ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 2 - 2026

Real-time Pest Detection in Complex Forestry Scenarios: An Adaptive Deep Learning Approach for Sustainable Forest Health Monitoring

Juhu Li[1,2], *Shihao Li*[1,2], *Yuli Xu*[1,2], *Jia Lu*[1,2] *and Feng Yang*[1,2,*]

[1] School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China

[2] Engineering Research Center for Forestry-Oriented Intelligent Information Processing of National Forestry and Grassland Administration, Beijing 100083, China

SUMMARY: *Timely monitoring and control of forest pests is of great significance for maintaining forest health, carbon sequestration and biodiversity. However, due to the influence of environmental factors such as rain, fog and insufficient light, it is often very difficult to track forest pests in the field, which leads to the deterioration of image quality and is difficult to use traditional methods for detection. In this paper, a new deep learning method PCSNet is proposed. In order to solve the problem of forest pest detection under harsh conditions, Chain of Mind Prompt Adaptive (CPA) enhancement module is added to the network, which can perform autonomous super-resolution enhancement for different degradation types of images. Wavelet Transform convolution (WTConv) and wavelet transform Down-sampling module (ADown) were proposed to speed up model convergence and optimize network performance. In addition, for edge devices, we have developed PCSNet-Light, a pruned and refined version that enables smaller model sizes with high accuracy. Experiments show that when using Raspberry PI 5 as the hardware platform, the mAP50 of PCSNet-Light reaches a high level of 96.6%, while maintaining a real-time detection speed of 15.67FPS while greatly reducing the number of parameters by 5 million. This micro-resolution strategy not only enhances the accuracy of pest detection, but also supports efficient and scalable pest monitoring systems that can help achieve more sustainable forest management actions.*

KEYWORDS: *Sustainable forest management; Forest pests and diseases detection; Complex scenes; Edge computation; Deep learning; Real-time monitoring; Image quality enhancement*

1 Introduction

Forest pests and diseases seriously hinder the photosynthetic growth process of trees, and even degrade the whole forest area (Lierop et al., 2015), which destroys the function of the ecosystem, reduces the carbon sequestration capacity in the affected forest, and exacerbates the loss of wood. Further exacerbating world climate change, timely and accurate artificial insect monitoring is therefore a key need for forest management. However, the traditional method relies heavily on the knowledge and experience of human experts. Although it has a certain role, this method is inefficient and can not meet the needs of large-scale insect monitoring.

Some early studies attempted to overcome the above problems by applying machine

*fengyang@bjfu.edu.cn
https://doi.org/10.65102/is2026998
Published: 30 April, 2026

learning techniques. For example, Ebrahimi et al. (2017) used the region index and enhanced color index of support vector machines as a method for pest detection. Similarly, Maharlooei et al. (2017) also used image processing techniques, including region of interest (ROI) determination, separation, and brightness adjustment, to develop a detection and counting technique for soybean aphids on leaves. However, affected by external conditions, it can not be fully applied to the complex and changeable forest environment.

Deep learning has shown great potential to achieve more advanced automatic pest identification capabilities. In 2022, Ye and his team proposed to use multi-scale attention and the Unet framework to cooperate with drones to process single-pixel remote sensing images, aiming to improve the early diagnosis level of pine tree weakness. In addition, Liu B. et al. also used skip connection and spatial pyramid pooling layer to optimize the DETR structure in their study in 2023. Similarly, in order to better realize the purpose of small pest target detection, Liu D. By introducing transfer learning and DyHead module to optimize the performance of YOLOv5, et al. achieve 98.1% accuracy in the test of 31 forest insects. Dinca et al. used a series of DL methods based on YOLOv8, Faster R-CNN, RetinaNet, SSD, and FCOS to achieve a more comprehensive pest detection system. It can significantly improve the identification rate and efficiency of orchard diseases and pests.

There have also been related studies on the problem that pests are not easily detected in woodland ecosystems. For example, J. Liu and Wang(2024) designed the Multi-source Dataset (PDD) and pepper network to deal with the pest identification problem under complex backgrounds, and used fine-grained multi-modal features and contrast learning to enhance the robustness of the model, and finally achieved the result of mAP50=91.93. Similarly, D. Sun et al.(2024) applied SimAM attention module to more challenging scenarios in tobacco pest detection, and the generalization ability still needs to be improved. M. Li et al.(2025) constructed an almond pest dataset in a multi-light environment, and constructed a targeted module to improve the robustness. Thirdly, L. Zhang(2024) uses the iterative attention mechanism and alternating loss strategy to improve the detection accuracy in small objects, which is better than DETR and other detectors in dense scenes. Z. Li et al.(2024) implemented pest detection in tea garden based on CNN transfer learning algorithm. The accuracy is above 98%. guo et al.(2022) proposed a swin transformer based segmentation method to detect rare tree pests, which has better performance than traditional models in occluded, low-contrast, and noisy scenes.

Although significant breakthroughs have been made, most research still relies on a single reinforcement method, which is not applicable to all complex environments and often reduces its effectiveness. In order to solve this problem, people begin to focus on adaptive image enhancement technology. For example, Liu X et al. (2023) proposed a method called LAE-Net, which combines local adaptive kernel selection with feature adaptation to improve the quality of low-light images. Zhou et al.(2023) effectively overcome the degradation in underwater images by using multiple interval sub-histogram equalization methods based on statistical inference. Chen et al.(2023) also designed an artificial neural network AIENet specifically for aerial forest images, and used visual dilation methods at different scales to recover high-resolution information. Similarly, Fang et al. (2024) also advocate fusing instance normalization and a Transformer based global attention mechanism to handle multiple degradation cases, and their proposed TAENet is thus derived. Finally, Park et al. (2023) created a model called ADMS that can automatically identify and filter out unknown degraded images and has achieved excellent results on various mixed degradation datasets. However, there is still a long way to go when it is applied to the identification of forest pests and diseases. Degradation conditions such as rain, fog, equipment stains, blurriness, and poor light can degrade the quality of photos taken in forests, which can affect the improvement of

recognition accuracy (Hu et al., 2024; F. Wang et al., 2023; S. Wang et al., 2025; X. Wang et al., 2021; Zhang et al., 2022). At the same time, some common augmentation algorithms have high computational resource requirements (Giakoumoglou et al., 2023; Liang et al., 2023; Peng et al., 2023; Qian et al., 2023; B. Sun et al., 2025; Liang et al., 2023). Z. Yang et al., 2024), so these methods are not suitable for the use of integrated devices.

In this paper, based on YOLOv11, a PCSNet forest pest recognition model is proposed. On this basis, the adaptive image enhancement algorithm and object detection ability are integrated to solve the complex detection challenges in forest scenes. We introduce a CPA enhancement layer at the front end of the backbone network, and use a hint mechanism to achieve the purpose of automatic enhancement of inferior images. At the same time, we also use wavelet transform convolution (WTConv) to alleviate the problem of too large parameters caused by expanding the receptive field of traditional CNN. To further reduce the size of the model, we incorporate a lightweight down module. For embedded application scenarios, we use the techniques of pruning and knowledge distillation to build a smaller version, PCSNet-Light, which can complete the intelligent monitoring and management tasks of forest pests and diseases in complex environments while ensuring high accuracy.

In order to overcome the main obstacles in the actual ecosystem monitoring process, this paper proposes a network-based solution PCSNet(Pest Detection in Complex Scene Network) to deal with the challenge of image quality deterioration. The innovation of this paper is to construct such a general architecture, which is not only resistant to disturbances in a variety of complex situations, but also more than a simple superposition of existing algorithms. More precisely, the joint application of the CPA module with WTConv and ADown modules provides us with an optimized and parameter-efficient basis for self-tuning degradation strengthening. Our final contribution to this strong foundation is PCSNet-Light, a highly compressible form that translates high-accuracy detection of complex environmental ecology into devices with limited resources at the boundaries of the environment.

2 Materials and Methods

2.1 Construction of the FOR31-CS Dataset

High-quality datasets capturing forest pests under complex environmental conditions are still scarce, mainly because field acquisition is limited by weather variations, illumination variations, sensor contamination, and defocus. To better approximate real-world monitoring scenarios, 31-CS (forest pest dataset with 31 categories in complex scenes) is constructed by applying four representative degradations (rain, fog, low light, and blur) to an existing, well-annotated dataset of 31 categories of forest pests. The design produces datasets containing both clean and degraded observations, thus supporting more robust model learning and evaluation under realistic forestry conditions.

2.1.1 Dataset Selection

For forest pest monitoring, B. Liu et al. (2022) developed a dataset consisting of 31 pest species collected in natural environments to reflect real forest conditions. A representative sample is shown in Fig. 1. A fine-grained labeling scheme was used for the dataset since pest morphology can vary greatly in different life stages such as larvae, pupae, nymphs.

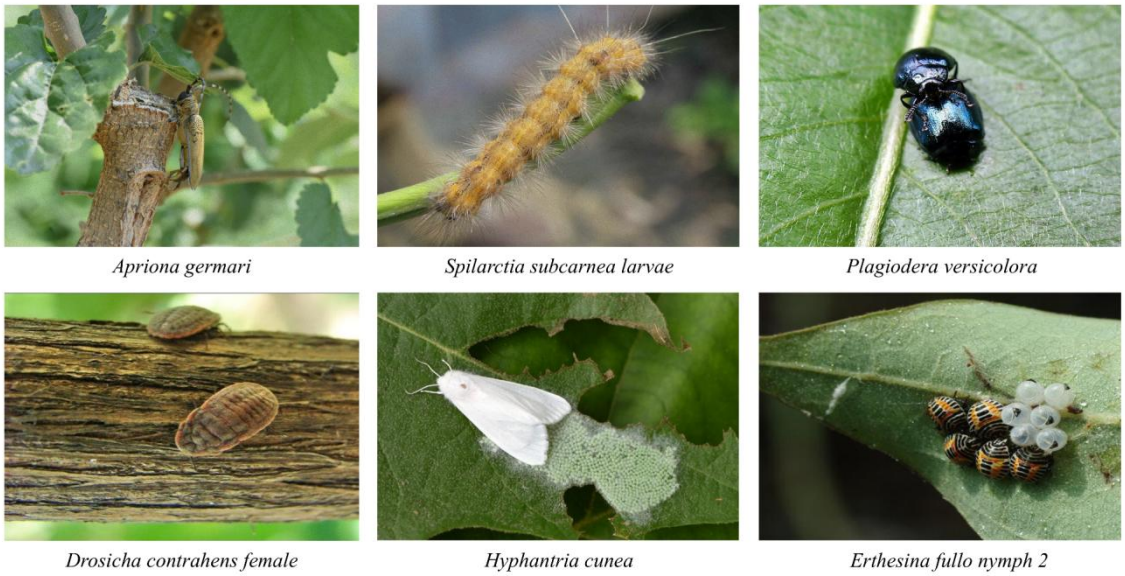

Figure 1: Examples from the Dataset

In addition, species with significant sexual dimorphism, such as cardinals, are classified into various categories to maintain their inherent visual diversity. After the annotation optimization process, the dataset consists of 6446 training images, 14177 annotated instances, 717 validation images, and 1578 annotated instances.

2.1.2 Complex Scenes Processing

Image acquisition in forest environments is inherently variable. Rain and fog often reduce contrast and produce scattering effects, lens contamination or defocus causes blur, cloudy or night conditions cause low brightness and reduced detail -all of which can greatly degrade the performance of pest identification. To simulate these challenges in a controlled and reproducible way, four complex scenarios were simulated: rainy weather, foggy weather, low illumination, and blur.

To further illustrate the construction process of the FOR31-CS dataset, Fig. 2 shows the processing path of the original forest pest data after it enters the complex scene generation flow. The original 31 types of forest pest images were first checked and samples were sorted to ensure that different insect states, gender differences and fine-grained categories could be retained in the dataset. Subsequently, the dataset was randomly divided into five subsets, one of which retained the original image state and was used to represent normal woodland acquisition conditions. The remaining four subsets were subjected to rain, fog, low light, and blur to simulate the image degradation that is common in field monitoring. Rainy scenes are mainly represented by striped occlusion and local brightness disturbance, foggy scenes weaken the contrast between the target edge and the background, low-light scenes compress the texture and color information of pests, and blurred scenes correspond to the detail loss caused by lens out of focus, motion jitter or remote acquisition. After the above processing, the normal samples and the four types of degraded samples were re-merged to form the FOR31-CS dataset containing a variety of complex environmental disturbances. This process enables the model training phase to access both clear images and degraded images, which provides a data basis for the subsequent verification of the detection robustness of PCSNet in complex forest environments.

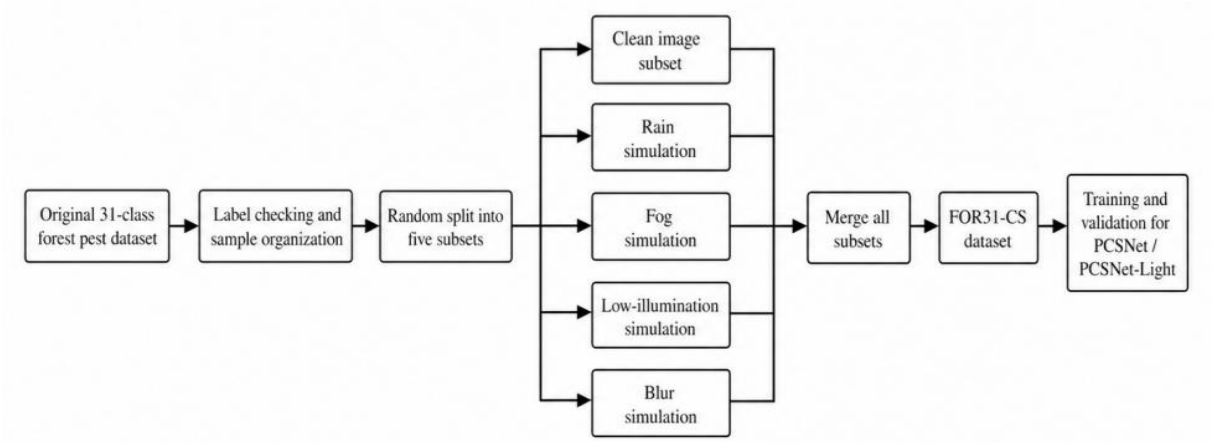

Figure 2: Process of FOR31-CS dataset construction and complex scene generation

Specifically, the original dataset is randomly divided into five subsets. One subset retains the original image and the remaining four subsets are processed separately to simulate one type of degradation. Thus, the generated FOR31-CS dataset contains a mixture of normal and degraded images, enhancing the model's ability to generalize to complex field conditions. An example of the processed image is shown in Fig. 3, and the process is as follows:

(1)Rainy Weather Processing

To simulate rainfall, we first generate random Gaussian noise and transform it via directional rotation and Gaussian smoothing to mimic streak-like raindrop structures. The synthesized rain layer is then alpha-blended with the original image:

$$I_{rain} = I_{original} \cdot (1 - \alpha) + (N \cdot \alpha) \quad (1)$$

where I_{rain} is the rain-augmented image, $I_{original}$ is the original image, N represents Gaussian noise, and α is the transparency of the raindrops.

(2)Foggy Weather Processing

Fog effects were simulated using an atmospheric scattering model (An et al., 2023), as illustrated in Fig. 3(b):

$$I_{fog} = I_{original} \cdot e^{-\beta \cdot d} + A \cdot (1 - e^{-\beta \cdot d}) \quad (2)$$

In this formula, I_{fog} denotes the fog-simulated image, $I_{original}$ is the scene radiance (original image), and A is the global atmospheric light, set to 0.5. The scattering coefficient β is defined based on resolution: for high-resolution images, $\beta = 0.04 + 0.01i$ with $i \in [0,4]$; for other images, $\beta = 0.05 + 0.01i$ with $i \in [0,9]$. The variable d denotes scene depth.

(3)Low Illumination Processing

Some forest pests are primarily active at night. For instance, Hyphantria cunea (fall webworm), which belongs to the genus Hyphantria in the family Arctiidae, tends to remain still during the day and becomes active at night for mating and oviposition (Schowalter & Ring, 2017). meaning that images are often captured under low-light conditions. To better reflect the natural behaviour of such pests, low-light effects were applied to a portion of the dataset.

Low-light effects were achieved by adjusting the image contrast followed by gamma correction, as shown in Fig. 3(c). The transformation is defined by:

$$I_{low-light} = 255 \cdot \left(\frac{\alpha \cdot I_{original} + \beta}{255}\right)^{\gamma} \quad (3)$$

where α is the contrast factor (set between 0.75 and 0.85), β is the brightness offset (set to 0), and $\gamma$is the gamma value (ranging from 2 to 4).

(4)Blur Processing

Blur commonly arises from defocus, motion, or long-range capture when pests occupy only a small region in the field of view. To simulate this degradation, we apply Gaussian blur Fig. 3(d):

$$I_{blur} = I_{original} * G \quad (4)$$

where G denotes a two-dimensional Gaussian kernel, and the convolution kernel size is set to (9, 9).

By incorporating common field degradations into a controlled benchmark, FOR31-CS supports the development and evaluation of pest monitoring models that are more reliable for operational forest protection and sustainable management.

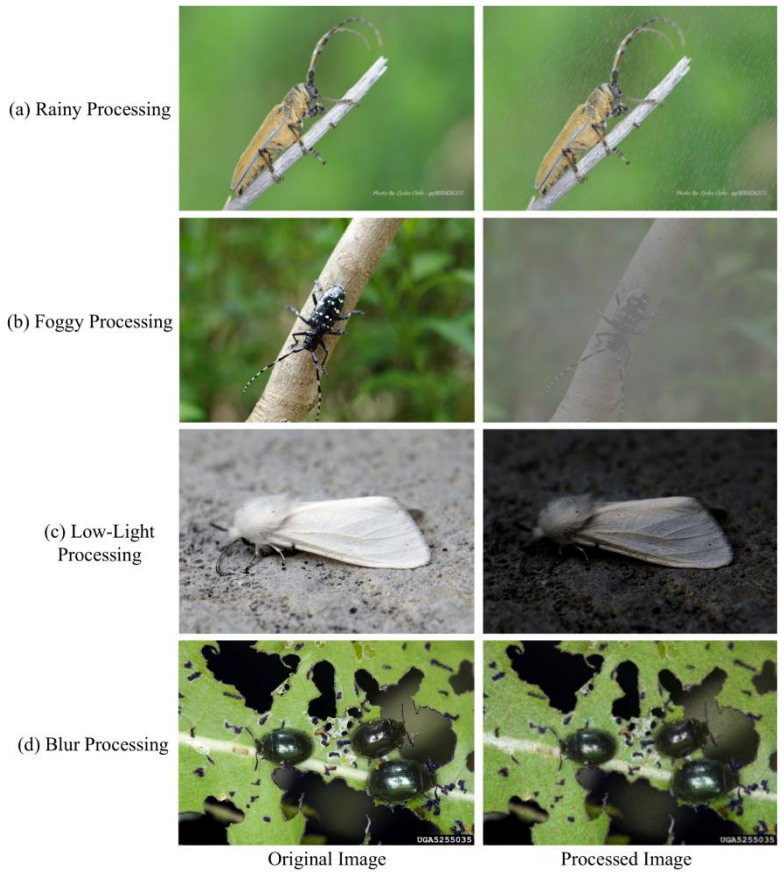

Figure 3: Example of FOR31-CS Processing

2.2 PCSNet Model

YOLO, as a salient object detection method, solves the problem of object recognition in a look-at-a-time manner by directly predicting bounding boxes and class probabilities across the entire image (Redmon et al. 2016). Based on this development trend, YOLOv11 has

become the latest achievement in the field of real-time object detection, ensuring both efficient performance and high accuracy (Alkhammash, 2025). An improved C3k2 module is designed to enhance the feature extraction function and a new attention mechanism (C2PSA) is proposed, so it can be applied to the forest pest recognition task.

Although it has good baseline performance, its robustness decreases in the face of image degradation in complex environments. To this end, an improved detection model pcsnet based on yolov11 is proposed. See Fig. 4 for the global framework of pcsnet.

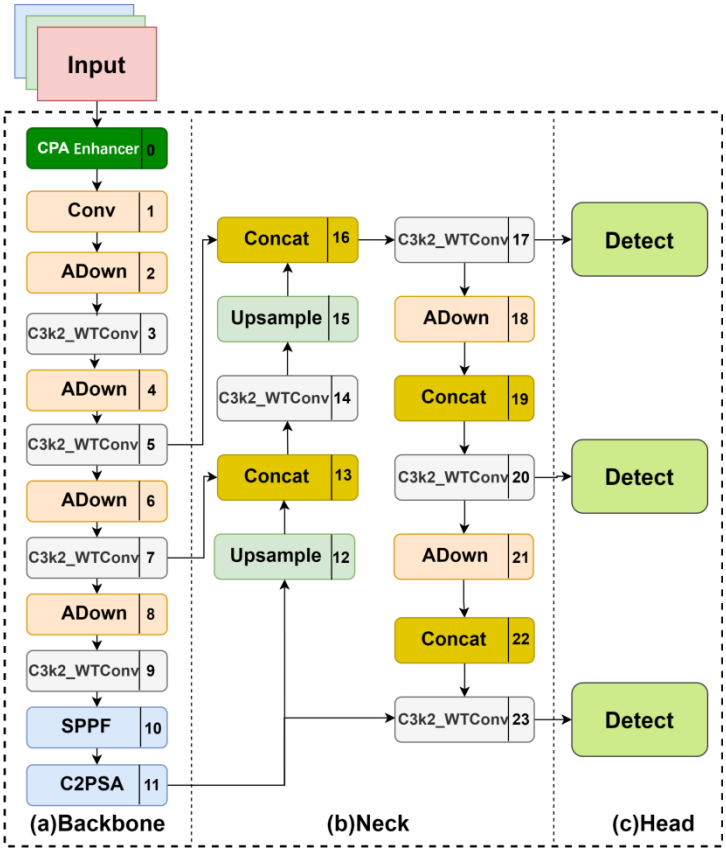

Figure 4: PCSNet Network Structure

PCSNet contains the following four improvements: (1) the CPA adaptive enhancement module is added to the previous core network to compensate for those images that do not know the extent of their degradation; (2) WTConv is used to expand the receptive field, enhance the background representation and reduce the parameter explosion problem; (3) Lightweight down module for downsampling; The computational complexity is reduced and it is easy to be applied under restricted conditions. Thirdly, the structure reduction and knowledge distillation methods are used to obtain the lightweight PCSNet-Light that can be used for landing.

2.2.1 Integration of the CPA Enhancement Module for Image-Adaptive Augmentation

In forest environments, pest images are often captured under complex and unpredictable conditions, and degradation types and intensities are usually unknown in advance. This uncertainty makes a single fixed augmentation strategy suboptimal and may even amplify artifacts of some scenes. To improve the robustness under mixed degradation, we integrated the CPA (Chain of Mind Cue Adaptation) augmentation module (Y. Zhang et al., 2025) into the detection backbone. CPA is inspired by chain of thought prompts in natural language

processing, which guide the model to perform multi-step, task-oriented reasoning. Here, the notion of prompting is extended to the visual domain to guide the enhancement behavior based on the estimated degradation features.

In detail, CPA adopts a series of guidance methods to automatically adjust the intensification strategy to the attenuation cues of the input (Lu et al., 2024), which makes it better able to deal with conditions such as rain, fog, low light and blur. This module is divided into two main parts: CoT-Prompt Generator (CGM) and Content-based Prompt Unit (CPB). Firstly, CGM will parse the input image, identify the relevant decay information, and regenerate the corresponding guide word. Operationally, the initial visual cue (P3) is translated into a set of parameters that can be learned. The following multi-level cues, including P2 and P1, are then generated step by step from transposed convolutional sequences and forced activations. During training, these bootstrap words are globally optimized throughout the pipeline, which all depends on the loss of downstream pest detection, and can therefore dynamically react to the decaying nature of the input without the need for special climate type markers or other explicit supervision. Injecting these prompters inside the CPB and combining image features enables adaptive augmentation to cope with attenuation situations.

Structurally, CPA starts from receptive field perceptual convolution (RFAConv) to extract low-level representations (Wei et al., 2025). A four-stage hierarchical encoder-decoder then progressively encodes the input into compact latent features and reconstructs high-resolution representations. Finally, RFAConv is applied again to generate the enhanced features, which are combined with the original input by residual addition to produce the enhanced output. The overall structure is shown in Fig. 5.

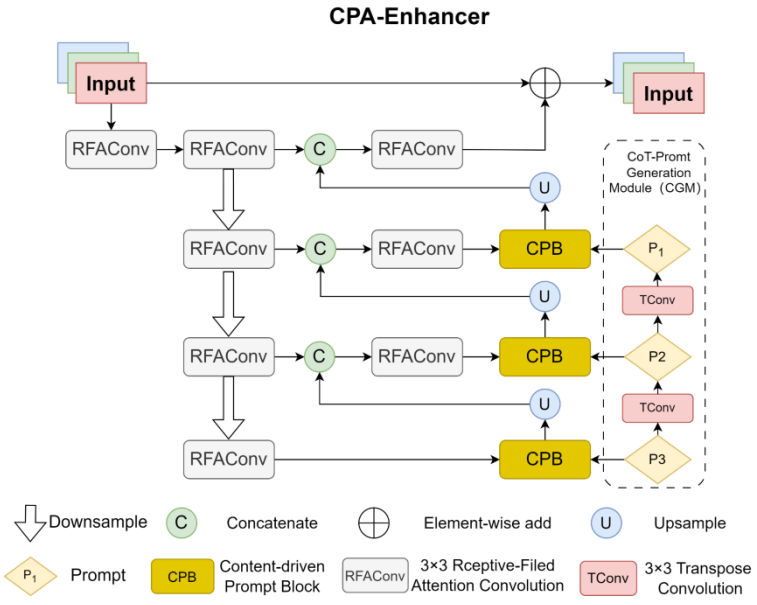

Figure 5: The overall structure of the CPA-Enhancer Module

2.2.2 Introduction of the ADown Downsampling Module to Reduce Computational Load

In this paper, an improved downsampling method is used to design the ADown module, which can reduce the space size of the feature map while ensuring that the key features are not lost, and effectively reduce the number of model parameters and computation, which is suitable for forest pest monitoring scenarios with high accuracy requirements. The traditional CNN model

often produces a lot of redundant information in its indirect feature map, which increases the number of parameters and computational cost of the model. Our Downsample Block is improved for this step downsample process to reduce this redundant information.

Initially, the ADown module is integrated into the YOLOv9 model as part of the multi-branch design, which makes use of AvgPool and MaxPool to improve its feature extraction capabilities (c.y. Wang et al., 2025). In detail, the module performs two operations on the input feature map: first, it is shrunk by doubling the size using average pooling, and then it is split into two branches along the dimension of the channel, one of which is used to extract features and reduce the dimension through a 3×3 convolution. The other branch first goes through Max-Pooling and then a 1×1 Pointwise Convolution to further strengthen the nonlinear feature representation and further compress the dimension. Finally, the results of these two branches are used as the output results of this module. See Fig. 6 for the ADown module construction method.

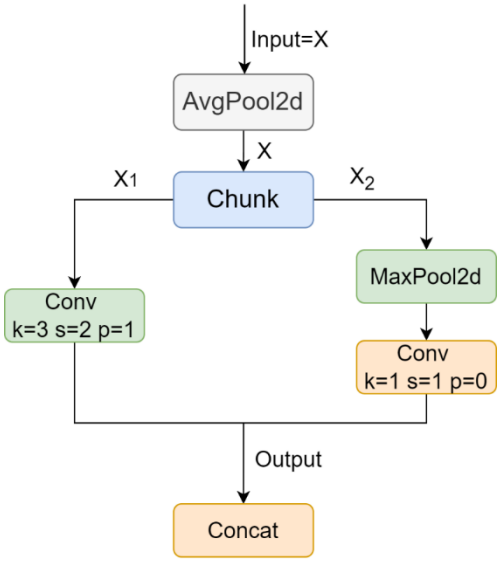

Figure 6: The Structure of ADown Module

Compared with traditional convolution-based downsampling methods, ADown achieves more comprehensive feature extraction by combining average pooling and Max pooling, thus minimizing information loss. In addition, its multi-branch design improves the flexibility of the network and the ability to capture multi-scale features. This lightweight downsampling mechanism significantly enhances the representation power of the model, making it particularly effective in detecting small pest targets.

2.2.3 Introduction of the WTConv Module Based on Wavelet Transform for Enhanced Feature Extraction

There are a lot of background clutter and multi-texture complex target objects with different sizes in the image of forest pests and diseases. Expanding the receptive field is beneficial to better model the environment and reduce interference between harmful insects and similar visual background patterns. However, simply increasing the convolution kernel will bring a large number of parameters and a surge in computational cost, which is not ideal for real-time or edge devices. Therefore, we introduce here the wavelet transform Convolution (WTConv) technique, which exploits wavelet feature deconvolution to extend the effective sensing range and alleviate the problem of too large parameters (Tian et al., 2023;Yao et al., 2022).

WTConv processes the feature maps by replacing part of the standard convolution

operation with Haar wavelet transform. For each input feature map X, a 1D Haar transform can be performed on its spatial axis by exploiting two fixed sets of kernels: [1,1]/√2 and [1, -1]/√2 depth convolutions, followed by a process of downsampling with a step of 2. When this step is extended to 2D scenes (that is, considering the factors of height and width), the four fixed filters can form the depth convolution of stride-2, as shown in Equation (5), where the first filter is the low frequency part, and the other three filters are the high frequency part in the opposite direction.

$$f_{LL} = \frac{1}{2}\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}, f_{LH} = \frac{1}{2}\begin{bmatrix}1 & -1\\ 1 & -1\end{bmatrix}, f_{HL} = \frac{1}{2}\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}, f_{HH} = \frac{1}{2}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} \quad (5)$$

As a result, each input path is broken down into a dense set of frequency-aware representations that preserve the basic structure (low frequency components) while capturing finer texture or boundary information (high frequency components). These secondary features then go through a learned convolution step and are further integrated together (via inverse wavelet transform) to produce the final feature image. The aim is to determine the parameter overhead of using frequency separation to achieve WTConv better feature extraction in complex forest areas and better robustness against aging false targets while maintaining high efficiency (cf. C.Liu et al.(2025)). The first-order WTConv structure is shown in Fig. 7.

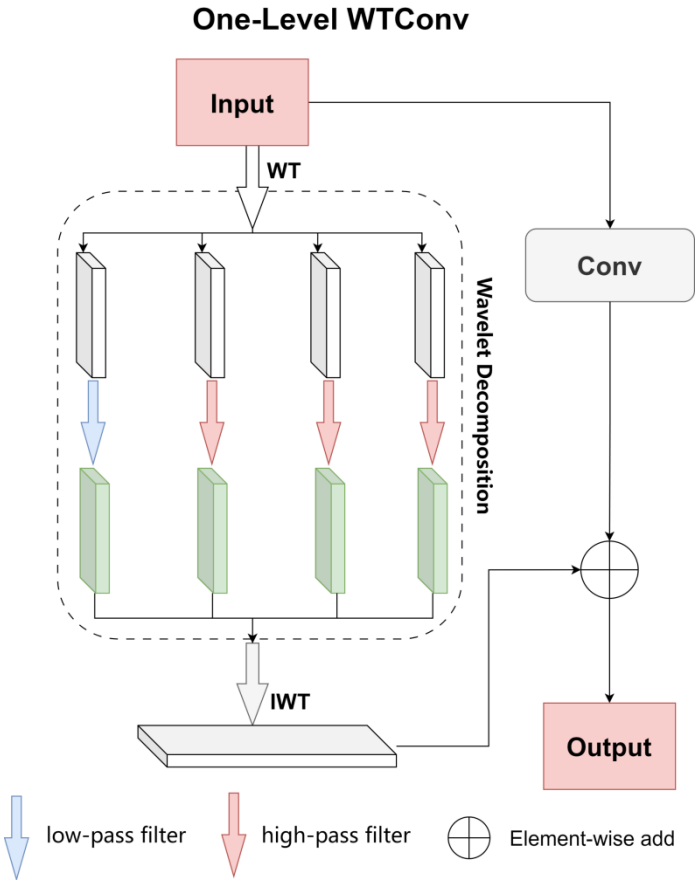

Figure 7: The structure of the 1-level WTConv

2.2.4 Model Pruning and Knowledge Distillation for Mobile Deployment

In order to further improve the application effect of pcsnet on embedded platforms, pcsnet is optimized by combining lightweight pruning and knowledge distillation to obtain

PCSNet-Light, which effectively reduces the network size and reasoning overhead under the premise of ensuring detection performance.

Firstly, structured channel pruning was integrated to eliminate redundant calculations and reduce the complexity of the model. A channel importance evaluation method based on Batch Normalization (BN) parameter is adopted. Specifically, the channel scaling factor γ in each BN layer is sorted by its absolute value and the channels with smaller magnitude are pruned. The importance score of the i channel is defined as follows.

$$\mathrm{Score(i)} = |\gamma_i| \tag{6}$$

where γ_i denotes the scaling factor of the i channel in the BN layer.

To compensate for the potential performance degradation caused by pruning, knowledge distillation based on soft labels is further adopted. A high-precision, unpruned PCSNet model (teacher) guides the pruned PCSNet model (student) by transferring knowledge to help the student model better approximate the original output distribution. The pruning and distillation processes are shown in Fig. 8. In this setting, the temperature-scaled Kullback-Leibler (KL) divergence is used as the distillation loss:

$$L_{KD} = \mathrm{KL}\left(\mathrm{Softmax}\left(\frac{z^T}{T}\right)||\mathrm{Softmax}\left(\frac{z^S}{T}\right)\right) \tag{7}$$

where z^T and z^S are the logits from the teacher and student models, respectively, and T is the temperature parameter that softens the probability distribution. The KL divergence measures the distance between the two distributions.

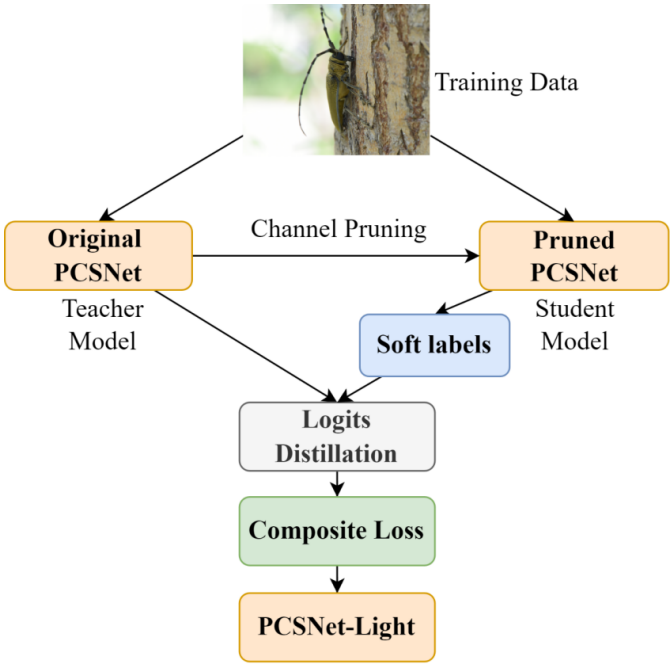

Figure 8: Flowchart of Pruning-Distillation Joint Optimisation

The total loss used for training is defined as:

$$L = \alpha \cdot L_{det} + \beta \cdot L_{KD} \tag{8}$$

where α and β are the weights for the original object detection loss and the distillation loss, respectively. This formulation ensures that the student model maintains detection performance while effectively incorporating the teacher's knowledge.

2.3 Evaluation Metrics

To measure the performance of a proposed model, we use the following metrics: Precision (p), Response Rate (r), mean Average Precision (map), frames per second (fps), Floating-point operations per second (gflops), and number of parameters.

Where, precision, recall, average precision (ap) and average precision (map) are defined as follows.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

$$\text{AP}_\text{i} = \int_0^1 \text{P}_\text{i}(\text{R})\, \text{dR} \tag{11}$$

$$\text{mAP} = \frac{\sum_{\text{i}=1}^{\text{N}} \text{AP}_\text{i}}{\text{N}} \text{AP} \tag{12}$$

where TP, FP and FN are the number of true positives, false positives and false negatives respectively.

In simple terms, map50 means that the average ap at the iou threshold is 0.5; map50:95 is the average ap value for different iou thresholds at 0.05 intervals. gflops refers to models that perform billions of floating-point operations per second and is used to express the computational complexity of the model.

3 Results

3.1 Experimental Setup

The study was conducted on a Linux server platform with a device setup containing 90GB of RAM, an NVIDIA RTX 4090 GPU (24GB of video memory), and an Intel® Xeon® Platinum 8352V CPU. We use CUDA 12.4 to implement parallel computing, the main programming language is Python 3.8.10, and the deep learning part uses PyTorch 2.0.0 framework. In this configuration, we can ensure the computing performance required for model training and random simulation and data validation.

3.2 Ablation Experiments

To evaluate the effectiveness of the proposed PCSNet model and its three key modules cpa, down, and wtconv, an ablation study was performed. In the experimental setup, "√" means active module and "×" means disabled module. Table 1 summarizes the results based on the FOR31-CS validation set.

Table 1: Ablation Experiment

Model	CPA Enhancer	ADown	WTConv	mAP50/%	mAP50:95/%	FLOPS/G	Params/M
YOLOv11n	×	×	×	95.1	78.1	6.5	2.60
Model A	√	×	×	96.2	79.3	19.2	3.09
Model B	×	√	×	95.5	77.8	5.3	2.11
Model C	×	×	√	96.1	80.2	6.4	2.80
Model D	√	√	×	96.6	81.4	18.0	2.61
Model E	√	×	√	96.6	82.0	19.1	3.03
Model F	×	√	√	96.3	80.3	6.4	2.31
PCSNet	√	√	√	96.8	82.0	19.1	2.82

The ablation results in Table 1 confirm the individual and joint contributions of each component. Compared with the baseline YOLOv11n, the separate integration of CPA module (model A) improves mAP50:95 from 78.1% to 79.3%, indicating that the degradation adaptive enhancement effectively improves the feature quality under complex environmental conditions. In addition, the introduction of the ADown module (model B) mainly improves efficiency, reducing the computational complexity from 6.5 GFLOPs to 5.3 GFLOPs (a reduction of 18.5%) and 18.8% fewer parameters with only a slight decrease in accuracy. Using the WTConv module alone (model C) further improves accuracy, increasing mAP50:95 by 2.1% (to 80.2%) while slightly reducing it by 0.1 GFLOPs, indicating that receptive field expansion is achieved without additional computational burden. Furthermore, the two-module combination (models D, E, and F) consistently exhibits an effective balance between accuracy and efficiency.

In summary, under the joint use of the three modules, the final PCSNet has more than 96.8% performance improvement on mAP50 and reaches 82.0% on Map50:95. Its parameter count is 2.82M, which is only 0.22M larger than the lightweight baseline, proving that it can achieve high accuracy while reducing model size. In this way, PCSNet becomes a practical and effective method to monitor the occurrence of forest diseases and pests.

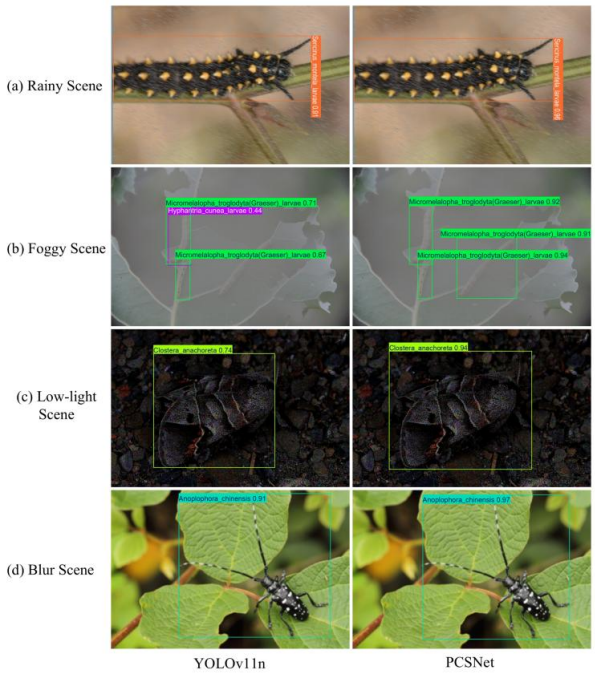

Figure 9: Comparison of detection results between YOLOv11n and PCSNet

To further illustrate this performance under challenging conditions, a visual comparison between baseline YOLOv11n and the final PCSNet is performed. The evaluation was performed on images of four degraded scenes (rain, fog, low light, and blur) from the FOR31-CS validation set (Fig. 9). It is clear that PCSNet consistently yields higher prediction confidence in all scenarios. For example, in the fog-day scene (Fig. 9b), YOLOv11n exhibits missed and false detections, while PCSNet accurately identifies all pest targets with precise bounding boxes, highlighting its robust advantage in complex environments.

3.3 Comparison with Representative Detection Networks

In order to verify the effectiveness of PCSNet in practical applications, ablation experiments are carried out on the FOR31-CS dataset, and the proposed lightweight version of PCSNet is compared with the classical detection networks: YOLOv8n, YOLOv10n, YOLOv11n. The rest are detection methods based on RCNN models (Faster R-CNN & Faster R-CNN) and dynamic R-CNN(H. Zhang et al., 2020) and the Transformer based DETR detector RT-DETR(Zhao et al., 2024).

We measure the expressiveness of the model using several evaluation metrics, including mAP, Precision, Recall, and GFLOPs. All models were trained on the FOR31-CS dataset based on the same hyperparameter setting. The results of the comparison are presented in Table 2, where the optimal performance for each category is marked in bold. In addition, the changes of mAP and loss of different categories of models during training are shown in Fig. 10.

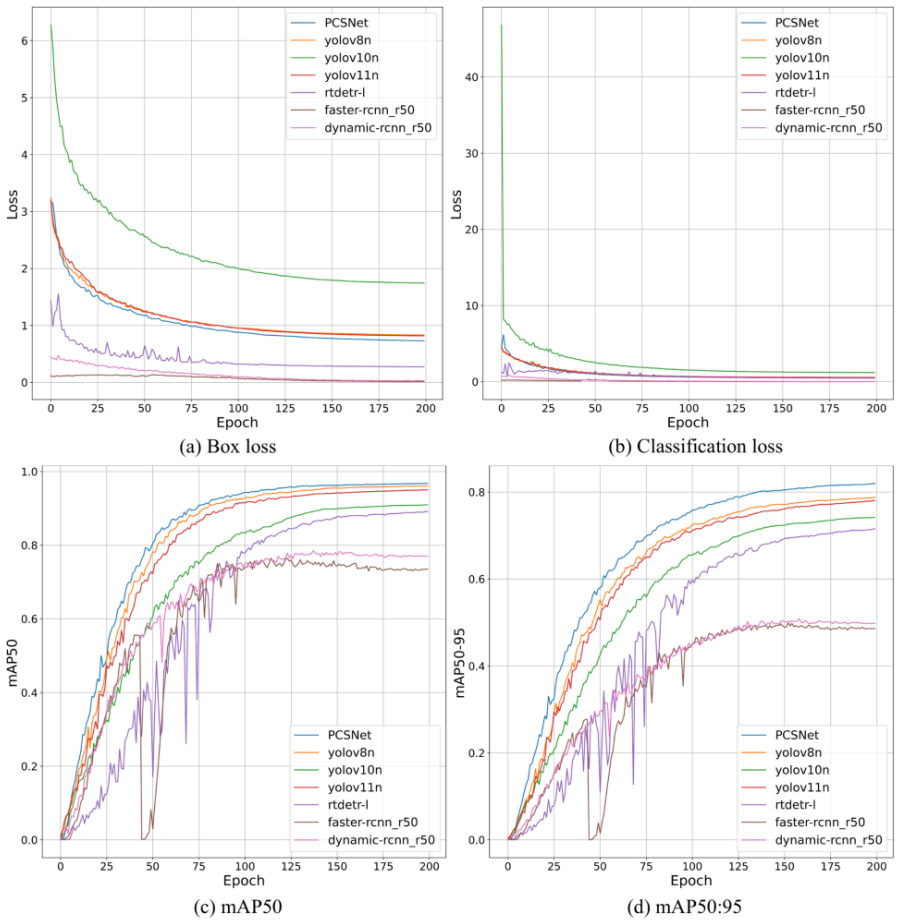

Figure 10: Comparison of indicators in common model training processes

Table 2: Performance Comparison of Various Object Detection Networks

Model	Precision/%	Recall/%	mAP50/%	mAP50:95/%	FLOPS/G	Params/M
YOLOv8n	92.8	87.0	94.2	74.5	8.2	3.02
YOLOv10n	85.1	84.7	91.0	74.2	8.5	2.72
YOLOv11n	94.3	88.6	95.1	78.1	6.5	2.60
Faster RCNN	46.5	43.8	73.5	48.6	56.3	41.50
Dynamic R-CNN	53.3	50.2	77.0	49.8	56.3	41.50
RT-DETR-l	89.6	83.1	89.1	71.6	108.1	32.87
PCSNet	95.5	93.6	96.8	82.0	7.0	2.82

As shown in Table 2 and Fig. 10, PCSNet achieves the best performance on several key evaluation metrics. Although the YOLO and RCNN series are relatively lightweight in structure, RCNN-based models usually exhibit weak accuracy and struggle to accurately detect forest pest targets. Transformer-based models, such as those in the DETR family, are competitive in terms of accuracy. For example, RT-DETR-l achieves an accuracy of 89.6% and a mAP50 of 89.1%.

However, RT-DETR relies on multi-head self-attention and deep encoder-decoder design, resulting in significant parameter and computational overhead (Di et al., 2024). Therefore, it is less efficient for resource-constrained deployment scenarios, such as embedded forest monitoring platforms.

In contrast, PCSNet can maintain a high level of recognition accuracy despite its relatively compact model size. Even with only 2.82M parameters, it achieves up to 95.5% accuracy and 96.8% mAP50 value. Compared with all other models, PCSNet performs well in multiple evaluation criteria, conforms to the computational constraints of embedded system, and optimizes and improves the heavy-load modeling.

3.4 Model Interpretability Analysis

To ensure that our model improvements are effective, we leverage Gradient-weighted Class Activation maps (Grad-CAM), which visually illustrate how the network makes decisions and highlight key parts of an image. This helps people understand how a deep learning network processes its input data (Selvaraju et al.,(2017). In the visualization, the red regions represent regions that contribute significantly to the model's prediction - the brighter the color, the greater the contribution.

As shown in the heat map in Fig. 11, although the baseline model YOLOv11n is able to identify pest categories, its prediction confidence is relatively low. This is mainly due to image degradation under complex environmental conditions, which hinders the ability of the model to accurately focus on pest areas. Therefore, a high activation response is also observed in the background region (i.e., the red region), which indicates that the model is affected by environmental noise and background clutter, which reduces the detection accuracy.

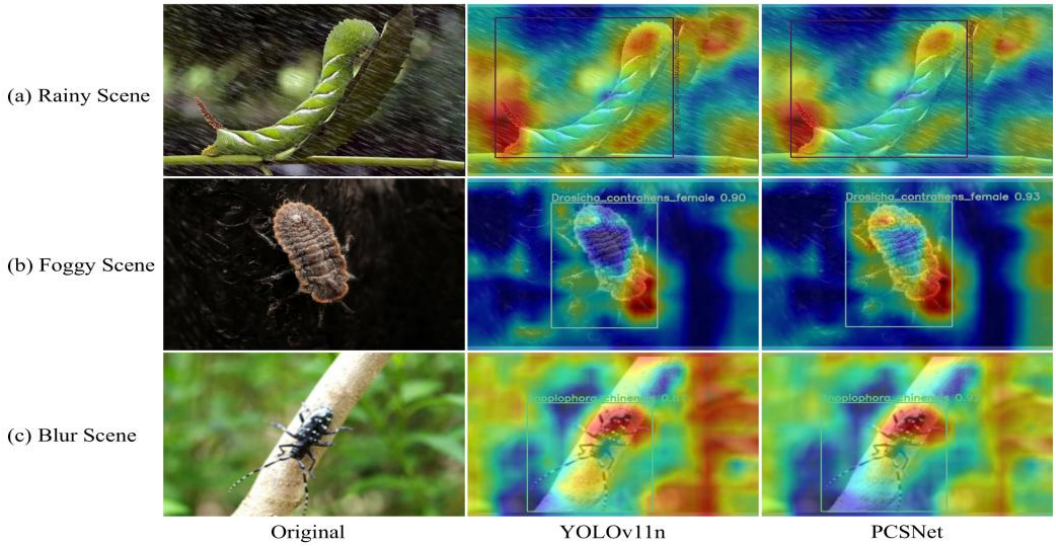

Figure 11: Comparison of Heatmap Detection between YOLOv11n and PCSNet

In contrast, the improved PCSNet enhances the feature representation through adaptive image enhancement to guide the model to focus more effectively on the actual pest area. The Grad-CAM visualization clearly shows that PCSNet significantly reduces the high response in irrelevant regions. The red area is more focused and restricted to the location of the pest, indicating better allocation of attention and higher confidence in the prediction of the target area. These observations suggest that PCSNet provides more reliable cues for target localization under real forest noise, which is beneficial for practical pest monitoring and decision support.

3.5 Lightweighting Experiment

To adapt to the requirements of embedded deployment, we evaluate the results of PCSNet in two lightweight Settings, namely pruning only as well as pruning with knowledge distillation. We fixed the pruning ratio to 0.3, the distillation temperature to 4 and the loss weight ratio between detection and distillation objects to 7:3. The value of the actual pruning ratio was determined after an initial sensitivity analysis. As can be seen from Table 3, when the pruning rate is too small (0.2), the compression space of the model will be restricted. Anything greater than 0.4 or 0.5 results in a large and uneven detection accuracy (especially on mAP50:95) that can't be recovered by further knowledge distillation. Therefore, the pruning rate is set to 0.3, which can achieve good detection results while ensuring model simplification.

Table 3: Sensitivity analysis of different pruning rates for PCSNet-Light

Pruning rate	Precision/%	Recall/%	mAP50/%	mAP50:95/%	GFLOPs	Params/M	Performance interpretation
0.0	95.5	93.6	96.8	82.0	7.0	2.82	Original PCSNet without pruning
0.2	95.4	93.3	96.7	81.8	6.4	2.37	Accuracy remains stable, but compression is limited
0.3	95.3	93.1	96.6	81.6	5.8	2.10	Best balance between accuracy and lightweight design
0.4	93.8	90.7	94.9	78.4	5.1	1.74	Detection accuracy drops evidently after stronger pruning
0.5	91.6	87.9	92.8	74.2	4.5	1.39	Excessive pruning causes severe loss of feature representation

As shown in Table 4, pruning reduces computation by 1.2 GFLOPs and parameters by 0.72M, but also results in a significant decrease in recall and mAP50:95. When knowledge distillation is applied, the pruned model recovers most of the lost accuracy (mAP50 = 96.6%, MAP50:95 = 81.6%) while maintaining a compact footprint (5.8 GFLOPs, 2.10M parameters). Overall, the distillation pruning model (PCSNet-Light) achieves a strong balance between accuracy and efficiency and outperforms YOLOv11n under similar lightweight constraints.

Table 4: Performance comparison of pruning and knowledge distillation strategies

Model	Pruning	Knowledge distillation	Precision/%	Recall/%	mAP50/%	mAP50:95/%	GFLOPs	Params/M
YOLOv11n	×	×	94.3	88.6	95.1	78.1	6.5	2.60
PCSNet	×	×	95.5	93.6	96.8	82.0	7.0	2.82
PCSNet-Pruned	√	×	94.8	90.9	95.8	79.6	5.8	2.10
PCSNet-Light	√	√	95.3	93.1	96.6	81.6	5.8	2.10

3.6 Embedded Device Deployment

To demonstrate practical feasibility beyond offline benchmarks, an embedded deployment experiment is conducted and real-time performance is evaluated under domain-like conditions. This section aims to answer a key question in operational forestry monitoring: can the proposed method provide accurate and stable detection on low-power hardware when images are captured in complex environments: rain, fog, low light, and blur?

3.6.1 Deployment setup and reproducible pipeline

Pcsnet-light is deployed on the Raspberry PI 5, the representative edge device of the portable forest monitoring system, and compared with the typical lightweight detectors (YOLOv8n, YOLOv11n) and the original PCSNet. The video stream is captured using an external camera and the inference results are displayed in real time on a portable screen overlaid with bounding boxes and category labels to support on-site inspection and quick decision making. This setup approximates the use of real operations: continuous video input, limited computational resources, and the need for low-latency responses.

For efficiency and deployment compatibility, the model weights are converted to NCNN for lightweight inference on ARM-based hardware. To ensure a consistent speed-accuracy trade-off for different methods, the input is resized during inference, and the long edge is fixed at 544 pixels. Under this unified configuration, the detection accuracy and running speed are tested on the embedded platform.

3.6.2 Qualitative results in complex forestry scenarios

Fig. 12 shows representative examples of real-time detection in four complex scenarios. Visually, PCSNet-Light exhibits more reliable localization and classification when degradation is present, especially in low-light and hazy conditions, where background clutter and reduced contrast often lead to missed detections or misclassification by the baseline lightweight model. These qualitative results are consistent with the design motivation: CPA adaptively enhances degraded inputs, while WTConv's integrated frequency-aware representation helps retain discriminative cues under challenging textures and noise.

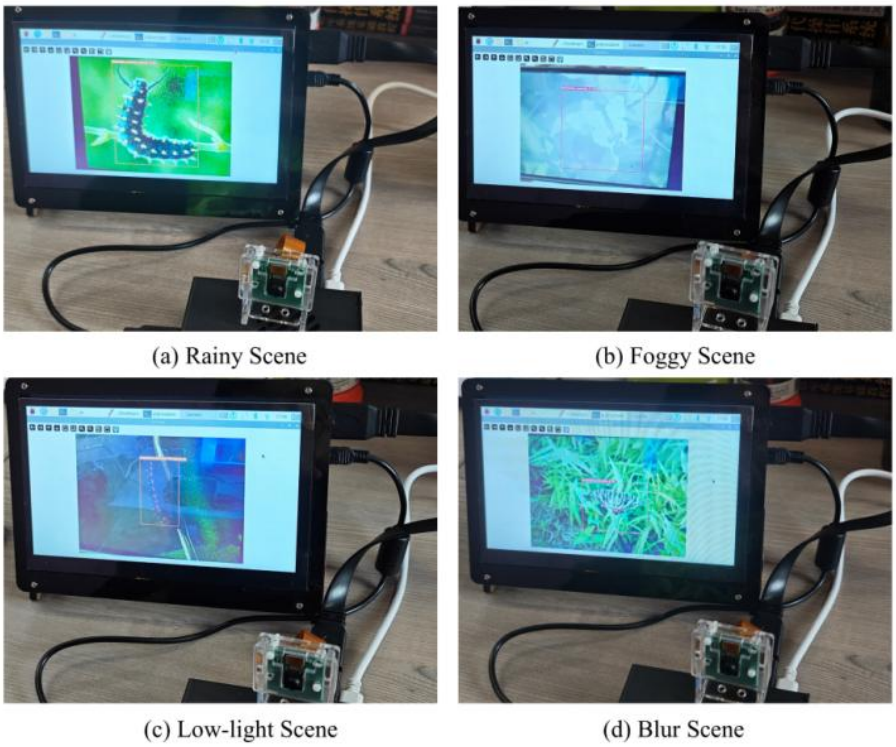

Figure 12: Real-time Detection Performance in Different Scenarios

3.6.3 Quantitative embedded performance and practical implications

Table 5 summarizes the results of our quantitative analysis. Compared with the YOLOv11n version running on Raspberry PI 5, our proposed PCSNet-Light achieves 1.5% increase in mAP50 value and 3.7% increase in MAP50:95, while reducing the number of model parameters by 0.50M. What is more remarkable is that PCSNet-Light successfully improves the inference rate from 2.14 to 15.67FPS, which meets the requirements of continuous monitoring in embedded systems.

Table 5: Quantitative performance comparison on the Raspberry Pi 5 platform

Model	mAP50/%	mAP50:95/%	GFLOPs	Params/M	FPS
YOLOv8n	94.2	74.5	8.2	3.02	12.86
YOLOv11n	95.1	77.9	6.5	2.60	13.53
PCSNet	96.8	82.0	7.0	2.82	10.94
PCSNet-Light	96.6	81.6	5.8	2.10	15.67

These results highlight an important practical advantage: PCSNet-Light not only improves accuracy under laboratory conditions, but also maintains a favorable precise-efficiency balance when deployed in an end-to-end embedded pipeline. This balance is critical in forest pest management, as monitoring systems often need to operate continuously, operate with limited power, and provide timely alerts for early intervention. By demonstrating stable real-time performance in complex scenarios on edge hardware, PCSNet-Light provides a deployable solution for practical forest pest monitoring.

Overall, the embedded deployment results validate PCSNet-Light as a practical detector that combines robustness to environmental degradation with real-time inference for low-power devices.

4 Discussion

Efficient and real-time detection of forest pests is an important component of sustainable forest management. While many deep learning models have been proposed for agricultural and forestry pest detection, there are still few studies that simultaneously address "adaptive feature enhancement in complex, unknown degraded environments" and "end-to-end real-time deployment on extreme edge hardware". Thus, a direct and identical comparison with the same studies in the literature is challenging. However, by systematically comparing our findings with the representative object detection paradigms evaluated in this study, the unique advantages and practical value of PCSNet become very apparent.

Compared with traditional lightweight frameworks and RCNN-based models, which usually show weak accuracy in multi-degradation conditions such as fog and low illumination, and it is difficult to accurately detect forest pest targets, PCSNet shows superior robustness. On the other hand, Transformer-based models, such as those in the DETR family (e.g. RT-DETR-l), are competitive in terms of theoretical accuracy, achieving 89.6% accuracy and 89.1% mAP50. However, RT-DETR relies on multi-head self-attention and deep encoder-decoder design, integrating a large number of parameters (32.87M) and computational overhead (108.1 GFLOPs). Therefore, the efficiency is very unfavorable for resource constrained deployment scenarios, such as embedded forest monitoring platforms. In sharp contrast, the proposed PCSNet-Light achieves 96.6% mAP50 with only 2.10M parameters and 5.8 GFLOPs, successfully running at 15.67 FPS on a Raspberry PI 5. This proves that the proposed integrated strategy -combining CPA, WTConv, and down as architectural foundations followed by a custom pruning-distillation pipeline -effectively Bridges the gap between high-accuracy theoretical algorithms and practical low-power forestry applications.

While the results are encouraging, there are a number of pressing issues that need to be addressed. The first is that the current database is relatively small and biased towards common pest categories. This means that the ability to identify rare or inadequate pest species may be limited. Future work should focus on expanding the scope and diversity of the database in order to improve generalization and handle long-tail problems (L. Liu et al., 2019). In addition, while the method simulation used for the FOR31-CS dataset provides a controlled infrastructure, it is generally accepted that the difference between simulated degradation and real-world disadvantages (e.g., physical water droplets on the lens surface, changes in leaf reflectance) is still significant in practice (M. Yang et al., 2023). Therefore, the performance of the model is likely to show a certain decline after moving from the simulated degradation state to the real degradation state. To fill this gap, future research will focus on capturing and evaluating photos of real pests in harsh climates to further close the gap. Finally, the ability to resist interference in more complex situations such as severe occlusion, partial visibility, and target cutting has also become a key point (Ouardirhi et al., 2024). Generative modeling techniques are possible to help synthesize more occluded samples and improve the ability of the model to identify objects in complex background noise (k. wang et al., 2017).

5 Conclusion

This study proposes a highly robust and efficient forest pest detection network named PCSNet, whose main goal is to cope with the performance degradation caused by common degradation phenomena in complex forest environments. We integrate the CPA adaptive enhancement module, the wavelet transform convolution technology of WTConv and the lightweight downsampling module down, so that PCSNet can ensure adaptive feature enhancement while

suppressing excessive parameter increase. A lightweight model PCSNet-Light is designed by using structured sparse pruning strategy and knowledge transfer technology.

Comprehensive evaluation on the FOR31-CS dataset shows that the PCSNet architecture significantly outperforms leading detectors such as YOLOv11n with a mAP50 performance of 96.8%. What is especially remarkable is that the lightweight PCSNet-Light can be effectively installed into the Raspberry PI 5 device, and can achieve a real-time detection rate of 15.67 FPS, while occupying only 2.10 megabytes of memory. This approach allows us to maximize the compactness of the model while maintaining high accuracy. Overall, PCSNet and PCSNet-Light provide us with an accurate and adaptive pest detection scheme for various environmental conditions, and also provide practical technical support for data-driven methods to monitor forest health and implement pest management strategies.

Funding

This work was supported by the National Natural Science Foundation of China under Grant 32071775.

CRediT authorship contribution statement

Juhu Li is a Professor and Department Head at the School of Information Science and Technology, Beijing Forestry University. He received his Ph.D. from Beijing University of Posts and Telecommunications, and his B.S. and M.S. from Lanzhou University. His research focuses on acoustic signal processing, deep learning, and IoT technology. A master's supervisor, he has published over 20 SCI/EI papers and led 10+ research projects.

Shihao Li: was born in Zhengzhou, Henan, P.R. China, in 2001. He obtained a bachelor's degree from Zhengzhou University of Light Industry, China. Currently, he is studying at the School of Information Science and Technology, Beijing Forestry University, Beijing, China. His main research directions are forestry pest recognition and object detection.

Yuli Xu:was born in Xinyang, Henan, P.R. China, in 1997. She obtained a bachelor's degree from Zhengzhou University of Light Industry, China. Currently, she is studying at the School of Information Science and Technology, Beijing Forestry University, Beijing, China. Her main research direction is agricultural pest image recognition.

Jia Lu: was born in Beijing, P.R. China, in 1997. She obtained a bachelor's degree from Beijing Forestry University, China. Currently, she is studying at the School of Information Science and Technology, Beijing Forestry University, Beijing, China. Her main research direction is software engineering.

Feng Yang: Feng Yang received his B.S. degree from Chongqing University and his Ph.D. degree from Beihang University. He is currently an Associate Professor at the School of Information Science and Technology, Beijing Forestry University. His research interests include multimodal learning, remote sensing data interpretation, and intelligent information processing for forestry and grassland.

References

[1] Alkhammash, E. H. (2025). Multi-Classification Using YOLOv11 and Hybrid YOLO11n-MobileNet Models: A Fire Classes Case Study. Fire, 8(1), 17. https://doi.org/10.3390/fire8010017

[2] An, S., Huang, X., Cao, L., & Wang, L. (2023). A comprehensive survey on image dehazing for different atmospheric scattering models. Multimedia Tools and Applications, 83(14), 40963–40993. https://doi.org/10.1007/s11042-023-17292-8

[3] Chen, Z., Wang, C., Zhang, F., Zhang, L., Grau, A., & Guerra, E. (2023). All-in-one aerial image enhancement network for forest scenes. Frontiers in Plant Science, 14, 1154176. https://doi.org/10.3389/fpls.2023.1154176

[4] Di, Y., Phung, S. L., Berg, J. V. D., Clissold, J., Bui, L., Le, H. T., & Bouzerdoum, A. (2024). Bio-DETR: A Transformer-based Network for Pest and Seed Detection with Hyperspectral Images. 2024 International Joint Conference on Neural Networks (IJCNN), 1–8. https://doi.org/10.1109/IJCNN60899.2024.10650195

[5] Dinca, M.-A., Popescu, D., Ichim, L., Angelescu, N., & Pinotti, C. M. (2024). Decision fusion-based system to detect two invasive stink bugs in orchards. Smart Agricultural Technology, 9, 100548. https://doi.org/10.1016/j.atech.2024.100548

[6] Ebrahimi, M. A., Khoshtaghaza, M. H., Minaei, S., & Jamshidi, B. (2017). Vision-based pest detection based on SVM classification method. Computers and Electronics in Agriculture, 137, 52–58. https://doi.org/10.1016/j.compag.2017.03.016

[7] Fang, W., Wang, C., Li, Z., Grau, A., Lai, T., & Chen, J. (2024). TAENet: Transencoder-based all-in-one image enhancement with depth awareness. Applied Intelligence, 54(15–16), 7509–7530. https://doi.org/10.1007/s10489-024-05569-w

[8] Giakoumoglou, N., Pechlivani, E. M., & Tzovaras, D. (2023). Generate-Paste-Blend-Detect: Synthetic dataset for object detection in the agriculture domain. Smart Agricultural Technology, 5, 100258. https://doi.org/10.1016/j.atech.2023.100258

[9] Guo, Y., Gao, J., Wang, X., Jia, H., Wang, Y., Zeng, Y., Tian, X., Mu, X., Chen, Y., & OuYang, X. (2022). Precious Tree Pest Identification with Improved Instance Segmentation Model in Real Complex Natural Environments. Forests, 13(12), 2048. https://doi.org/10.3390/f13122048

[10] Hu, X., Li, X., Huang, Z., Chen, Q., & Lin, S. (2024). Detecting tea tree pests in complex backgrounds using a hybrid architecture guided by transformers and multi-scale attention mechanism. Journal of the Science of Food and Agriculture, 104(6), 3570–3584. https://doi.org/10.1002/jsfa.13241

[11] Li, M., Tao, Z., Yan, W., Lin, S., Feng, K., Zhang, Z., & Jing, Y. (2025). Apnet: Lightweight network for apricot tree disease and pest detection in real-world complex backgrounds. Plant Methods, 21(1), 4. https://doi.org/10.1186/s13007-025-01324-5

[12] Li, Z., Sun, J., Shen, Y., Yang, Y., Wang, X., Wang, X., Tian, P., & Qian, Y. (2024). Deep migration learning-based recognition of diseases and insect pests in Yunnan tea under complex environments. Plant Methods, 20(1), 101. https://doi.org/10.1186/s13007-024-01219-x

[13] Liang, Z., Li, C., Zhou, S., Feng, R., & Loy, C. C. (2023). Iterative Prompt Learning for

Unsupervised Backlit Image Enhancement. 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 8060–8069. https://doi.org/10.1109/ICCV51070.2023.00743

[14] Lierop, P. V., Lindquist, E., Sathyapala, S., & Franceschini, G. (2015). Global forest area disturbance from fire, insect pests, diseases and severe weather events. Forest Ecology and Management, 352, 78–88. https://doi.org/10.1016/j.foreco.2015.06.010

[15] Liu, B., Jia, Y., Liu, L., Dang, Y., & Song, S. (2023). Skip DETR: end-to-end Skip connection model for small object detection in forestry pest dataset. Frontiers in Plant Science, 14, 1219474. https://doi.org/10.3389/fpls.2023.1219474

[16] Liu, B., Liu, L., Zhuo, R., Chen, W., Duan, R., & Wang, G. (2022). A Dataset for Forestry Pest Identification. Frontiers in Plant Science, 13, 857104. https://doi.org/10.3389/fpls.2022.857104

[17] Liu, C., Chen, K., Wang, N., Shi, W., & Jia, N. (2025). A lightweight multi-scale feature fusion method for detecting defects in water-based wood paint surfaces. Measurement, 253, 117505. https://doi.org/10.1016/j.measurement.2025.117505

[18] Liu, D., Lv, F., Guo, J., Zhang, H., & Zhu, L. (2023). Detection of Forestry Pests Based on Improved YOLOv5 and Transfer Learning. Forests, 14(7), 1484. https://doi.org/10.3390/f14071484

[19] Liu, J., & Wang, X. (2024). A multimodal framework for pepper diseases and pests detection. Scientific Reports, 14(1), 20. https://doi.org/10.1038/s41598-024-80675-w

[20] Liu, L., Wang, R., Xie, C., Yang, P., Wang, F., Sudirman, S., & Liu, W. (2019). PestNet: An End-to-End Deep Learning Approach for Large-Scale Multi-Class Pest Detection and Classification. IEEE Access, 7, 45301–45312. https://doi.org/10.1109/ACCESS.2019.2909522

[21] Liu, X., Ma, W., Ma, X., & Wang, J. (2023). LAE-Net: A locally-adaptive embedding network for low-light image enhancement. Pattern Recognition, 133, 109039. https://doi.org/10.1016/j.patcog.2022.109039

[22] Lu, P., Jia, Y. S., Zeng, W. X., & Wei, P. (2024). CDF-YOLOv8: City Recognition System Based on Improved YOLOv8. IEEE Access, 12, 143745–143753. https://doi.org/10.1109/ACCESS.2024.3471690

[23] Maharlooei, M., Sivarajan, S., Bajwa, S. G., Harmon, J. P., & Nowatzki, J. (2017). Detection of soybean aphids in a greenhouse using an image processing technique. Computers and Electronics in Agriculture, 132, 63–70. https://doi.org/10.1016/j.compag.2016.11.019

[24] Ouardirhi, Z., Mahmoudi, S. A., & Zbakh, M. (2024). Enhancing Object Detection in Smart Video Surveillance: A Survey of Occlusion-Handling Approaches. Electronics, 13(3), 541. https://doi.org/10.3390/electronics13030541

[25] Park, D., Lee, B. H., & Chun, S. Y. (2023). All-in-One Image Restoration for Unknown

Degradations Using Adaptive Discriminative Filters for Specific Degradations. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5815–5824. https://doi.org/10.1109/CVPR52729.2023.00563

[26] Peng, L., Zhu, C., & Bian, L. (2023). U-Shape Transformer for Underwater Image Enhancement. IEEE Transactions on Image Processing, 32, 3066–3079. https://doi.org/10.1109/TIP.2023.3276332

[27] Qian, S., Du, J., Zhou, J., Xie, C., Jiao, L., & Li, R. (2023). An effective pest detection method with automatic data augmentation strategy in the agricultural field. Signal, Image and Video Processing, 17(2), 563–571. https://doi.org/10.1007/s11760-022-02261-9

[28] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection (arXiv:1506.02640). arXiv. https://doi.org/10.48550/arXiv.1506.02640

[29] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

[30] Schowalter, T. D., & Ring, D. R. (2017). Biology and Management of the Fall Webworm , Hyphantria cunea (Lepidoptera: Erebidae). Journal of Integrated Pest Management, 8(1), 6. https://doi.org/10.1093/jipm/pmw019

[31] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. 2017 IEEE International Conference on Computer Vision (ICCV), 618–626. https://doi.org/10.1109/ICCV.2017.74

[32] Shi, Y., Wang, H., Wang, F., Wang, Y., Liu, J., Zhao, L., Wang, H., Zhang, F., Cheng, Q., & Qing, S. (2025). Efficient and accurate tobacco leaf maturity detection: An improved YOLOv10 model with DCNv3 and efficient local attention integration. Frontiers in Plant Science, 15, 1474207. https://doi.org/10.3389/fpls.2024.1474207

[33] Sun, B., Zhang, W., Xing, C., & Li, Y. (2025). Underwater moving target detection and tracking based on enhanced you only look once and deep simple online and realtime tracking strategy. Engineering Applications of Artificial Intelligence, 143, 109982. https://doi.org/10.1016/j.engappai.2024.109982

[34] Sun, D., Zhang, K., Zhong, H., Xie, J., Xue, X., Yan, M., Wu, W., & Li, J. (2024). Efficient Tobacco Pest Detection in Complex Environments Using an Enhanced YOLOv8 Model. Agriculture, 14(3), 353. https://doi.org/10.3390/agriculture14030353

[35] Tian, C., Zheng, M., Zuo, W., Zhang, B., Zhang, Y., & Zhang, D. (2023). Multi-stage image denoising with the wavelet transform. Pattern Recognition, 134, 109050. https://doi.org/10.1016/j.patcog.2022.109050

[36] Wang, C.-Y., Yeh, I.-H., & Mark Liao, H.-Y. (2025). YOLOv9: Learning What You

Want to Learn Using Programmable Gradient Information. In A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, & G. Varol (Eds), Computer Vision – ECCV 2024 (Vol. 15089, pp. 1–21). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-72751-1_1

[37] Wang, F., Huang, Y., Huang, Z., Shen, H., Huang, C., Qiao, X., & Qian, W. (2023). MRUNet: A two-stage segmentation model for small insect targets in complex environments. Journal of Integrative Agriculture, 22(4), 1117–1130. https://doi.org/10.1016/j.jia.2022.09.004

[38] Wang, K., Gou, C., Duan, Y., Lin, Y., Zheng, X., & Wang, F.-Y. (2017). Generative adversarial networks: Introduction and outlook. IEEE/CAA Journal of Automatica Sinica, 4(4), 588–598. https://doi.org/10.1109/JAS.2017.7510583

[39] Wang, S., Xu, D., Liang, H., Bai, Y., Li, X., Zhou, J., Su, C., & Wei, W. (2025). Advances in Deep Learning Applications for Plant Disease and Pest Detection: A Review. Remote Sensing, 17(4), 698. https://doi.org/10.3390/rs17040698

[40] Wang, X., Liu, J., & Zhu, X. (2021). Early real-time detection algorithm of tomato diseases and pests in the natural environment. Plant Methods, 17(1), 43. https://doi.org/10.1186/s13007-021-00745-2

[41] Wei, H., Zhao, L., Li, R., & Zhang, M. (2025). RFAConv-CBM-ViT: enhanced vision transformer for metal surface defect detection. The Journal of Supercomputing, 81(1), 155. https://doi.org/10.1007/s11227-024-06662-0

[42] Yang, M., Han, X., Ping, X., Li, Z., & Xiao, J. (2023). A Clearer Image: Improving Object Detection in Real Rainy Conditions with Two-Stage Processing. 2023 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), 57–62. https://doi.org/10.1109/ICMEW59549.2023.00016

[43] Yang, Z., Fang, S., & Huang, H. (2024). Maize leaf disease image enhancement algorithm using TFEGAN. Crop Protection, 182, 106734. https://doi.org/10.1016/j.cropro.2024.106734

[44] Yao, Y., Jiang, X., Fujita, H., & Fang, Z. (2022). A sparse graph wavelet convolution neural network for video-based person re-identification. 10.1016/j.Patcog.2022.108708, 129, 108708. https://doi.org/10.1016/j.patcog.2022.108708

[45] Ye, W., Lao, J., Liu, Y., Chang, C.-C., Zhang, Z., Li, H., & Zhou, H. (2022). Pine pest detection using remote sensing satellite images combined with a multi-scale attention-UNet model. Ecological Informatics, 72, 101906. https://doi.org/10.1016/j.ecoinf.2022.101906

[46] Zhang, H., Chang, H., Ma, B., Wang, N., & Chen, X. (2020). Dynamic R-CNN: Towards High Quality Object Detection via Dynamic Training. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds), Computer Vision – ECCV 2020 (Vol. 12360, pp. 260–275). Springer International Publishing. https://doi.org/10.1007/978-3-030-58555-6_16

[47] Zhang, L., Chen, K., Zheng, L., Liao, X., Lu, F., Li, Y., Cui, Y., Wu, Y., Song, Y., & Yan, S. (2024). Enhancing Fruit Fly Detection in Complex Backgrounds Using Transformer Architecture with Step Attention Mechanism. Agriculture, 14(3), 490. https://doi.org/10.3390/agriculture14030490

[48] Zhang, Y., Ma, B., Hu, Y., Li, C., & Li, Y. (2022). Accurate cotton diseases and pests detection in complex background based on an improved YOLOX model. Computers and Electronics in Agriculture, 203, 107484. https://doi.org/10.1016/j.compag.2022.107484

[49] Zhang, Y., Wu, Y., Liu, Y., & Peng, X. (2025). CPA-Enhancer: Chain-of-Thought Prompted Adaptive Enhancer for Downstream Vision Tasks Under Unknown Degradations. ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1–5. https://doi.org/10.1109/ICASSP49660.2025.10888304

[50] Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., & Chen, J. (2024). DETRs Beat YOLOs on Real-time Object Detection. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 16965–16974. https://doi.org/10.1109/CVPR52733.2024.01605

[51] Zhou, J., Pang, L., Zhang, D., & Zhang, W. (2023). Underwater Image Enhancement Method via Multi-Interval Subhistogram Perspective Equalization. IEEE Journal of Oceanic Engineering, 48(2), 474–488. https://doi.org/10.1109/JOE.2022.3223733