

Article

Research on Lightweight Citrus Leaf Pest and Disease Detection Based on PEW-YOLO

Renzheng Xue ^{1,2} and Luqi Wang ^{1,2,*} 

¹ School of Computer and Control Engineering, Qiqihar University, Qiqihar 161000, China; 00254@qqhru.edu.cn

² Heilongjiang Key Laboratory of Big Data Network Security Detection and Analysis, Qiqihar University, Qiqihar 161000, China

* Correspondence: 13124581002@163.com; Tel.: +86-13124581002

Abstract: Timely detection and prevention of citrus leaf diseases and pests are crucial for improving citrus yield. To address the issue of low efficiency in citrus disease and pest detection, this paper proposes a lightweight detection model named PEW-YOLO. First, the PP-LCNet backbone is optimized using a novel GSConv convolution, and a lightweight PGNet backbone is introduced to reduce model parameters while enhancing detection performance. Next, the C2f_EMA module, which integrates efficient multi-scale attention (EMA), replaces the original C2f module in the neck, thereby improving feature fusion capabilities. Finally, the Wise-IoU loss function is employed to address the challenge of identifying low-quality samples, further improving both convergence speed and detection accuracy. Experimental results demonstrate that PEW-YOLO achieves a 1.8% increase in mAP50, a 32.2% reduction in parameters, and a detection speed of 1.6 milliseconds per frame on the citrus disease and pest dataset, thereby meeting practical real-time detection requirements.

Keywords: image identification; citrus leaf disease detection; deep learning; convolutional neural network; YOLOv10; feature fusion



Academic Editors: Lei Wang, Xin Zhang and Xixing Li

Received: 19 March 2025

Revised: 23 April 2025

Accepted: 27 April 2025

Published: 29 April 2025

Citation: Xue, R.; Wang, L. Research on Lightweight Citrus Leaf Pest and Disease Detection Based on PEW-YOLO. *Processes* **2025**, *13*, 1365. <https://doi.org/10.3390/pr13051365>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Citrus, as a delicious and nutritious fruit, is widely distributed around the world. With its unique fragrance, rich nutrients, and various health benefits, citrus has become an important part of global agriculture and an indispensable component of daily diets [1]. However, various diseases and pests, including canker disease, melanosis disease, fuliginous, the citrus fruit fly, and psyllids, seriously threaten the health of citrus trees. These diseases and pests are highly contagious and cause significant damage to citrus leaves, buds, and fruits [2], hindering the normal growth and development of citrus trees, with severe cases potentially leading to tree death. Hence, prompt identification and effective management of these pathogens and insects are essential for preserving citrus production and fruit standards [3]. To effectively control the spread of citrus leaf diseases and pests, various identification technologies have been widely applied. Among these technologies, computer vision is undoubtedly representative, playing a significant role in suppressing the spread of citrus leaf diseases and pests and reducing economic losses in orchards [4].

In recent years, significant advancements have been made in deep learning methodologies, which have been widely adopted for visual computing tasks, especially for identifying and classifying foliage in agricultural settings [5]. Deep learning, through automatic feature learning, has significantly improved detection accuracy [6]. Application areas include

leaf image detection and fruit image analysis [7]. Despite significant achievements, deep learning still requires large-scale training and model optimization. This field continues to evolve, with new algorithms and technologies continuously emerging to further improve leaf recognition performance. In research on detecting and recognizing citrus leaf diseases and pests, various researchers have proposed different approaches. For example, Leng et al. [8] proposed a network model, for leaf disease detection, which enhances the fusion of multi-scale position information while reducing parameter computation by introducing feature reconstruction and fusion networks. However, this model is susceptible to interference from complex texture edges when the image resolution is low. Xing et al. [9] developed a new feature reuse model, BridgeNet-19, which can identify common citrus leaf diseases and pests in citrus plantations, achieving a classification accuracy of 95.47%. However, feature extraction for leaves in complex backgrounds requires significant parameter computation, resulting in slower detection speed. Lin et al. [10] built a pest recognition system using a two-stage deep learning algorithm. While the detection results were satisfactory, the system could only detect a limited number of pests, which did not meet the needs for detecting leaves with varying degrees of disease in orchards. Liang et al. [11] proposed the YOLOv5-sbic algorithm for detecting late autumn young buds of lychees, with a detection efficiency of 79.6%. However, it does not significantly improve the detection accuracy for small objects. Gao et al. [12] proposed an apple leaf disease detection model, YOLOv8n-GGi, which can identify disease features on leaves with intricate backgrounds while retaining valuable information about the leaf disease. The use of depthwise separable convolution helps reduce model parameters, achieving a model size of 3.8 MB. Gangwar et al. [13] introduced an enhanced Transformer model, which performs well in detecting multiple categories of tomato leaves, with a precision of 93.51% and a model size of 5.8 MB. This model can be conveniently deployed on edge devices for real-time monitoring, but its large computational load and poor real-time performance remain limitations.

In summary, although there have been notable advancements in detecting plant diseases and pests in challenging environments, challenges remain, such as the multi-scale issues arising from different citrus plant diseases and pests, as well as environmental variations. Furthermore, occlusion interference among citrus leaves affects target recognition, leading to the need for significant parameter computation for feature extraction, which increases deployment and operational costs and reduces the model's practical efficiency. To address this, we present a lightweight model for detecting citrus leaf diseases and pests, which drastically reduces the number of parameters and floating-point operations while maintaining effective detection capabilities, resulting in enhanced detection efficiency. The key contributions of this work include:

- (1) We propose an improved real-time citrus leaf pest and disease detection model, PEW-YOLO, based on YOLOv11. The model is specifically designed to address the challenges of detecting citrus leaf diseases and pests under complex natural environmental conditions.
- (2) A lightweight backbone network, PGNet, is designed to enhance the interaction of information between channels. It leverages the novel GSConv convolution to enable efficient citrus leaf feature extraction while significantly reducing computational complexity.
- (3) A new neck structure, C3k2-EMA, is introduced to expand the receptive field across pixels and strengthen multi-scale contextual feature fusion. At the same time, it improves the model's focus on the disease-affected target regions.
- (4) The original CIoU loss function is replaced with the Wise-IoU loss function to optimize bounding box regression, thereby improving accuracy and reliability in the detection of small objects associated with citrus leaf diseases and pests.

In this paper, Section 2 introduces some related research work. Section 3 describes the presented detection model for citrus leaf diseases and pests. Section 4 analyzes the experimental performance of the PEW-YOLO model and presents the detection results of citrus leaf pests and diseases in real environments. Section 5 concludes this article.

2. Related Work

2.1. YOLOv11

Released by the Ultralytics team in September 2024, the YOLOv11 model [14] is the most recent iteration of the YOLO series. It employs an end-to-end neural network to predict object bounding boxes and labels in images. Compared to the previous version, YOLOv10, YOLOv11 introduces a novel C2PSA structure in its fusion architecture and removes the PSA and SCDown modules. These changes reduce redundant feature information while enhancing the model's focus on target regions, thereby improving its feature extraction capability. Additionally, YOLOv11 incorporates the newly designed C3k2 module, which enables the model to learn more effective features and strengthens multi-branch information fusion in the neck network. These improvements make YOLOv11 particularly well suited for detecting citrus leaf diseases and pests in complex environmental conditions. YOLOv11 is available in five versions: YOLOv11n, YOLOv11s, YOLOv11m, YOLOv11l, and YOLOv11x. This study chooses the YOLOv11n model due to its compact size and superior accuracy. The YOLOv11n architecture includes the input layer, backbone, neck, and head networks. The backbone and neck networks of YOLOv11 are inspired by the C2f structure in YOLOv10 [15], featuring a more gradient-rich C3k2 structure. The detection head uses a decoupling approach, where the classification head and detection head are separated. YOLOv11 excels in scalability and has significant advantages for small object detection. Additionally, YOLOv11 employs VFL loss for classification and CIOU + DFL loss for regression. VFL introduces an asymmetric weighting operation, modeling the bounding box position as a general distribution, enabling the network to rapidly converge to the object's location and align the probability density as closely as possible with the target. Overall, compared to previous YOLO versions, YOLOv11n has a simpler network structure while achieving higher accuracy and faster detection speed.

2.2. Lightweight Object Detection Algorithms

To improve the efficiency of image data processing on edge devices, designing lightweight deep neural network architectures is crucial. CNNs offer high computational efficiency when extracting feature relationships, but excessive convolution operations and stacking of multiple layers can result in large model parameters, reducing the efficiency of long-range semantic information exchange [16]. In recent years, this field has gained significant attention in both academia and industry, with researchers proposing innovative approaches. For instance, Google's MobileNet architecture [17] successfully reduces complexity by substituting ordinary convolutions with depthwise separable modules. Networks based on CNNs and ShuffleNet [18] employ concurrent structures, integrating local features of different resolutions and global representations into an interactive network, significantly enhancing the network's expressive power and effectively fusing multi-scale features of citrus disease and pest leaves. PP-LCNet [19] further develops this concept by combining 3×3 depthwise separable convolutions with 1×1 pointwise convolutions, ensuring a low parameter count while improving convolution efficiency. Wang et al. [20] proposed an apple leaf disease detection model, LCGSC-YOLO, which utilizes the lightweight PP-LCNet to rebuild the backbone structure network. This approach increases the receptive field and decreases parameter computation, enabling efficient and rapid leaf disease detection. These techniques effectively lower the network's parameter

count while improving its detection performance. Inspired by these advancements, this paper incorporates PP-LCNet into the YOLOv11n network, enabling lightweight feature extraction and collaborative integration of contextual features, thus improving detection efficiency.

2.3. Attention Mechanisms

Attention mechanisms mimic human perceptual processes by focusing more on the feature information regions of the target itself. They have demonstrated excellent performance in various image processing tasks [21]. Attention mechanisms [22] calculate attention weights from the input tensor and re-weight the input tensor based on these weights. Typically, attention modules are inserted after convolutions to improve the model's capacity to handle both short- and long-term dependencies, thus improving the learning of target features. In terms of attention improvements, Jia et al. [23] suggested incorporating the ECA module into the VGG16 network, which notably improved the model's citrus pest classification performance while reducing training time and computational costs. Dai et al. [24] introduced an enhanced R-CNN model, combining the CBAM module with the HRNet for feature extraction, improving the recognition of small target pests. However, the model they proposed still lacks sufficient multi-scale feature extraction. Therefore, we introduce a powerful EMA module in the proposed C3k2_EMA neck network structure, which maintains internal high resolution and fuses SoftMax–sigmoid combinations only in the channel and spatial attention blocks.

3. Proposed Method

The overall architecture of the lightweight citrus leaf pest and disease detection algorithm based on PEW-YOLO is illustrated in Figure 1. First, the original backbone of the YOLOv11n model is replaced with the PP-LCNet structure while retaining the original C2PSA module. Then, the GSConv lightweight convolution is integrated into PP-LCNet, leading to the development of a novel lightweight backbone named PGNet. Next, the EMA attention mechanism is embedded into the bottleneck structure to form the enhanced Bottleneck_EMA module. This module is further utilized to optimize the C3k2 module in the neck network, resulting in a newly proposed C3k2_EMA feature extraction structure that strengthens multi-scale feature fusion in the neck. Finally, the Wise-IoU [25] loss function is introduced during training, replacing the original CIoU loss to improve the performance of bounding box regression.

3.1. PGNet Lightweight Backbone Network

In the YOLOv11n network architecture, the C3k2 module in the backbone network has a relatively deep network structure, which, although helpful in improving the model's feature extraction capability, increases the computational complexity, thus leading to longer training and inference times. PP-LCNet is a compact convolutional neural network optimized using Intel's MKL-DNN, which enhances processing efficiency, offering advantages such as high efficiency, low latency, and low computational cost. It has improved the performance of lightweight models on multiple tasks, enhancing detection accuracy while maintaining the fast speed characteristic of the MobileNet network, making it more suitable for embedded devices and mobile applications. PP-LCNet builds efficient deep neural networks through local connection blocks and utilizes depthwise separable convolutions to minimize computational cost and improve the network's ability to generalize. Additionally, the GSConv [26] module is a lightweight convolution that maximizes the advantages of depthwise separable convolutions while eliminating the negative effects of feature extraction performance degradation caused by channel information separation in depthwise

separable convolutions. GSConv initially applies a standard convolution on the feature map, then performs a depthwise separable convolution, and subsequently combines the two feature maps before executing a shuffle operation to recombine the channels. This method fully utilizes the strengths of both convolutions and significantly enhances model detection performance. Figure 2 illustrates the structure of GSConv.

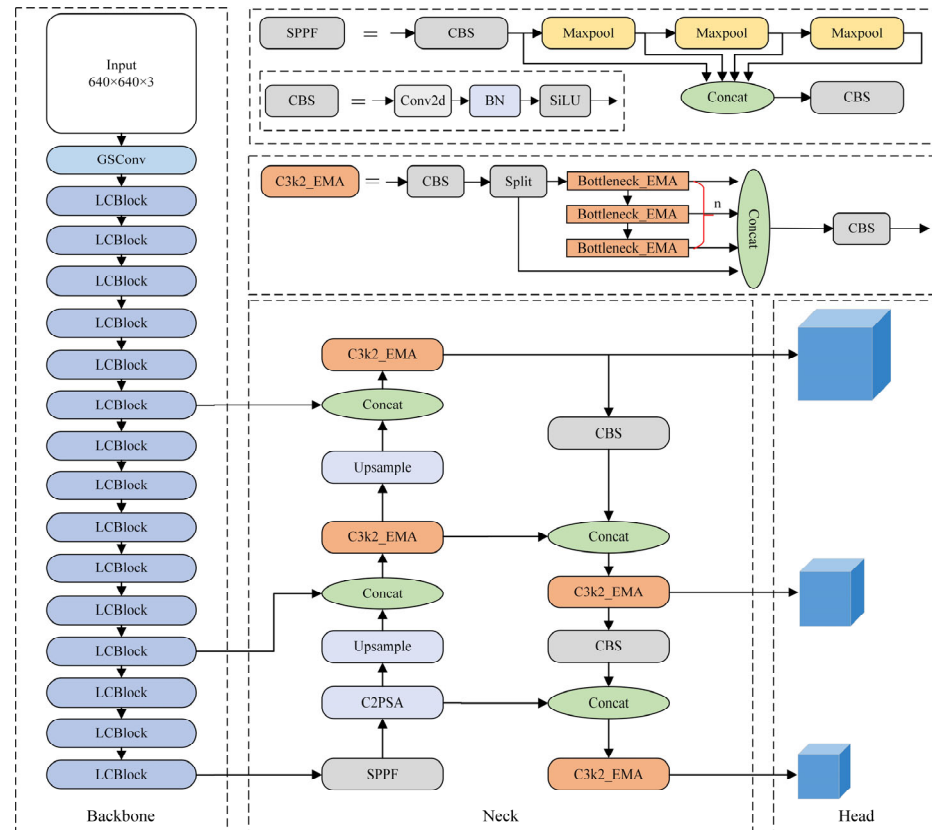


Figure 1. PEW-YOLO network structure.

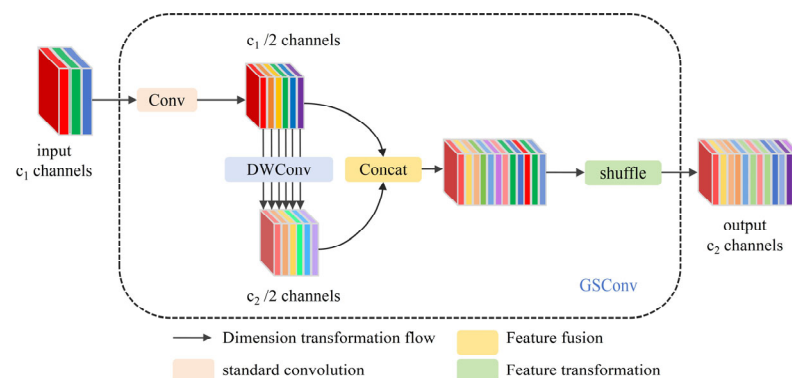


Figure 2. Structure of GSConv.

In order to simplify the model structure while ensuring detection accuracy, this paper removes the SENet attention mechanism from PP-LCNet and introduces the GSConv module at the input of the backbone network, constructing a new lightweight network, PGNet. “P” represents PP-LCNet, and “G” represents GSConv. In YOLOv11n, the PGNet lightweight network substitutes the original backbone, enabling both reduced weight and faster detection. The enhanced backbone structure is segmented into 16 sections (S1–S16). To capture more significant features, S1 is a GSConv convolution module that performs feature extraction using a 3×3 convolution. By introducing GSConv convolutions, the

network is able to adaptively prioritize channels, boosting its focus on key features and enhancing its discriminative power. S2 and S3 are composed of LCBlocks designed to reduce model parameters and network computation. Batch normalization (BN) layers are added between depthwise convolutions (DWConv) and pointwise convolutions (PWConv) in all LCBlocks. The convolution layers from S2 to S8 utilize a 3×3 kernel, while the following 8 layers from S9 onward, including those in S17, employ a 5×5 kernel. The network incorporates batch normalization and activation functions in the S2 to S16 blocks to enhance its ability to model non-linear relationships, thereby capturing intricate data patterns. To accelerate inference speed, the network uses the approximate Hard-Swish function in the convolutional layers. The structure of PGNet is shown in Figure 3.

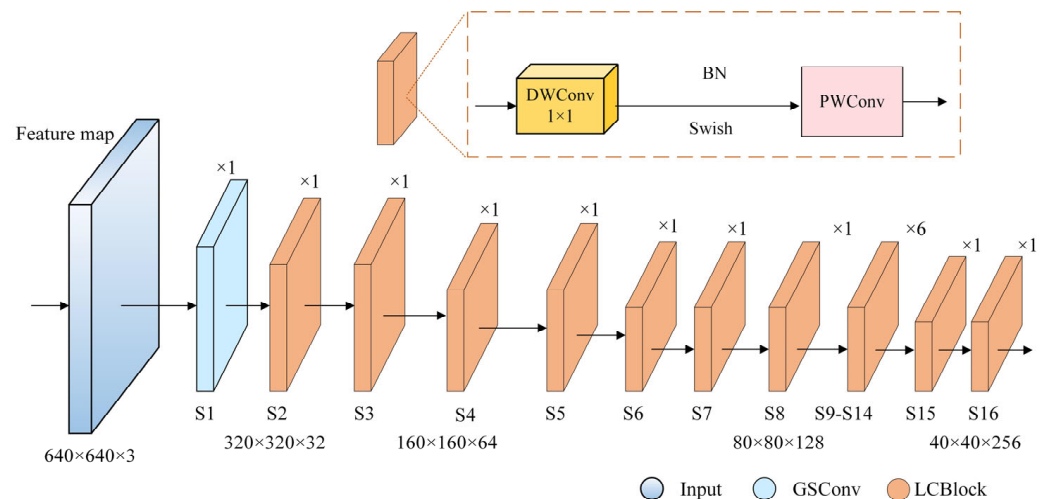


Figure 3. Structure of PGNet.

3.2. C3k2_EMA Module

During citrus leaf disease and pest detection, challenges such as partial occlusion between citrus leaves and changes in environmental lighting can lead to missed detections. Additionally, there are similarities between the characteristics of various leaf diseases and nearby objects, leading to possible false detection. These issues stem from the model's inability to efficiently capture essential features of citrus leaf diseases and pests or its inadequate filtering of the extracted feature information.

To improve the model's detection capability for citrus leaf diseases and pests, this study introduces the EMA module into the C3k2 structure of the neck network in YOLOv11n. The EMA attention module, proposed by Ouyang et al. [27] in 2023, is a high-efficiency multi-scale attention mechanism that leverages cross-space learning. By reorganizing certain channels within the batch dimension and grouping them, without the need for dimensionality reduction operations, it prevents the loss of channel feature information and reduces computational overhead, offering high accuracy with fewer parameters. Figure 4 illustrates the design of the EMA module. Its workflow is as follows: first, for any input $X \in R^{C \times H \times W}$, EMA divides the input along the channel axis into G sub-features, i.e., $X = [X_0, X_1, \dots, X_{G-1}]$, where $X \in R^{C \parallel G \times H \times W}$, to capture various semantic information, EMA then employs three pathways to extract attention weight descriptors from the grouped feature maps.

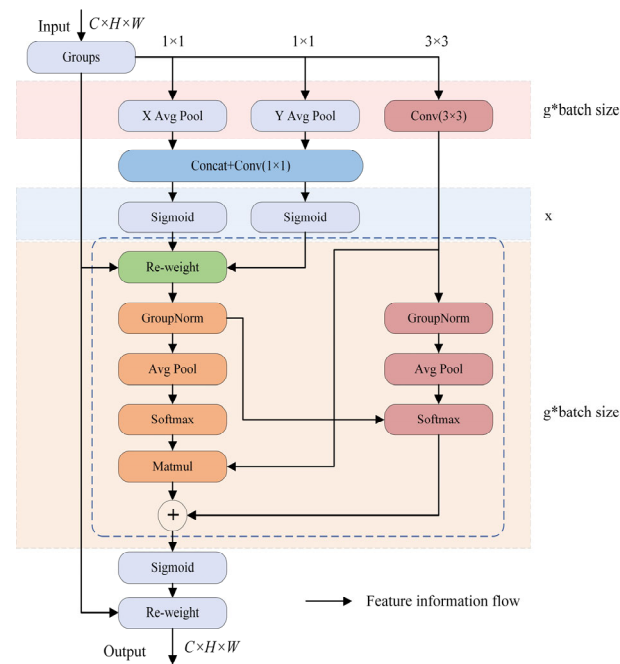


Figure 4. Structure of EMA.

The EMA structure consists of three parallel channel branches: the first two branches use convolution modules with a kernel size of 1 for feature extraction, and cross-channel information interaction is achieved through one-dimensional pooling operations in the horizontal and vertical directions; the third branch uses a convolution module with a kernel size of 3 to capture information from a larger receptive field while omitting pooling operations and GroupNorm to retain multi-scale feature representations. Finally, a pooling operation is applied to encode global spatial information from the outputs of the three branches, thus integrating multi-scale features. The pooling operation is given by the Formula (1):

$$Z_c = \frac{1}{H \times W} \sum_j \sum_i X_C(i, j) \quad (1)$$

In the formula, Z_C represents the pooled output of the C -th channel, where H and W are the height and width of the input feature map, C denotes the total number of channels, and $X_C(i, j)$ is the value at coordinate (i, j) in the C -th channel. The feature map is obtained by combining two attention weights. Finally, pixel-level dependencies are captured using the sigmoid activation function to extract semantic features of the entire scene.

As illustrated in Figure 5, we integrate the EMA module into the bottleneck architecture of C3k2 to build the C3k2_EMA module. This design improves the bottleneck structure, significantly improving the C3k2 module's capability to recognize leaf disease target features in complex garden scenarios. Based on previous experimental experience, this study improves the YOLOv11n architecture by replacing the C3k2 module in its neck network with the optimized C3k2_EMA module. With the integration of the EMA module, the model can leverage both 1×1 and 3×3 convolutions to capture more context information in the intermediate feature maps, further extracting and filtering feature information related to citrus leaf diseases and pests. This approach of integrating information across varying spatial scales enables the model to better handle challenges related to undetected and incorrect identifications in the detection of citrus leaf diseases and pests.

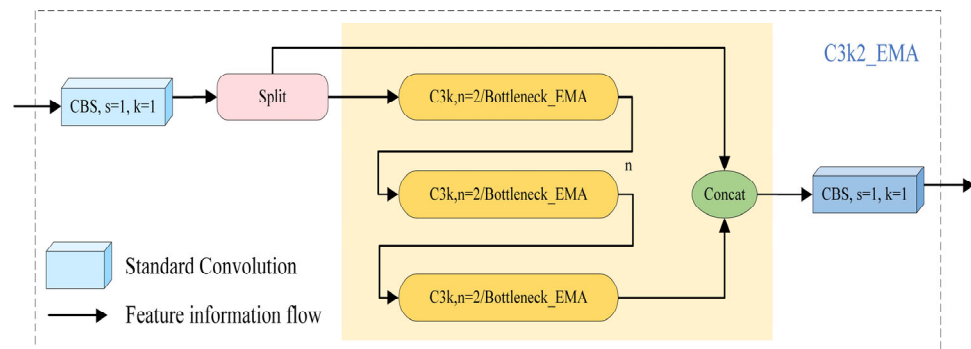


Figure 5. C3k2 structure with EMA attention mechanism.

3.3. Design of the Loss Function

3.3.1. Limitations of CIoU

For the bounding-box regression task, YOLOv11n employs DFL Loss and CIoU Loss. While CIoU incorporates a monotonic focusing mechanism, it often struggles to balance hard and easy samples, which can impair model performance—particularly when the dataset includes suboptimal or noisy instances.

3.3.2. Proposed Wise-IoU Loss

To address this limitation, we introduce the Wise-IoU loss, which incorporates an adaptive non-monotonic focusing strategy to better balance sample contributions. Unlike traditional IoU losses, Wise-IoU replaces the standard IoU term with an outlier-aware evaluation of anchor box quality, alleviating excessive penalties caused by spatial factors such as misalignment or small object scale. The mathematical formulation is shown in Equations (2)–(4).

$$L_{\text{Wise-IoU}} = r R_{\text{Wise-IoU}} L_{\text{IoU}}, r = \frac{\beta}{\delta \alpha^{\beta-\alpha}} \quad (2)$$

$$\beta = \frac{L_{\text{IoU}}^*}{L_{\text{IoU}}} \in [0, +\infty) \quad (3)$$

$$R_{\text{Wise-IoU}} = \exp\left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{(c_w^2 + c_h^2)^*}\right) \quad (4)$$

In these equations, $L_{\text{IoU}} \in [0, 1]$ refers to the IoU loss, which reduces the penalty for well-matched anchor boxes and increases attention on the center point distance when the overlap between the anchor and predicted box is significant. $R_{\text{Wise-IoU}} \in [1, \exp]$ represents the Wise-IoU penalty, which strengthens the loss for anchor boxes of ordinary quality. The asterisk (*) indicates terms that do not participate in backpropagation, effectively preventing non-convergent gradients in the model. $\overline{L_{\text{IoU}}}$ is the normalization factor, representing the incremental moving average. β denotes the outlier degree, with smaller values indicating higher-quality anchor boxes, which are assigned smaller gradient gains. This mechanism also reduces the gradient impact on predicted boxes with large outlier values, minimizing the negative influence of low-quality training samples. As a result, the bounding box regression loss emphasizes anchor boxes of average quality, leading to enhanced network performance.

4. Experiments

4.1. Dataset

The dataset used in this study is derived from a publicly available citrus leaf disease dataset on Kaggle, which includes 3166 images of citrus pest and disease leaves collected under different lighting conditions and weather. The dataset is divided into five categories:

Oriental fruit fly (*Bactrocera dorsalis* Hendel), canker, sooty mold (Fuliginous), melanose, and citrus psyllid (Psyllid). The final dataset, named “LPD Datasets”, consists of 2125 disease images and 1041 pest images. The dataset is divided into training, validation, and test subsets with an 8:1:1 distribution, comprising 2532 images for training, 317 images for validation, and 317 images for testing. Figure 6 shows some typical examples from the dataset. Although the dataset includes only five common citrus disease and pest categories, we recognize that real-world scenarios may present a wider variety of conditions and classes. To improve generalizability, we employed extensive data augmentation techniques (e.g., random occlusion, brightness jittering, and noise injection) during training. Figure 7 shows some sample images from the dataset after data augmentation.



Figure 6. Examples of typical samples from the dataset. (a) *Bactrocera dorsalis* Hendel; (b) canker; (c) fuliginous; (d) melanose; (e) psyllid.



Figure 7. Partially augmented images from the dataset.

4.2. Experiment Platform

The experimental platform used for model training and testing is as follows: the network structure model was built using the Python 3.9.12 programming software, utilizing the CUDA 11.1 and PyTorch 1.11 neural network frameworks. The model was run on an NVIDIA GeForce RTX 4090 GPU, operating on the Ubuntu 18.04 system.

4.3. Model Training

During the model training process, the input citrus leaf images were resized to 640×640 , with a batch size of 32. The SGD optimizer was used, with a learning rate of 0.01, and the learning rate was reduced using a cosine annealing schedule. The model was trained for 150 epochs, gradually reaching stability. Mosaic data augmentation was employed, where four images were read at a time and various transformations (e.g., flipping, scaling) were applied before concatenating them, thereby enriching the image scenes. Label smoothing was applied with a value of 0.01 to reduce overfitting and enhance the model's ability to generalize. As demonstrated in Figure 8, after 38,850 iterations, the PEW-YOLO model exhibited a more stable loss curve and improved detection consistency compared to the YOLOv11n model.

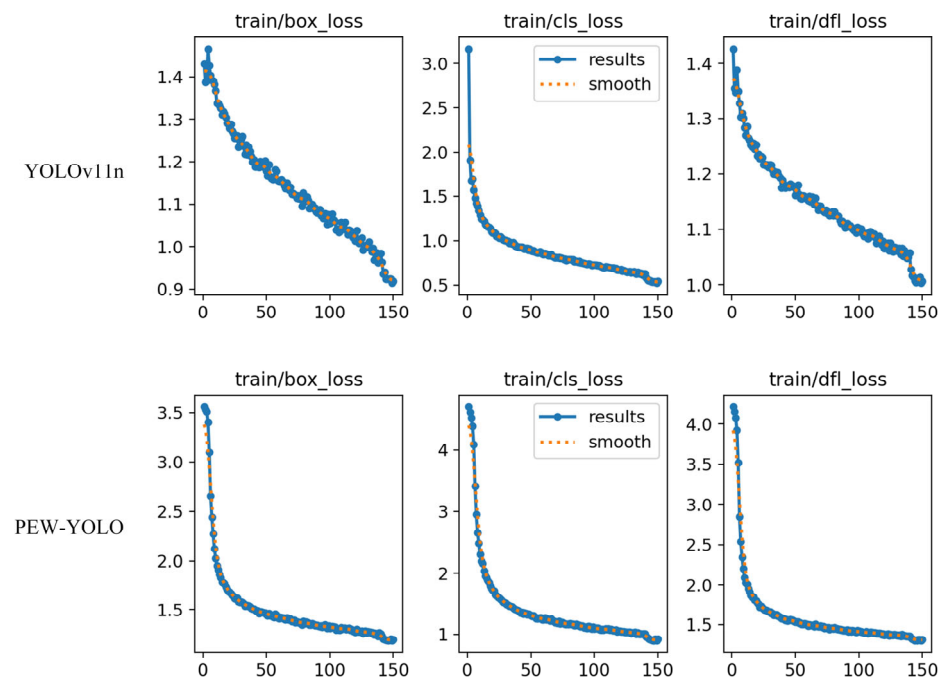


Figure 8. Loss curve comparison for training YOLOv11n and PEW-YOLO models.

4.4. Evaluation Metrics

In order to better evaluate the detection performance of different models, several evaluation metrics were selected. These metrics include mean average precision (mAP50), recall (R), precision (P), number of parameters (Params), and inference time (Times). A performance comparison between YOLOv11 and the improved YOLOv11 was conducted using these five evaluation metrics. The formulas for these metrics are as follows:

$$Precision = \frac{N_{TP}}{N_{TP} + N_{FP}} \quad (5)$$

$$Recall = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad (6)$$

$$AP = \int_0^1 P(r) dr \quad (7)$$

$$mAP = \frac{\sum_{i=1}^m AP_i}{m} \quad (8)$$

In the formula, TP denotes true positives, where the model's prediction matches the actual positive samples, and FN represents false negatives, where the model predicts negative while the actual sample is positive. By considering these metrics, a more comprehensive

understanding of the citrus leaf disease and pest leaf detection method's performance can be gained, offering targeted guidance for further improvements to the method.

4.5. Experimental Results and Analysis

4.5.1. Impact of Different Lightweight Backbone Networks on Model Performance

Within the YOLOv11n framework, the feature extraction backbone plays a key role in determining the model's overall performance. To evaluate its impact, we compared ShuffleNetv2, GhostNetv2, MobileNetv3, and PP-LCNet. Table 1 presents the performance evaluation results of these lightweight backbone networks.

Table 1. Comparison of different lightweight backbone network detection results for YOLOv11n.

Model	P (%)	R (%)	mAP50 (%)	Params (M)	Time (ms)
ShuffleNetv2-YOLOv11n	78.6	77.0	83.3	1.88	1.6
GhostNetv2-YOLOv11n	77.2	78.7	83.6	1.66	1.6
MobileNetv3-YOLOv11n	79.9	75.7	83.8	1.95	2.1
PP-LCNet-YOLOv11n	81.6	75.6	84.2	1.86	1.2

MobileNetv3-YOLOv11n has the highest parameter count but a low recall rate (R) of 75.7%, and it has a longer detection time of 2.1 ms/frame. GhostNetv2-YOLOv11n has the fewest parameters but a low precision (P) value and an mAP50 of only 83.6%. ShuffleNetv2-YOLOv11n shows average performance, with P and mAP50 values of 79.9% and 83.8%, respectively, though it has more parameters and slower detection speed. PP-LCNet-YOLOv11n achieved an mAP50 of 84.2%, which is 0.9%, 0.6%, and 0.4% higher than ShuffleNetv2-YOLOv11n, GhostNetv2-YOLOv11n, and MobileNetv3-YOLOv11n, respectively. Additionally, the parameter count of PP-LCNet is 1.86 M, which is 0.02 M and 0.09 M lower than that of ShuffleNetv2-YOLOv11n and MobileNetv3-YOLOv11n, respectively. It also achieved a recognition accuracy of 81.6%, higher than that of other lightweight models, with a detection time of only 1.2 ms/frame, which is sufficient for real-time detection.

In summary, the PP-LCNet-YOLOv11n model was selected for citrus leaf pest and disease detection due to its combination of strong detection accuracy and efficient design, making it ideal for use on devices with limited computational and storage capacities.

4.5.2. Impact of GSConv Integration on Network Performance

Integrating the GSConv module into the YOLOv11n model enables effective learning of citrus disease and pest features under complex environmental conditions, which is particularly important for efficient and lightweight image detection. Through a detailed evaluation and preliminary screening of four potential lightweight convolution modules, GSConv was selected to be incorporated into the S1 stage of the PP-LCNet backbone. The objective was to identify the optimal configuration that achieves the highest performance in terms of mAP50 while minimizing the model's parameter count and computational cost. The first alternative involved incorporating MBConv into the backbone, the second utilized GhostConv, the third adopted DWConv, and the fourth integrated GSConv. As shown in Table 2, comparative analysis of these configurations demonstrates that the introduction of GSConv into the backbone yields the best overall performance. Although the inference time slightly increased compared to the baseline model, the precision reached 82.8%, recall was 80.6%, and mAP50 achieved 85.2%. Notably, the model size remained unchanged, with the number of parameters maintained at 1.86 M.

Table 2. Performance comparison of different lightweight convolutions in the PP-LCNet backbone.

Model	P (%)	R (%)	mAP50 (%)	Params (M)	Time (ms)
Baseline (YOLOv11n+PP-LCNet)	81.6	75.6	84.2	1.86	1.2
+MBCConv	77.7	78.5	84.2	1.86	2.0
+GhostConv	80.4	76.1	83.3	1.86	2.1
+DWConv	80.3	76.6	83.3	1.86	2.0
+GSConv	82.8	80.6	85.2	1.86	1.4

4.5.3. Effect of Different Attention Mechanisms on Model Performance

For citrus leaf pest and disease detection, this study evaluated four different attention mechanisms: LSK Attention, SE Attention, GAM Attention, and EMA. The results of the comparison are presented in Table 3.

Table 3. Comparison of detection results using different attention mechanisms.

Model	P (%)	R (%)	mAP50 (%)	Params (M)	Time (ms)
Baseline	83.8	79.1	86.8	2.58	1.3
+LSKAttention	82.3	81.2	86.9	2.64	1.5
+SEAttention	81.6	77.7	85.7	2.58	1.6
+GAMAttention	80.5	82.0	86.4	3.22	1.4
+EMA	84.0	79.9	89.2	2.47	1.4

EMA performs the best in terms of the mAP50 metric, achieving a performance of 89.2%, which is 2.4%, 2.3%, 3.5%, and 2.8% higher than Baseline, LSKAttention, SEAttention, and GAMAttention, respectively. The precision and recall rates of EMA were 84.0% and 79.9%, with a parameter count of 2.47 M. GAM Attention achieved the highest recall rate (82.0%) but had a relatively large parameter count. SE Attention showed slower detection speed. LSK Attention had the largest parameter count, which contributed to its strong expressive power but may lead to overfitting. Considering these results, EMA strikes a good balance between performance and parameter count, and it was selected as the attention mechanism to integrate into the YOLOv11n model.

To illustrate the impact of the EMA module on the performance of the lightweight citrus leaf pest and disease detection model, the Grad-CAM [28] heatmap method was used to observe changes in the model's focus on target features after incorporating LSK Attention, SE Attention, GAM Attention, and EMA attention mechanisms into the bottleneck structure of the C3k2 module in the neck layer of the YOLOv11n model. The results presented in Figure 9 show that, in the absence of the attention module, the model did not properly highlight the citrus leaf disease targets in the image. However, after incorporating the attention modules, all models successfully concentrated on the target features, with the EMA module playing a more prominent role in guiding the model's attention to the relevant areas. This helped to minimize the focus on less important background regions, effectively reducing the loss of critical target-related information.

4.5.4. Ablation Experiment Results and Analysis of PEW-YOLO

This section validates the effectiveness of the PP-LCNet, GSConv, C3k2, and Wise-IoU loss function in the citrus leaf pest and disease detection task. The results of the experiments are presented in Table 4. In this table, “√” denotes the application of the respective improvement strategy, and “×” indicates its absence.

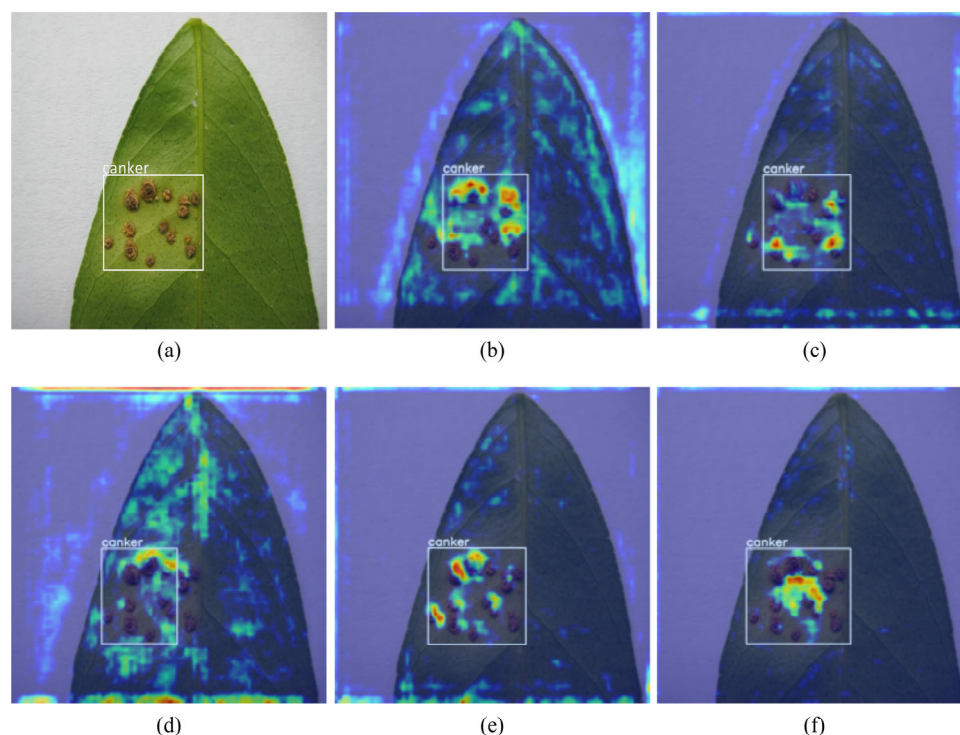


Figure 9. Visualization analysis of target heatmap under different attention mechanisms. (a) Ground truth. (b) Baseline. (c) LSK Attention. (d) SE Attention. (e) GAM Attention. (f) EMA Attention.

Table 4. Ablation results of the enhanced YOLOv11n model.

Model	PP-LCNet	GSConv	C3k2_EMA	Wise-IoU	P (%)	R (%)	mAP50 (%)	Params (M)	Time (ms)
YOLOv11n	×	×	×	×	83.8	79.1	86.8	2.58	1.3
YOLO-P	✓	×	×	×	81.6	75.6	84.2	1.86	1.2
YOLO-PG	✓	✓	×	×	82.8	80.6	85.2	1.86	1.4
YOLO-C	×	×	✓	×	84.0	79.9	89.2	2.47	1.4
YOLO-W	×	×	×	✓	83.9	80.7	88.6	2.58	1.7
YOLO-PGC	✓	✓	✓	×	84.0	80.5	88.4	1.75	1.5
PEW-YOLO	✓	✓	✓	✓	84.3	80.9	88.6	1.75	1.6

Table 4 shows that replacing the YOLOv11n backbone with the lightweight PP-LCNet network in the YOLO-P model leads to a decrease of 2.2%, 3.5%, and 2.6% in precision, recall, and mAP50, respectively. Additionally, the number of parameters is reduced by 27.9%, and the detection time drops by 0.1 ms. YOLO-PG, which introduces GSConv, exhibits a precision increase of 1.2%, a recall increase of 5.0%, and an mAP50 increase of 1.0% compared to those of the YOLO-P model. This demonstrates that the new lightweight PGNet backbone network can effectively utilize depthwise separable convolutions for efficient feature extraction. The YOLO-C model, enhanced by integrating the EMA module into the C3k2 module of the neck network, achieved a 0.2% improvement in precision, a 0.8% increase in recall, and a 2.4% boost in mAP50 compared to the YOLOv11n model. This demonstrates that the C3k2_EMA architecture effectively retains critical feature details during fusion and excels in identifying small-scale citrus leaf pest and disease targets. Further introducing the C3k2 module to the YOLO-PG model, the YOLO-PGC model is formed, showing a 1.2% increase in precision and a 3.2% rise in mAP50 over the YOLO-PG model. This indicates that combining the PGNet network with the C3k2 structure in the YOLOv11n framework significantly enhances the stability of the citrus leaf detection model. Furthermore, after introducing the Wise-IoU loss function, the YOLO-C model shows a notable improvement in its ability to detect citrus leaves in complex orchard

environments, with the mAP50 metric increasing to 88.6%. Finally, the PEW-YOLO model, which incorporates the PGNet network, C3k2_EMA structure, and Wise-IoU loss function, demonstrates a 0.5% increase in precision, a 1.8% rise in recall, and a 1.8% improvement in mAP50 over the YOLOv11n model, while reducing the parameter size by 32.2%. However, the detection time increases slightly to 1.6 ms/frame due to the increased depth of the model after introducing new modules. Despite the increased inference time, the model still meets real-time detection requirements. In conclusion, the PEW-YOLO model enhances detection performance and reduces the number of parameters, making it well suited for real-world citrus leaf pest and disease detection.

Figure 10 shows the mAP50 training curve for different YOLOv11n improved models. It can be seen that all models did not experience overfitting during training. Among them, the PEW-YOLO model's mAP50 curve converges rapidly and stabilizes, performing better in citrus leaf pest and disease detection.

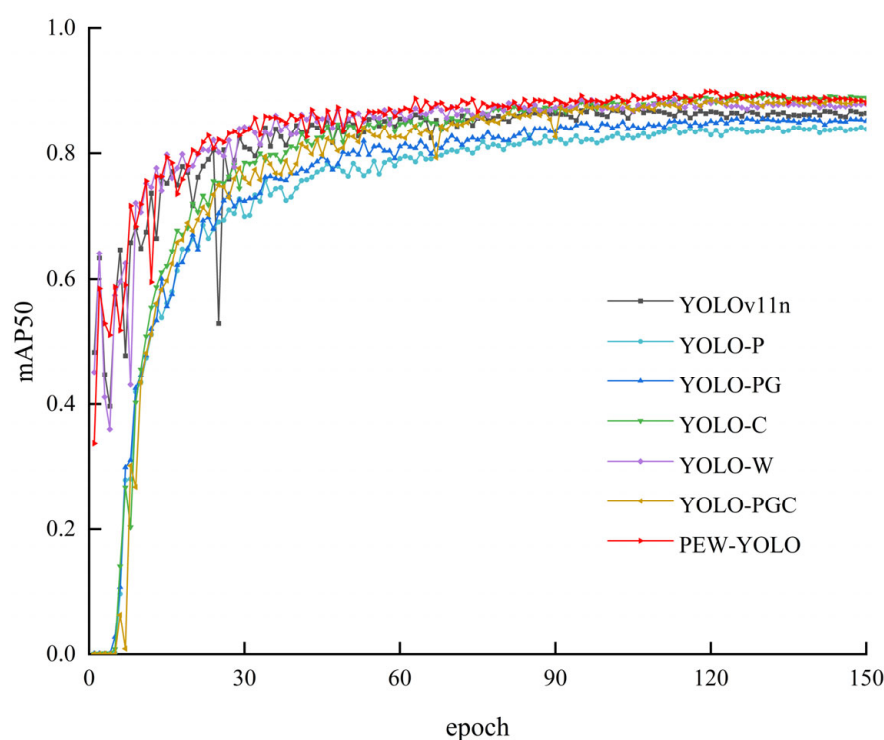


Figure 10. mAP50 training curve comparison.

During individual testing of each citrus leaf pest and disease image using the PEW-YOLO model, the predicted results and actual values for each citrus pest and disease category were recorded, resulting in a confusion matrix for analyzing the detection results and errors, as shown in Figure 11. It can be observed that there is misdetection between melanose and psyllid, mainly due to the similar features between pest-damaged and normal leaves, as well as the target becoming unclear in complex backgrounds. Therefore, a background category was added in the detection. Similarly, misdetections between Canker and Fuliginous were observed, mainly due to the small target size and the interference from background textures, which make disease-damaged leaves prone to misidentification. In conclusion, the effects of complex backgrounds and multi-scale objects significantly influence citrus leaf pest and disease detection.

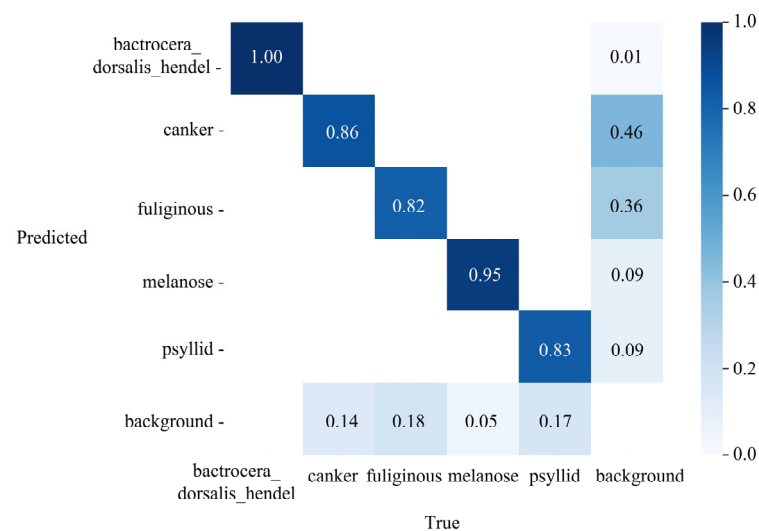


Figure 11. Confusion matrix of the PEW-YOLO model. Misclassification cases are further illustrated with example images in Figure 13, highlighting errors between visually similar categories such as melanose and fuliginous, as well as misidentifications between canker and background.

4.5.5. Performance Comparison with Mainstream Models

To assess the effectiveness of PEW-YOLO, a comparative analysis was performed on the citrus leaf pest and disease dataset, evaluating PEW-YOLO against other leading models, including YOLO-World, Swin Transformer, Faster R-CNN, RT-DETR, YOLOv7-tiny, YOLOv8n, YOLOv9-t, and YOLOv10n.

The results, presented in Table 5, indicate that PEW-YOLO outperforms the other models in precision, recall, and mAP50, while maintaining a smaller parameter size. Specifically, compared to YOLO-World, Swin Transformer, Faster R-CNN, RT-DETR, YOLOv7-tiny, YOLOv8n, YOLOv9-t, and YOLOv10n models, PEW-YOLO increased mAP50 by 2.3%, 3.2%, 8.7%, 6.9%, 3.4%, 2.7%, 0.7%, and 2.9%, respectively. It also reduced the model parameters by 2.30 M, 0.76 M, 26.57 M, 7.04 M, 4.27 M, 1.26 M, 0.87 M, 0.95 M, and 0.83 M, and decreased the detection time by 0.5 ms, 0.9 ms, 21.6 ms, 1.0 ms, 1.3 ms, 0.6 ms, 7.0 ms, 0.4 ms, and 0.3 ms, respectively. As shown in Figure 12, the radar chart representing the overall model performance shows that the PEW-YOLO model's area is the most filled, indicating that its performance in precision, recall, and mAP50 is closer to the ideal state compared to other models.

Table 5. Comparison of detection performance of mainstream target detection models.

Model	P (%)	R (%)	mAP50 (%)	Params (M)	Time (ms)
YOLO-World	79.1	81.9	86.3	4.05	2.1
Swin Transformer	81.5	79.6	85.4	2.51	2.5
Faster R-CNN	77.5	72.4	79.9	28.32	23.2
RT-DETR	78.7	76.1	81.7	8.79	2.6
YOLOv7-tiny	83.0	79.4	85.2	6.02	2.9
YOLOv8n	84.0	79.2	85.9	3.01	2.2
YOLOv9-t	82.9	80.8	87.9	2.62	8.6
YOLOv10n	80.7	80.5	85.7	2.70	2.0
YOLOv11n	83.8	79.1	86.8	2.58	1.3
PEW-YOLO	84.3	80.9	88.6	1.75	1.6

To evaluate the real-world detection performance of each model, several images of citrus leaf pest and disease scenes were selected for testing. The visualization results can be seen in Figure 13. From these results, it is evident that the YOLO-World, Swin Transformer, Faster R-CNN, RT-DETR, YOLOv7-tiny, YOLOv8n, YOLOv9-t, YOLOv10n, and YOLOv11n

models all experienced missed detections, particularly for fuliginous detection. In the case of *Bactrocera dorsalis* Hendel, the RT-DETR and YOLOv11n models incorrectly detected citrus leaf background as orange psyllid. In the detection of canker, the RT-DETR and YOLOv11n models were not precise enough to detect the small targets of ulcer disease, and both of them had detection box position deviation. In the detection of melanose, the RT-DETR model incorrectly detected normal citrus leaves as sandpeel disease leaves. For psyllid detection, YOLO-World, Swin Transformer, Faster R-CNN, RT-DETR, YOLOv8n, YOLOv9-t, YOLOv10n, and YOLOv11n had offset detection frame positioning. This is due to the formation of textured or notched features on citrus woodlouse leaves, which are easily confused with environmental features in the background, increasing the difficulty of detection.

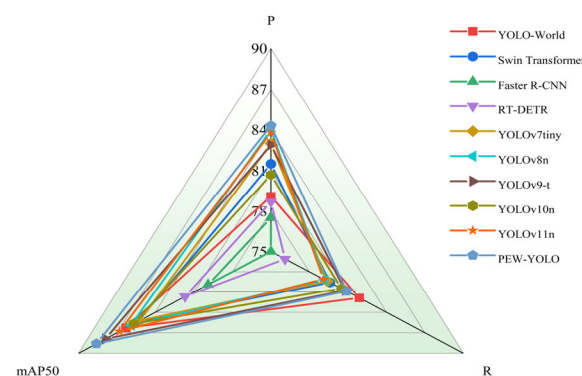


Figure 12. Radar chart of the comprehensive performance comparison of the mainstream models.

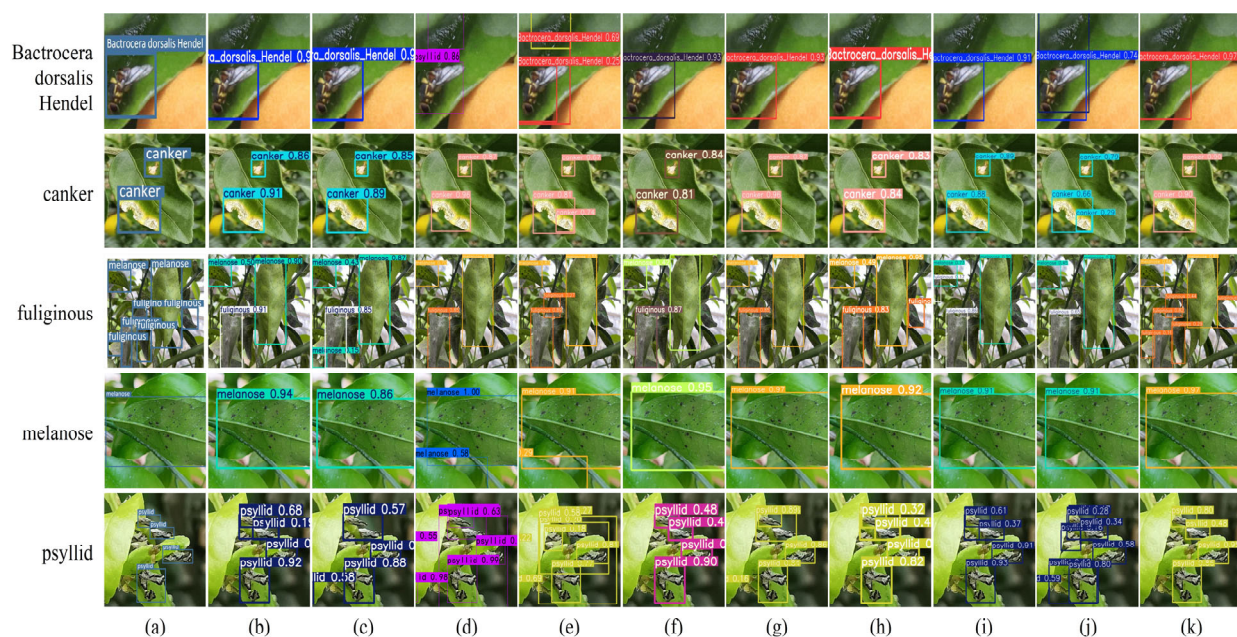


Figure 13. Comparison of visualization results of different models for citrus leaf pest and disease detection. (a) Ground truth. (b) Detection results of YOLO-World. (c) Detection results of Swin Transformer. (d) Detection results of Faster R-CNN. (e) Detection results of RT-DETR. (f) Detection results of YOLOv7tiny. (g) Detection results of YOLOv8n. (h) Detection results of YOLOv9-t. (i) Detection results of YOLOv10n. (j) Detection results of YOLOv11n. (k) Detection results of PEW-YOLO.

The reasons for these missed and incorrect detections are the models' limited ability to extract features for detecting pest and disease objects in complex backgrounds, as well as their vulnerability to interference from background noise in the images. This further

underscores the importance of adding attention mechanisms to improve model robustness. In summary, when compared with the nine other models, the PEW-YOLO model not only achieved superior citrus leaf pest and disease detection accuracy but also demonstrated better lightweight performance. Moreover, it exhibited stronger field detection capabilities for citrus leaf pests and diseases.

4.5.6. Evaluation on Rice Leaf Disease Dataset

To further validate the generalization ability of the proposed model, we evaluated its performance on a rice leaf disease dataset. This dataset consists of 5372 training images, 671 validation images, and 672 test images, collected under various natural environmental conditions, including backlighting, front lighting, sunny, and cloudy scenarios. It includes three object categories relevant to rice leaf diseases. We trained the model on this dataset for 150 epochs. As shown in Table 6, compared to the baseline YOLOv11n model, PEW-YOLO achieved a 1.0% improvement in precision, a 2.2% increase in recall, and a 0.7% boost in mAP50. Notably, the detection accuracy of small target categories such as Brown Spot and Leaf Smut also improved, demonstrating the effectiveness of the proposed C3k2_EMA structure in enhancing small object detection. These results indicate that PEW-YOLO delivers superior performance and exhibits strong robustness and effectiveness when applied to a different dataset involving leaf diseases in rice, thus supporting its generalizability across related agricultural domains.

Table 6. Detection results on rice leaf disease dataset.

Model	AP (%)			P (%)	R (%)	mAP50 (%)
	Bacteria Leaf Blight	Brown Spot	Leaf Smut			
YOLOv11n	99.5	98.0	98.7	98.4	96.0	98.7
PEW-YOLO	99.5	99.3	99.4	99.4	98.2	99.4

5. Conclusions

To enable accurate and rapid detection of citrus leaf pests and diseases under natural conditions, the proposed PEW-YOLO model achieves efficient detection of citrus leaf pests and diseases. Based on YOLOv11n, a lightweight PGNet backbone network was designed by introducing the new GSConv convolution to re-architect the PP-LCNet network. This leads to a reduction in model parameters and computational complexity, resulting in a more compact model with efficient feature extraction. Furthermore, the C3k2 module in the neck network was enhanced by incorporating the EMA module to improve feature fusion. Lastly, the Wise-IoU was used to enhance the model's performance in bounding box regression, enabling better learning of the positions and sizes of citrus pest and disease features. Ultimately, a lightweight, low-latency model with high detection accuracy was developed for citrus leaf pest and disease detection. The results of the experiment indicate that the PEW-YOLO model, with only 1.75M parameters, achieved a precision of 84.3% and mAP50 of 88.6%, demonstrating good detection performance on the citrus leaf pest and disease dataset. However, the algorithm's success is currently limited to five common citrus leaf pest and disease detection tasks. Given the diverse and imbalanced distribution of citrus pest and disease types, future work will focus on expanding the citrus pest and disease sample collection and supplementing pest information for further recognition and real-time detection research. Additionally, efforts will be made to transplant and improve the model for edge and mobile platforms, making it lighter and easier to deploy.

Author Contributions: Conceptualization, R.X. and L.W.; methodology, R.X.; software, R.X.; validation, R.X. and L.W.; formal analysis, R.X.; investigation, R.X.; resources, R.X.; data curation, R.X.; writing—original draft preparation, L.W.; writing—review and editing, L.W.; visualization, R.X.; supervision, R.X.; project administration, R.X.; funding acquisition, R.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Basic Scientific Research Business Project of Heilongjiang Provincial Department of Education (grant number 145209127). The authors acknowledge the anonymous reviewers for their helpful comments on the manuscript.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors would like to thank Qiqihar University for technical support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yan, K.; Song, X.; Jing, Y.; Xiao, J.; Xu, X.; Guo, J.; Zhu, H.; Lan, Y.; Zhang, Y. Citrus Huanglongbing Detection: A Hyperspectral Data-Driven Model Integrating Feature Band Selection with Machine Learning Algorithms. *Crop Prot.* **2025**, *188*, 107008. [\[CrossRef\]](#)
2. Chen, L.; Jia, Y.; Zhang, J.; Wang, L.; Yang, R.; Su, Y.; Li, X. Research on Citrus Fruit Freshness Detection Based on Near-Infrared Spectroscopy. *Processes* **2024**, *12*, 1939. [\[CrossRef\]](#)
3. Zhang, X.; Xun, Y.; Chen, Y. Automated identification of citrus diseases in orchards using deep learning. *Biosyst. Eng.* **2022**, *223*, 249–258. [\[CrossRef\]](#)
4. Mei, S.; Ding, W.; Wang, J. Research on the Real-Time Detection of Red Fruit Based on the You Only Look Once Algorithm. *Processes* **2023**, *12*, 15. [\[CrossRef\]](#)
5. Khanramaki, M.; Asli-Ardeh, E.A.; Kozegar, E.J.C.; Agriculture, E.I. Citrus pests classification using an ensemble of deep learning models. *Comput. Electron. Agric.* **2021**, *186*, 106192. [\[CrossRef\]](#)
6. Saini, R.; Garg, P.; Chaudhary, N.K.; Joshi, M.V.; Palaparthi, V.S.; Kumar, A. Identifying the source of water on plant using the leaf wetness sensor and via deep learning-based ensemble method. *IEEE Sens. J.* **2024**, *24*, 7009–7017. [\[CrossRef\]](#)
7. Patel, R.K.; Chaudhary, A.; Chouhan, S.S.; Pandey, K.K. Mango leaf disease diagnosis using Total Variation Filter Based Variational Mode Decomposition. *Comput. Electr. Eng.* **2024**, *120*, 109795. [\[CrossRef\]](#)
8. Leng, S.; Musha, Y.; Yang, Y.; Feng, G.J.A.S. CEMLB-YOLO: Efficient detection model of maize leaf blight in complex field environments. *Appl. Sci.* **2023**, *13*, 9285. [\[CrossRef\]](#)
9. Xing, S.; Lee, M.J.S. Classification accuracy improvement for small-size citrus pests and diseases using bridge connections in deep neural networks. *Sensors* **2020**, *20*, 4992. [\[CrossRef\]](#)
10. Lin, T.-L.; Chang, H.-Y.; Chen, K.-H. The pest and disease identification in the growth of sweet peppers using faster R-CNN and mask R-CNN. *J. Internet Technol.* **2020**, *21*, 605–614.
11. Liang, J.; Chen, X.; Liang, C.; Long, T.; Tang, X.; Shi, Z.; Zhou, M.; Zhao, J.; Lan, Y.; Long, Y.J.C.; et al. A detection approach for late-autumn shoots of litchi based on unmanned aerial vehicle (UAV) remote sensing. *Comput. Electron. Agric.* **2023**, *204*, 107535. [\[CrossRef\]](#)
12. Gao, L.; Zhao, X.; Yue, X.; Yue, Y.; Wang, X.; Wu, H.; Zhang, X.J.A.S. A Lightweight YOLOv8 Model for Apple Leaf Disease Detection. *Appl. Sci.* **2024**, *14*, 6710. [\[CrossRef\]](#)
13. Gangwar, A.; Dhaka, V.S.; Rani, G.; Khandelwal, S.; Zumpano, E.; Vocaturo, E. Time and Space Efficient Multi-Model Convolution Vision Transformer for Tomato Disease Detection from Leaf Images with Varied Backgrounds. *Comput. Mater. Contin.* **2024**, *79*, 117–142. [\[CrossRef\]](#)
14. Khanam, R.; Hussain, M. YOLOv11: An Overview of the Key Architectural Enhancements. *arXiv* **2024**, arXiv:2410.17725.
15. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
16. Yang, X.; Duan, L.; Zhou, Q. PileNet: A high-and-low pass complementary filter with multi-level feature refinement for salient object detection. *J. Visual Commun. Image Represent.* **2024**, *102*, 104186. [\[CrossRef\]](#)
17. Ali, S.G.; Wang, X.; Li, P.; Li, H.; Yang, P.; Jung, Y.; Qin, J.; Kim, J.; Sheng, B. Egdnet: An efficient glomerular detection network for multiple anomalous pathological feature in glomerulonephritis. *The Visual Computer* **2024**, *41*, 2817–2834. [\[CrossRef\]](#)
18. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Volume 11218, pp. 122–138.

19. Cui, C.; Gao, T.; Wei, S.; Du, Y.; Guo, R.; Dong, S.; Lu, B.; Zhou, Y.; Lv, X.; Liu, Q.J.a.p.a. PP-LCNet: A lightweight CPU convolutional neural network. *arXiv* **2021**, arXiv:2109.15099.
20. Wang, J.; Qin, C.; Hou, B.; Yuan, Y.; Zhang, Y.; Feng, W. LCGSC-YOLO: A lightweight apple leaf diseases detection method based on LCNet and GSConv module under YOLO framework. *Front. Plant Sci.* **2024**, *15*, 1398277. [[CrossRef](#)]
21. Zhao, Y.; Sun, C.; Xu, X.; Chen, J. RIC-Net: A plant disease classification model based on the fusion of Inception and residual structure and embedded attention mechanism. *Comput. Electron. Agric.* **2022**, *193*, 106644. [[CrossRef](#)]
22. Zhao, H.; Jia, J.; Koltun, V. Exploring self-attention for image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10076–10085.
23. Jia, X.; Jiang, X.; Li, Z.; Mu, J.; Wang, Y.; Niu, Y. Application of deep learning in image recognition of citrus pests. *Agriculture* **2023**, *13*, 1023. [[CrossRef](#)]
24. Dai, F.; Wang, F.; Yang, D.; Lin, S.; Chen, X.; Lan, Y.; Deng, X. Detection method of citrus psyllids with field high-definition camera based on improved cascade region-based convolution neural networks. *Front. Plant Sci.* **2022**, *12*, 816272. [[CrossRef](#)] [[PubMed](#)]
25. Tong, Z.; Chen, Y.; Xu, Z.; Yu, R. Wise-IoU: Bounding box regression loss with dynamic focusing mechanism. *arXiv* **2023**, arXiv:2301.10051.
26. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424.
27. Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient multi-scale attention module with cross-spatial learning. In Proceedings of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
28. Feng, Z.; Ji, H.; Daković, M.; Cui, X.; Zhu, M.; Stanković, L. Cluster-CAM: Cluster-weighted visual interpretation of CNNs' decision in image classification. *Neural Netw.* **2024**, *178*, 106473. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.