powdery Mildew Detection Results

(a) Original image; (b) Detection results of YOLOv11n; (c) Detection results of HSLG-YOLO

To further examine the overall error distribution, Fig. 9 presents the normalized confusion matrix of HSLG-YOLO on the test set. The confusion matrix indicates that the proposed model achieved high recognition accuracy for both downy mildew and powdery mildew, while the remaining errors were mainly associated with missed detections and unmatched predictions in visually complex regions. Overall, the confusion matrix further supports the discriminative ability of the proposed model for the two cucumber mildew categories.

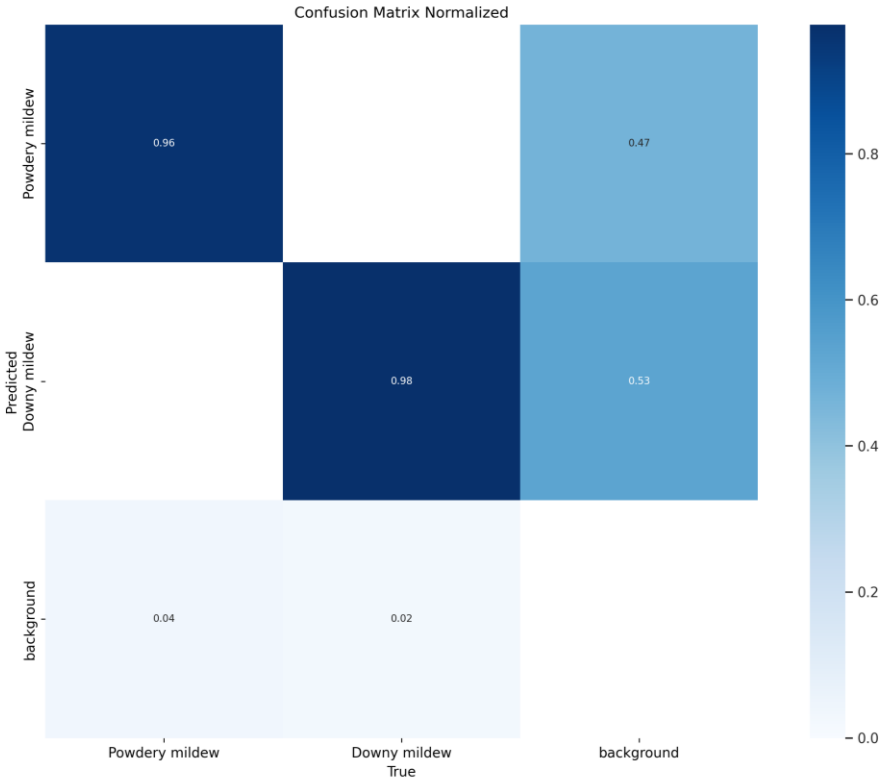


Fig. 9 - Normalized confusion matrix of HSLG-YOLO on the test set

In general, the visualization results are consistent with the quantitative comparison and ablation experiments. HSLG-YOLO provides more complete lesion detection, more accurate localization, and stronger robustness than the YOLOv11n baseline under complex greenhouse conditions. These qualitative results further confirm the effectiveness of the proposed CAA-HGNet backbone, MPE-FPN neck, and TPL-Head for cucumber mildew lesion detection.

Generalization Experiment

Table 7

Generalization performance of HSLG-YOLO and YOLOv11n on an additional private dataset and a Roboflow dataset

Dataset	Model	Precision (%)	Recall (%)	mAP@0.5 (%)	Model Size (MB)
Private Dataset	YOLOv11n	88.23	86.78	90.67	5.3
	HSLG-YOLO	90.54	88.43	93.67	4.6
Roboflow Dataset	YOLOv11n	89.65	88.03	92.34	5.3
	HSLG-YOLO	92.15	90.65	93.96	4.6

To further evaluate cross-dataset generalization, HSLG-YOLO and YOLOv11n were tested on an additional private cucumber disease dataset and a publicly available Roboflow dataset, as shown in Table 7. These two datasets were used only for external testing and were not involved in model training, hyperparameter tuning, or ablation experiments. For fair comparison, the same inference settings and evaluation metrics were applied to both models. On both datasets, HSLG-YOLO achieved higher precision, recall, and mAP@0.5 than YOLOv11n while maintaining a smaller model size (4.6 MB vs. 5.3 MB). These results indicate that the proposed model has better cross-dataset adaptability and preserves its lightweight advantage.

Edge Deployment Evaluation on Jetson Orin Nano

Table 8

Inference performance of HSLG-YOLO on the computer platform and Jetson Orin Nano.

Device Name	Power (W)	FPS	Latency (ms)
Computer Platform	135	88.79	9.6
Jetson Orin Nano	8	9.777	102.28

To examine deployment feasibility, inference testing was conducted on Jetson Orin Nano, and the results are shown in Table 8. HSLG-YOLO achieved 9.777 FPS with a latency of 102.28 ms under an 8 W power setting, while reaching 88.79 FPS and 9.60 ms latency on the computer platform. Although edge-side inference was slower than that on the high-performance platform, the model still maintained stable operation under low-power conditions.

From a practical application perspective, this performance indicates that HSLG-YOLO can support near-real-time cucumber disease scouting in greenhouse environments. The model could be deployed in a portable handheld device for manual inspection, a fixed monitoring unit installed near planting rows, or a mobile scouting platform moving along greenhouse aisles. In these scenarios, images or video frames can be periodically collected and processed on the edge device, allowing early warning of visible downy mildew and powdery mildew lesions without relying on high-power server hardware. Therefore, the Jetson Orin Nano results demonstrate not only computational feasibility, but also practical potential for lightweight greenhouse disease monitoring.

CONCLUSIONS

In this study, HSLG-YOLO, an improved lightweight detection model based on YOLOv11n, was proposed for visible lesion detection of cucumber downy mildew and powdery mildew under greenhouse conditions.

By integrating the CAA-HGNet backbone, the MPE-FPN neck, and the TPL-Head, the proposed model improved the balance between detection accuracy and lightweight deployment for dense small-lesion detection.

Experimental results on the self-collected cucumber disease dataset showed that HSLG-YOLO achieved 93.58% precision, 91.88% recall, 96.49% mAP@0.5, and 67.89% mAP@0.5:0.95, while reducing the model size from 5.3 MB to 4.6 MB compared with the YOLOv11n baseline. Ablation experiments further demonstrated that the proposed backbone, neck, and detection head contributed jointly to the final performance gain, indicating that the effectiveness of HSLG-YOLO results from the coordinated optimization of multiple components rather than from a single module alone.

The generalization experiments on an additional private dataset and a Roboflow dataset further showed that HSLG-YOLO maintained consistent performance advantages over YOLOv11n, indicating better robustness and cross-dataset adaptability within the tested scope. In addition, the edge deployment experiment on Jetson Orin Nano demonstrated that the proposed model could maintain stable inference under a low-power setting, suggesting its potential for practical deployment in embedded and intelligent greenhouse monitoring scenarios. These results suggest that HSLG-YOLO could be integrated into portable, fixed, or mobile edge-side monitoring systems for near-real-time cucumber disease scouting in greenhouse production.

Overall, the main contribution of this study is to provide a lightweight technical solution for automated monitoring of visible cucumber mildew lesions under greenhouse conditions. The proposed method is suitable as a technical basis for intelligent greenhouse disease scouting and may support the development of portable and edge-side agricultural vision systems.

Nevertheless, this study still has several limitations. Although the dataset was expanded, it currently includes only two disease categories and mainly focuses on visible lesion detection rather than early symptom identification, mixed infection analysis, or disease severity grading. In addition, the deployment evaluation was conducted on a single embedded platform, and broader validation under more diverse greenhouse conditions is still needed. Future work will expand the dataset to more disease categories and acquisition conditions, further evaluate the model on additional edge devices, and explore its integration with downstream tasks such as disease severity assessment and greenhouse management assistance.

ACKNOWLEDGEMENT

The authors thank the cucumber planting bases in Shouguang City, Shandong Province, for supporting field image acquisition. This work was supported by a Project of Shandong Province Higher Educational Program for Introduction and Cultivation of Young Innovative Talents in 2021.

REFERENCES

[1] Abade, A. S., Ferreira, P. A., and Vidal, F. D. B. (2021). Plant diseases recognition on images using convolutional neural networks: a systematic review. *Computers and Electronics in Agriculture*, 185, 106125.

[2] Cao, Y., Chen, L., Yuan, Y., and Sun, G. (2023). Cucumber disease recognition with small samples using image-text-label-based multi-modal language model. *Computers and Electronics in Agriculture*, 211, 107993.

[3] Cao, Y., Sun, G., Yuan, Y., and Chen, L. (2025). Small-sample cucumber disease identification based on multimodal self-supervised learning. *Crop Protection*, 188, 107006.

[4] George, R., Thuseethan, S., Ragel, R. G., Mahendrakumaran, K., Nimishan, S., Wimalasooriya, C., et al. (2025). Past, present and future of deep plant leaf disease recognition: A survey. *Computers and Electronics in Agriculture*, 234, 110128.

[5] Gupta, C., Gill, N. S., Gulia, P., Duhan, S., Alduaiji, N., Shukla, P. K., et al. (2025). Deep vision in agriculture: assessing the function of YOLO in the classification of plant leaf diseases. *BioData Mining*, 18, 91.

[6] Hou, Q., Zhou, D., and Feng, J. (2021). Coordinate attention for efficient mobile network design. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13713–13722.

[7] Jocher, G., Chaurasia, A., and Qiu, J. (2024). Ultralytics YOLO11 [software documentation]. *Ultralytics*. Available at: https://docs.ultralytics.com/models/yolo11/ (accessed 22 April 2026).

[8] Li, S., Li, K., Qiao, Y., and Zhang, L. (2022). A multi-scale cucumber disease detection method in natural scenes based on YOLOv5. *Computers and Electronics in Agriculture*, 202, 107363.

[9] Liu, J., and Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods*, 17, 22.
[10] Ma C., Zhang H., Ma X., Wang J., Zhang Y., and Zhang X., (2024). Lightweight wheat disease detection method based on improved YOLOv8 (基于改进 YOLOv8 的轻量化小麦病害检测方法), *Transactions of the Chinese Society of Agricultural Engineering*, vol.40, pp.187–195, Beijing/China.
[11] Qiao, C., Li, K., Zhu, X., Jing, J., Gao, W., and Zhang, L. (2025). Detection of cucumber downy mildew spores based on improved YOLOv5s. *Information Processing in Agriculture*, 12(2), 179–194.
[12] Qin, D., Leichner, C., Delakis, M., Fornoni, M., Luo, S., Yang, F., et al. (2024). MobileNetV4: Universal models for the mobile ecosystem. *arXiv preprint*, arXiv:2404.10518.
[13] Rana, S., Hensel, O., Nasirahmadi, A., et al. (2026). From vineyard to vision: Multi-domain analysis and mitigation of grape cluster detection failures in complex viticultural environments. *Results in Engineering*, 29, 108833.
[14] Raufer, L., Wiedey, J., Mueller, M., Penava, P., and Buettner, R. (2025). A deep learning-based approach for the detection of cucumber diseases. *PLOS ONE*, 20(4), e0320764.
[15] Upadhyay, A., Chandel, N. S., Singh, K. P., Chakraborty, S. K., Nandede, B. M., Kumar, M., et al. (2025). Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models, and trends in precision agriculture. *Artificial Intelligence Review*, 58, 92.
[16] Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7464–7475.
[17] Wang, C. Y., Yeh, I. H., and Liao, H. Y. M. (2024). YOLOv9: Learning what you want to learn using programmable gradient information. In: *Computer Vision – ECCV 2024*, pp. 1–21. https://doi.org/10.1007/978-3-031-72751-1_1
[18] Wang, X., et al. (2024). Detection of small targets in cucumber disease images through global information perception and feature fusion. *Frontiers in Sustainable Food Systems*, 8, 1366387.
[19] Xie, J., Xie, J., Li, Z., et al. (2025). An improved YOLOv8-based method for detecting pests and diseases on cucumber leaves in natural backgrounds. *Sensors*, 25(5), 1551.
[20] Zeng, T., Li, S., Song, Q., et al. (2023). Lightweight tomato real-time detection method based on improved YOLO and mobile deployment. *Computers and Electronics in Agriculture*, 205, 107625.
[21] Zhao, J., et al. (2025). A review of plant leaf disease identification by deep learning. *Frontiers in Plant Science*, 16, 1637241.