Article

Not peer-reviewed version

Detection of Mulberry Leaf Diseases in Natural Environments Based on Improved YOLOv8

Ming Zhang , Chang Yuan , Qinghua Liu * , Hongrui Liu , Xiulin Qiu , Mengdi Zhao *

Posted Date: 14 June 2024

doi: 10.20944/preprints202406.1021.v1

Keywords: mulberry leaf disease; YOLOv8; object detection; attention mechanism; NWD loss function

Article

Detection of Mulberry Leaf Diseases in Natural Environments Based on Improved YOLOv8

Ming Zhang [1], Chang Yuan [2], Qinghua Liu [1,*], Hongrui Liu [2], Xiulin Qiu [1] and Mengdi Zhao [3,*]

[1] Colleage of Automation, Jiangsu University of Science and Technology, Zhenjiang212003, China
[2] Colleage of Computer, Jiangsu University of Science and Technology, Zhenjiang212003, China
[3] Department of Materials Science and Engineering, Suzhou University of Science and Technology, Suzhou 215011, China
* Correspondence: liuqh@just.edu.cn (Q.L.); mdzhao@usts.edu.cn (M.Z.); Tel.: +86-139-1455-7059 (Q.L.)

Abstract: Mulberry leaves, when infected by pathogens, can suffer significant yield loss or even death if early disease detection and timely spraying are not performed. To enhance the detection performance of mulberry leaf diseases in natural environments and to precisely locate early small lesions, we propose a high-precision, high-efficiency disease detection algorithm named YOLOv8-RFMD. Based on improvements to You Only Look Once version 8 (YOLOv8), we first proposed the Multi Dimension Feature Attention (MDFA) module, which integrates important features at the pixel-level, spatial and channel dimensions. Building on this, we designed the RFMD Module, which consists of the Conv-BatchNomalization-SiLU (CBS) module, Receptive-Field Coordinated Attention (RFCA) Conv, and MDFA, replacing the Bottleneck in the model's Residual block. We then employed Adown down sampling structure to reduce the model size and computational complexity. Finally, to improve the detection precision of small lesion features, we replaced the complete intersection over union (CIOU) loss function with the Normalized Wasserstein Distance (NWD) loss function. Results show that the YOLOv8-RFMD model achieves an mAP50 of 94.3% and an mAP50:95 of 67.8% on experimental data, representing increases of 2.9% and 4.3% respectively compared to the original model. The model size is reduced by 0.53 MB to just 5.45 MB, and the number of floating-point operations is reduced by 0.3 G to only 7.8 G. The improved model meets the deployment requirements of mobile embedded devices and can provide a theoretical reference for the automated spraying operations of mulberry leaves.

Keywords: mulberry leaf disease; YOLOv8; object detection; attention mechanism; NWD loss function

1. Introduction

The mulberry tree, a member of the Moraceae family and the genus Morus, is one of the world's most important economic crops. It is widely distributed across all continents and is considered one of the most suitable plants for sustainable development [1]. Cultivating mulberry trees not only improves environmental quality but also provides significant economic value across various industries, including sericulture, pharmaceuticals, food, and cosmetics. In China, sericulture has a history spanning thousands of years. Mulberry leaves serve as the primary food for silkworms and are indispensable to the entire sericulture industry, as their quality and quantity directly affect the cocooning rate of silkworms [2]. During the growth of mulberry leaves, they are inevitably susceptible to various pathogens, which can cause diseases. If these diseases are not detected and controlled early, the infected leaves can quickly spread the pathogens to healthy leaves, leading to a significant reduction in yield or even death of the mulberry plants, causing substantial economic losses to the sericulture industry [3]. Currently, the detection and treatment of mulberry leaf diseases are mainly performed manually. This manual process is time-consuming and labor-intensive, and subjective judgment can lead to incorrect pesticide application. Therefore, promoting intelligent and automated pesticide application is essential to enhance the management efficiency of mulberry plantations. The primary task for achieving intelligent pesticide application is to efficiently and precisely detect and locate diseased mulberry leaves, which is crucial for ensuring the yield and quality of mulberry leaves.

With the continuous advancement of computer technology, an increasing number of scholars are utilizing deep learning techniques to identify and detect crop diseases, achieving rapid progress [4]. Although there is limited research on mulberry leaf diseases in the literature, there have been significant achievements in other crop fields. Javidan et al. [5] used a novel image processing algorithm and multi-class support vector machine (SVM) to diagnose and classify grape leaf diseases (black measles, black rot, and leaf blight). The K-means clustering was used to automatically separate disease symptom areas from healthy parts of the leaves, achieving an precision of 98.97%. Sladojevic et al. [6] developed a new method for crop image-based pest and disease recognition using the AlexNet [7] convolutional network, achieving an precision of 91% to 98% in single-category tests and an overall model precision of 96.3%. RANGARAJAN et al. [8] trained tomato disease images using improved AlexNet and Visual Geometry Group (VGG16) [9] models, but the model recognition precision was not satisfactory. Nahiduzzaman et al. [10] proposed an explainable AI (XAI) framework and developed a unique lightweight parallel depthwise separable Convolutional neural network(CNN) model, PDS-CNN, to classify mulberry leaf diseases using a newly established mulberry leaf dataset. The results showed that the XAI-based PDS-CNN model had higher classification precision, fewer parameters, fewer layers, and a smaller overall size compared to other transfer learning models. Waheed et al. [11] proposed an optimized DenseNet model to better identify and classify maize leaf diseases that are indistinguishable at the seedling stage to monitor crop health. The optimized model achieved an precision of 98.06% and required significantly fewer parameters and computation time than existing CNNs.

In the actual cultivation environment of mulberry leaves, images for automated detection of mulberry leaf diseases often contain different types of diseases, and partial occlusion between leaves can occur. Based on this, compared to image classification algorithms, object detection algorithms are more suitable for detecting multiple targets in images and determining the size and location of actual disease features. Wen et al. [12] proposed an algorithm that combines a multi-scale residual network with Squeeze-and-excitation Net(SENet) for identifying mulberry leaf diseases. To broaden the network's capacity to capture more information, multi-scale convolution was used instead of traditional single-scale convolution, and SENet was introduced to enhance the extraction of key features. The results showed a recognition precision of 98.72%, with a recall rate and F1 score of 98.73% and 98.72%, respectively. Xue et al. [13] proposed an improved You Only Look Once (YOLO) v5-based model for detecting tea leaf diseases and pests, named YOLO-Tea, for detecting tea leaf diseases and pests in natural environments. Compared to models like YOLOv5s, (Faster region-based convolutional neural networks)Faster R-CNN [14], and Single Shot Multibox Detector (SSD) [15], YOLO-Tea improved by 0.3% to 15.0% on all test data. Li et al. [16] enhanced the effective information of feature maps and reduced the loss of feature information by adding the Coordinated Attention (CA) attention module [17] and the spatial pyramid pooling (SSP) model to YOLOv5s to improve the detection precision of maize leaf diseases. Nie et al. [18] proposed a disease detection network based on Faster R-CNN and multi-task learning for precisely detecting strawberry wilt disease. The strawberry wilt disease detection network (SVWDN) can automatically classify petioles and young leaves while determining whether strawberries are infected with downy mildew, achieving an precision of 99.95% in strawberry wilt disease detection. Dwived et al. [19] proposed a grape leaf disease detection network (GLDDN) that utilizes dual attention modules for feature evaluation, detection, and classification. Experiments on a benchmark dataset confirmed that GLDDN is more suitable than existing methods.

Although the aforementioned research has significantly promoted the development of crop disease recognition and detection, some problems remain unresolved. First, under natural conditions, the complexity of detection conditions can result in many models having poor detection precision and low localization precision, leading to missed detections and false positives. Second, there is currently limited research on the detection of minute features of diseases, which are usually not apparent in the early stages of leaf disease. Timely detection of these disease features is crucial for disease prevention. Finally, researchers often use model fusion methods, which, although they ensure a certain level of detection precision, increase model size and computational load, making deployment on mobile devices difficult. Therefore, after comparing numerous studies and deep learning algorithms, this study employs an improved YOLOv8 algorithm to detect mulberry leaf diseases in natural environments. Compared to previous versions of YOLO [20], the YOLOv8 model is a more stable detection model, utilizing advanced training methods that result in shorter training times and faster convergence. While improving inference speed, it still maintains high detection precision.

The main contributions of this paper are as follows: (1) We proposed the Multi Dimension Feature Attention (MDFA) module, which integrates important features at the pixel-level, spatial and channel dimensions. This multi-dimensional consideration of feature information maximizes the retention of important features. (2) Building on the MDFA module, we designed the RFMD Module , consisting of the Conv-BatchNomalization-SiLU(CBS) module, Receptive-Field Coordinated Attention (RFCA Conv), and MDFA, to replace the Bottleneck in the Residual block of the original model. The RFCA Conv within the RFMD Module not only focuses on important local information at the receptive field level but also precisely captures the location information of diseases, addressing the parameter sharing issue inherent in traditional convolutions. (3) We used the ADown sampling structure to replace the CBS modules in the backbone network at P3, P4, and P5, as well as in the neck network. This down sampling structure combines various down sampling and feature processing methods to prevent the loss of effective information when the spatial resolution of the image is reduced. It effectively reduces the model size and computational load while maintaining precision. (4)we replaced the complete intersection over union (CIOU) loss function with the Normalized Wasserstein Distance (NWD) loss function, significantly enhancing the detection of small disease features. (5) We evaluated the performance and effectiveness of the YOLOv8-RFMD model by comparing it with other network models, including YOLOv8n, YOLOv7-Tiny [21], YOLOv5s, Faster R-CNN and so on. (6) We added the MDFA module and other attention mechanisms to the model separately and conducted comparative analysis using heatmaps and experimental data.

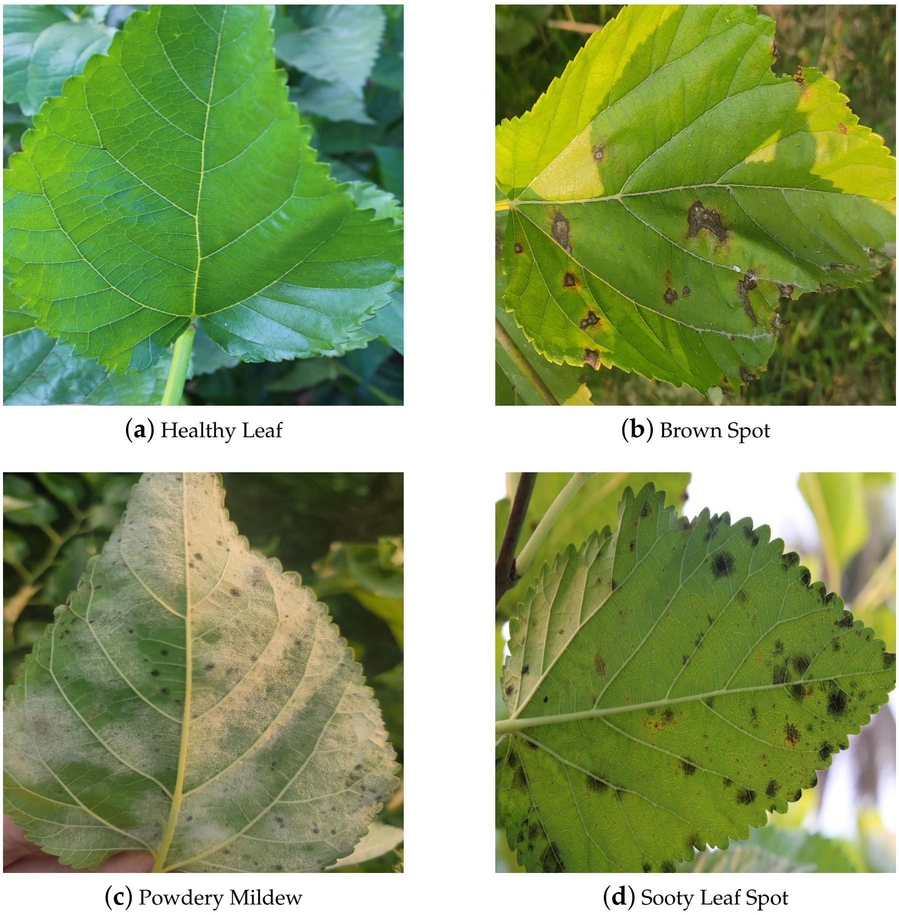
2. Materials and Methods

2.1. Dataset Construction

Mulberry leaf diseases are widespread globally. This study focuses on three common and distinct diseases: mulberry powdery mildew, brown spot, and sooty leaf spot, An example of the mulberry leaf disease images is shown in the Figure 1. In the early stages of powdery mildew infection in mulberry, small circular powdery mildew spots appear on the underside of the leaves.The symptoms of mulberry brown spot disease manifest as brown spots of varying shapes and sizes on both sides of the leaves.The pathogen of sooty leaf spot disease is Sirosporium mori (H. & P. Syb.) M. B. Ellis, which initially manifests as small coal dust-like black spots. In this study, it is referred to as sooty leaf spot.

Through the Kaggle official website, a high-quality dataset of mulberry leaf diseases with 871 images has been obtained. This dataset includes images of both diseased and healthy leaves. The images were captured using a camera, depicting mulberry leaves affected by powdery mildew, brown spot disease, dirty leaf disease, as well as healthy leaves. They were taken under various environmental conditions such as sunny days, cloudy days, front light, back light, upward angle, downward angle, leaf surface, and leaf back, aiming to increase the diversity of images as much as possible. To ensure relative uniformity across categories, unsuitable images were removed, resulting in a dataset of 600 images. Approximately half of these images contain two or more diseases.

To prevent overfitting in the neural network and enhance the robustness of the samples as well as the generalization ability of the network, the original images were augmented using methods such as flipping, adding noise, and adjusting brightness. A total of 3000 images were generated through these augmentations. The augmented images were then manually annotated using the LabelImg tool to obtain the category and location information of the target diseases in the images. The annotated information was saved in txt files, completing the construction of the mulberry leaf disease dataset. The dataset was randomly divided into training, validation, and test sets in a ratio of 7:2:1, with 2100 images in the training set, 600 images in the validation set, and 300 images in the test set.



(**a**) Healthy Leaf

(**b**) Brown Spot

(**c**) Powdery Mildew

(**d**) Sooty Leaf Spot

Figure 1. Mulberry Leaf Diseases.

2.2. *YOLOv8 Algorithm*

The YOLOv8 algorithm is a popular detection algorithm that is divided into four parts: input, backbone network, neck network, and head network. Depending on the application scenario, it has different models including n, s, l, m, and x. To ensure real-time performance and a manageable model size, this study uses the YOLOv8n version, which has the smallest number of parameters and the fastest detection speed. Compared to previous generations of YOLO algorithms, the YOLOv8 algorithm's backbone network adopts the Cross Stage Partial(CSP) Darknet53 [22] architecture, incorporating CBS, Faster Implementation of CSP Bottleneck with 2 convolutions (C2f), and SPPF (spatial pyramid pooling fusion) structures. The C2f structure is the primary module for learning residual features, combining the CSP Bottleneck with 3 convolutions (C3) structure from YOLOv5 and the Efficient Layer Aggregation Network (ELAN) structure from YOLOv7, providing richer gradient flow information. The neck network uses the path aggregation network [23] (PAN) and the feature pyramid network [24] (FPN) structures to achieve the fusion and enhancement of features of different sizes, providing richer information for the head network to detect. The head network uses a decoupled head structure, separating the classification and detection processes. The detection head uses Bbox Loss, combining distribution focal loss (DF Loss) and CIOU loss [25] to measure loss, while the classification head uses binary cross-entropy (BCE) loss, comprehensively improving the model's precision in predicting bounding boxes. Although the original YOLOv8 algorithm can achieve good detection precision, the presence of small objects, varied scales, and similar features in images can affect detection precision. Additionally, the original model's computational complexity and model size need to be reduced, indicating that YOLOv8 still requires improvements.

2.3. Improved YOLOv8 Algorithm

This study aims to improve the original YOLOv8 algorithm. The improved network structure is shown in Figure 2, with specific improvements detailed as follows:

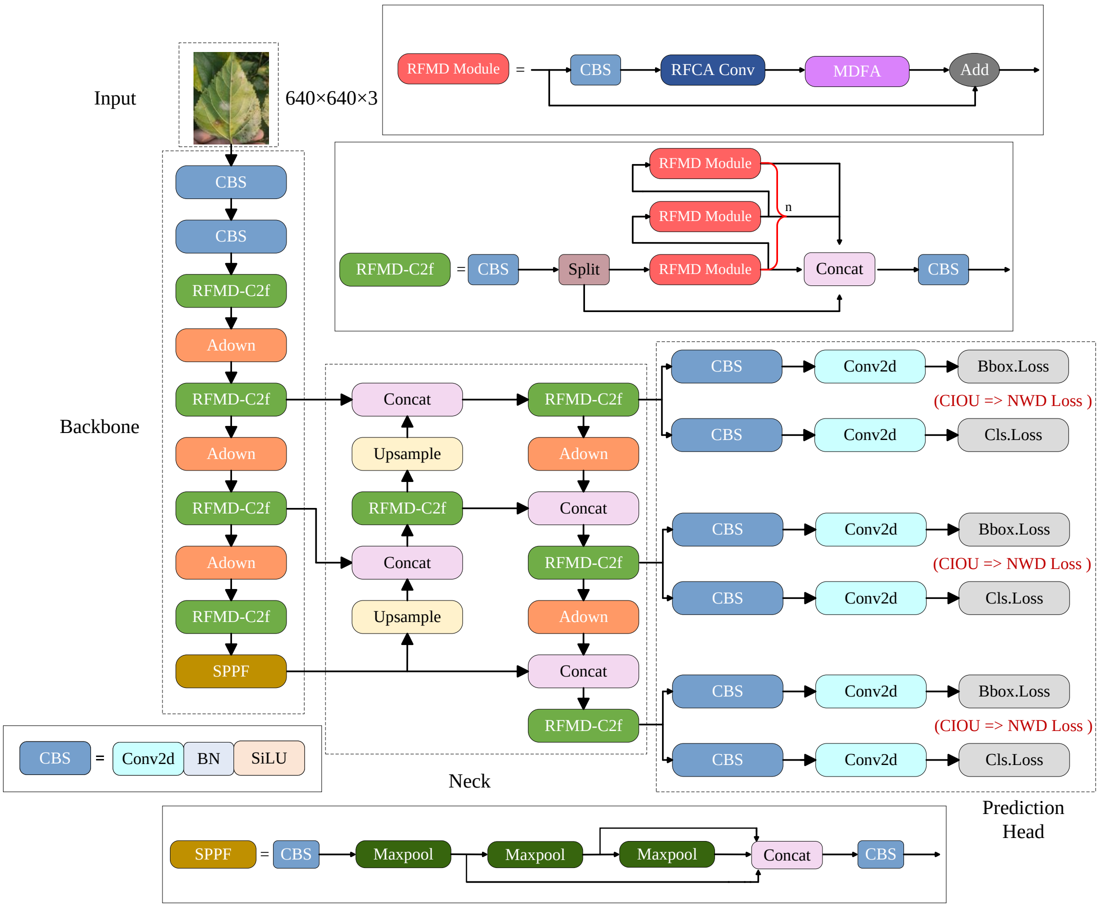


Figure 2. Schematic Diagram of the Improved YOLOv8 Model.

(1) The Bottleneck in the C2f module is replaced with the RFMD Module, which consists of the CBS module, RFCA Conv, and MDFA. The RFMD Module uses the MDFA module proposed in this paper, which focuses on features from pixel-level dimension, spatial dimension and channel dimension. This enhances the extraction of effective feature information from channels while integrating both global and local spatial information. Additionally, the RFCA Conv not only focuses on important local information at each receptive field level but also enables the model to more precisely locate defect positions during detection, addressing the parameter sharing issue inherent in traditional convolutions.

(2) The CBS modules in P3, P4, and P5 of the backbone network, as well as the CBS modules in the neck network, are replaced with the Adown down sampling structure. This structure uses various down sampling methods to extract features, preventing the loss of important features while reducing the model size and computational complexity.

(3) The original YOLOv8's loss function, CIOU Loss, is replaced with NWD Loss. This new loss function improves the detection precision of small targets.

2.3.1. MDFA Attention

In the detection of mulberry leaf diseases, using the original YOLOv8 algorithm often results in missed and false detections, particularly in the early stages of diseases such as powdery mildew and sooty leaf spot. This is because the algorithm fails to filter out important feature information during feature extraction. To more precisely identify disease spots, this study proposes the MDFA module. Common channel attention modules (e.g., SE [26], Efficient Channel Attention(ECA) [27]) only consider relationships between channels and ignore spatial dimension information. If a channel has a low weight but its spatial information is significant, this important feature information will be lost. Additionally, common spatial attention modules (e.g., Convolutional Block Attention Module (CBAM) [28]) perform redundant operations in spatial dimension attention, as the weight distribution of specific local areas is generally uniform. The proposed MDFA simultaneously considers information from pixel-level dimension, channel dimension, and spatial dimension . Initially, the input features are preprocessed through an energy function [29], which assigns a 3-D weight to each feature point to evaluate its importance and highlight key feature points. The preprocessed feature map is then divided into multiple patches to further incorporate channel and spatial dimension information. One-dimensional convolution is used to reduce computational and parameter complexity. This approach allows the model to capture channel information, spatial information, local information, and global information simultaneously, achieving multi-dimensional feature attention. The specific structure is illustrated in Figure 3.

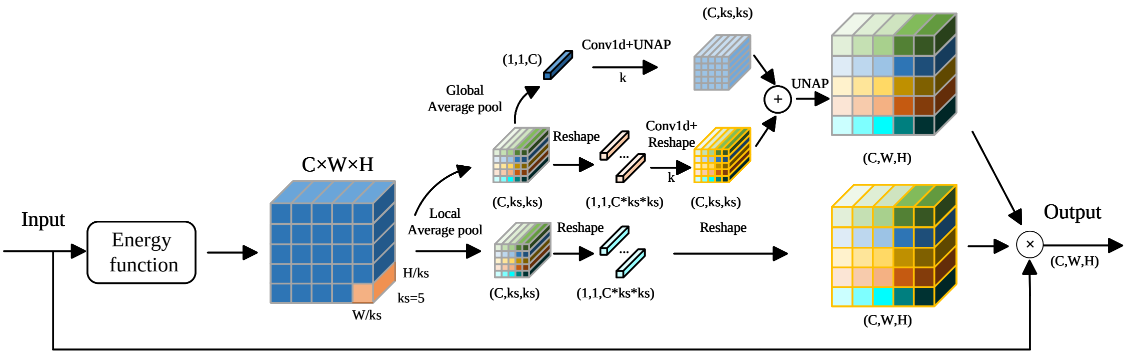


Figure 3. Structure of the MDFA Attention module.

The input feature vector for MDFA is first preprocessed through an energy function, the energy function is shown in Equation (1):

$$e_t^* = \frac{4(\widehat{\sigma}^2 + \lambda)}{(t - \widehat{\mu})^2 + 2\widehat{\sigma}^2 + 2\lambda} \tag{1}$$

Here, $\widehat{\mu} = \frac{1}{M}\sum_{i=1}^{M} x_i$, $\widehat{\sigma}^2 = \frac{1}{M}\sum_{i=1}^{M}(x_i - \widehat{\mu})^2$, t and x_i are the target feature points and other feature points within a single channel of the input features. i is the index in the spatial dimension, and $M = H \times W$ is the number of feature points in that channel. The lower the energy e_t^* , the greater the difference between feature point t and other surrounding feature points, indicating higher importance. Therefore, the importance of each feature point can be obtained by $1/\ e_t^*$.

After the energy function preprocessing, the feature map enters the pooling section, which consists of two steps: local average pooling and global average pooling. The input is converted into a vector $1 \times C \times \text{ks} \times \text{ks}$ to extract local spatial information through local pooling. Based on the initial stage, the input is transformed into a one-dimensional vector using three branches. The first branch contains global information, and the second branch contains local spatial information. After one-dimensional convolution, these are restored to the size of $C \times$ ks $\times$ ks through unpooling and reshaping, and the information from the two branches is added and unpooled back to the original resolution. The third branch, after one-dimensional convolution, is reshaped back to the size of $C \times$ H $\times$ W. Finally, the information from the three branches is fused, achieving the goal of multi-dimensional feature attention. In the diagram, Conv1d represents one-dimensional convolution, where the kernel size k is proportional to the number of channels C. This implies that in capturing local cross-channel interaction information, only the relationship between each channel and its k adjacent channels is considered. The formula for selecting k is as follows:

$$k = \varphi(C) = \left| \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right|_{odd} \tag{2}$$

Here, C represents the number of channels, and k is the kernel size. Both γ and b are hyperparameters with default values of 2. k is chosen to be an odd number; if k is even, 1 is added to make it odd.

2.3.2. RFCA Conv in the RFMD Module

In natural settings, mulberry leaves often grow densely, and the pathological features usually vary in size. The original YOLOv8 uses standard convolution operations, which typically apply fixed weights to input data at all locations. While this method simplifies the model's parameters, it overlooks the uniqueness of local areas in the image and fails to precisely locate disease features. Therefore, using standard convolution has certain limitations. To address this issue, this study employs RFCA Conv [30] in the RFMD Module of the RFMD-C2f module. RFCA Conv combines Receptive-Field Attention (RFA) and CA module. Receptive-Field Attention calculates attention at the receptive field level for each convolution operation, adjusting the weights of feature processing for each local area and weighting the features within the receptive field based on the calculation results. This highlights important features and overcomes the performance limitations caused by parameter sharing in traditional convolution, making it more effective for complex or fine-grained visual tasks. The CA module simultaneously calculates attention in both channel and spatial dimensions, better capturing dependencies between features. By coordinating spatial and channel attention, it comprehensively integrates important feature information and enhances the feature representation ability of mobile networks. The structure is shown in Figure 4.

RFCA Conv captures local spatial information within each channel and computes attention at the receptive field level. It uses grouped convolution to process each input channel individually, expanding each channel spatially in preparation for receptive field expansion, with the output feature map size denoted as $CK^2 \times H \times W$ and K^2 representing the expansion size. Batch normalization and ReLU activation are then applied to enhance the model's nonlinearity. The feature map dimensions are adjusted to allow independent processing of features within each receptive field, resulting in an adjusted size of $C \times KH \times KW$.RFCA Conv captures attention in both height and width dimensions of the image and encodes precise positional information. The adjusted feature map undergoes global average pooling along both height and width for each channel. After obtaining the feature maps in horizontal and vertical directions, they are concatenated and processed with a 1×1 convolution operation. The merged features are normalized and activated using a nonlinear activation function, resulting in an intermediate feature map f of size $C \times (KH + KW) \times 1$, which integrates global information from both directions.Next, the feature map f is split again along vertical and horizontal directions and processed using 1×1 convolution, followed by Sigmoid activation to obtain attention maps in height and width directions. These attention maps are applied to the original expanded feature map for reweighting via element-wise multiplication, highlighting important features at different spatial locations for each channel. Finally, the reweighted feature map is processed through a convolution layer with a stride of K, producing a final output of size $C \times H \times W$. The stride equals the receptive field size K, meaning each convolution kernel's output is only related to its receptive field, thereby reducing parameter sharing issues.

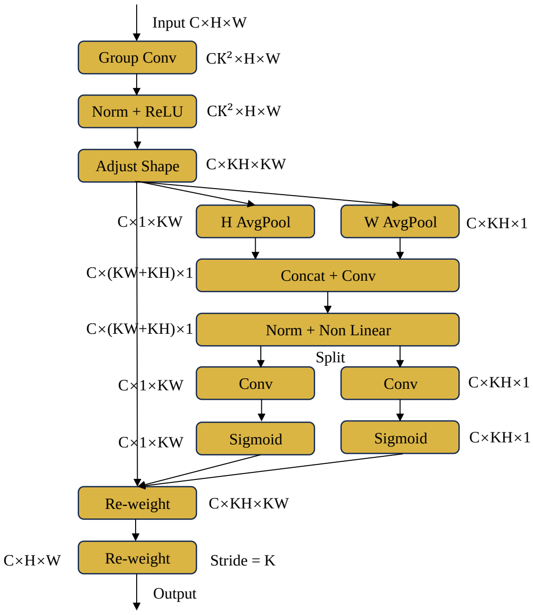


Figure 4. Schematic Diagram of RFCA Conv.

2.3.3. Lightweight Down Sampling Structure ADown

In CNN-based object detection methods, down sampling images is a key operation to reduce data dimensions and complexity while preserving important feature information. The original YOLOv8 employs 3×3 convolution kernels with a stride of 2, Batch Normalization, and SiLU activation functions for down sampling. However, in real-world applications, devices often have limited computational power, which can slow down the model's performance. This study replaces the original down sampling method with the lightweight down sampling structure ADown [31].ADown combines various down sampling techniques, including average pooling, max pooling, and convolution operations, to extract and retain critical information from different perspectives. This approach avoids information loss that may result from a single down sampling strategy. By splitting the input feature map along the channel dimension and processing each part separately, ADown reduces the spatial size of feature maps while extracting richer feature information through different paths. This structure helps improve the model's learning ability while reducing computational complexity. The specific structure is shown in Figure 5.

First, ADown applies average pooling to the input, using a pooling kernel size of 2×2 and a stride of 1. This operation reduces the feature map size while preserving background information. After average pooling, the output is split along the channel dimension into two parts, $x1$ and $x2$, each containing half of the channels. A convolution operation is applied to $x1$ with a 3×3 kernel size, a stride of 2, and a padding of 1, producing an output with half the final number of channels. This 3×3 convolution captures spatial information and achieves further down sampling. Meanwhile, $x2$ undergoes max pooling with a 3×3 kernel, a stride of 2, and a padding of 1, followed by a 1×1 convolution to adjust the number of feature channels, integrating and transforming features in a computationally efficient manner, resulting in an output with half the final number of channels. Finally, the Concat method is used to merge $x1$ and $x2$ along the channel dimension, forming the final output.ADown effectively combines multiple down sampling and feature processing strategies, reducing computational complexity while enhancing feature extraction capabilities and the model's generalization performance.

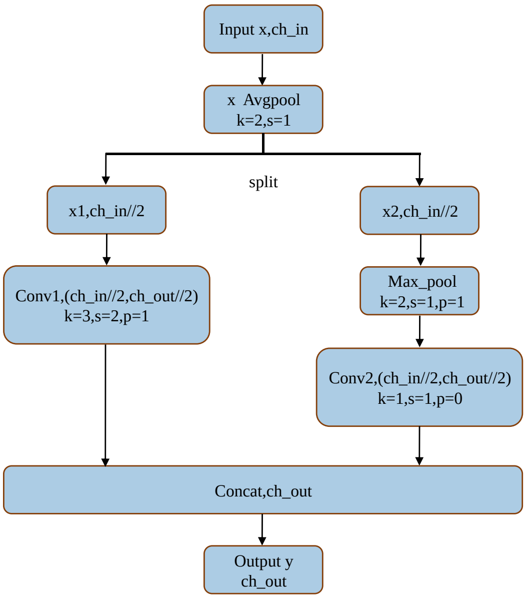


Figure 5. Structure of the ADown Module.

2.3.4. Normalized Wasserstein Distance Loss [32] function

YOLOv8 originally utilizes CIOU as its coordinate loss function. CIOU loss is an improved version of the IoU loss function, which not only considers the overlap between the predicted bounding box and the ground truth bounding box but also takes into account the consistency of the bounding box's center point distance and aspect ratio. In practical scenarios, some disease features on mulberry leaves are quite small, with some appearing in images containing only a few pixels, lacking sufficient visual information for effective detection. However, traditional evaluation metrics based on IoU and its extensions are highly sensitive to localization errors for small targets, which significantly degrade performance during detection. Therefore, employing the NWD loss function to replace the original loss function enhances the precision of detecting small targets. This novel evaluation metric first models bounding boxes as 2D Gaussian distributions, then introduces a new measure called Normalized Wasserstein Distance to compute their similarity. The advantage of this approach is that even when two bounding boxes have no overlap or very minimal overlap, Wasserstein Distance can measure the similarity of their distributions. NWD exhibits scale invariance for targets of different sizes and smoother handling of positional deviations, making it more suitable for detecting small targets compared to IoU and its extensions.

The importance of the distribution of foreground pixels (target objects) and background pixels within the bounding box varies, with their distribution weights gradually decreasing from the center of the bounding box to the edges. Therefore, leveraging the properties of Gaussian distributions to model the bounding box as a 2D Gaussian distribution enables better capture of this distribution characteristic. For bounding box $R = (cx, cy, w, h)$, its 2D Gaussian distribution's mean μ is located at the center (cx, cy) of the bounding box, representing the central position of the target object. The covariance matrix Σ represents the scale of the bounding box in the x and y directions, which can be determined by its width w and height h. Therefore, R can be modeled as a 2D Gaussian distribution $N(\mu, \Sigma)$, allowing the spatial distribution of the bounding box to be expressed in the form of a Gaussian distribution.

Wasserstein Distance is a measure of the difference between two probability distributions, particularly suitable when the two distributions have little to no overlap or are completely disjoint.For the 2D Gaussian distributions representing two bounding boxes, denoted as $N(\mu_A, \Sigma_A)$ and $N(\mu_B, \Sigma_B)$, the square of their Wasserstein Distance can be expressed as:

$$W_2^2(N_A, N_B) = \|\mu_A \ -\mu_B\|_2^2 + Tr(\Sigma_A + \Sigma_B - 2(\Sigma_A^{1/2}\Sigma_B\Sigma_A^{1/2})^{1/2}) \tag{3}$$

In Equation (3), the first term $\|\mu_A \ -\mu_B\|_2^2$ is the Euclidean distance between the centers of the two distributions, reflecting differences in position; the second term, computed by tracing the covariance matrix (Tr), considers differences in the shape (including size and orientation) of the two distributions.

In order to convert Wasserstein Distance into a similarity measure and make it suitable for replacing IoU in object detection, the formula for Normalized Wasserstein Distance (NWD) is:

$$NWD(N_A, N_B) = \exp(-\frac{\sqrt{W_2^2(N_A, N_B)}}{C}) \tag{4}$$

In Equation (4), C is a constant used to adjust the range of NWD values. Through this transformation, the value of NWD is normalized to between 0 and 1, where values closer to 1 indicate higher similarity between the two bounding boxes. This normalization process not only preserves the ability of Wasserstein Distance to measure differences between two distributions but also allows it to be directly used to evaluate the similarity between bounding boxes, especially in the context of detecting small objects. Finally, Normalized Wasserstein Distance is designed as a loss function:

$$L_{NWD} = 1 - NWD(N_p, N_g) \tag{5}$$

In Equation (5), N_p represents the Gaussian distribution model of the predicted bounding box, and N_g represents the Gaussian distribution model of the ground truth bounding box. NWD Loss addresses the issue where traditional IoU Loss fails to provide gradients for network optimization in two cases: when there is no overlap between the predicted bounding box P and the ground truth bounding box G ($|P \cap G| = 0$), or when box P completely contains box G or vice versa ($|P \cap G| = P$ or G).

2.4. Training Environment and Evaluation Metrics

2.4.1. Training Environment

The main parameters of the training platform used in this experiment are as follows: Intel Core i5-12600KF CPU with a clock speed of 3.7 GHz, 16GB of memory, a 1TB solid-state drive, Nvidia GeForce RTX 3060Ti with 8GB of memory, CUDA version 12.1, and Python version 3.8. The experiment was conducted on the Windows operating system, using the Pytorch deep learning framework for model building, training, and evaluation, with Pytorch version 2.1.2.

The training parameters are set as follows: the input image size is 640×640, the batch size is 8, multithreading is set to 4, the optimizer used is stochastic gradient descent, the number of training epochs is set to 400, the initial learning rate is 0.01, the weight decay rate is set to 0.0005, and the momentum is set to 0.937.

2.4.2. Evaluation Metrics

The experimental results are measured using the mean Average Precision (mAP, %) to evaluate the precision of model detection. Mean Average Precision is related to the model's precision (P, %) and recall (R, %), where precision P represents the proportion of samples correctly detected as mulberry leaf diseases out of the samples classified as mulberry leaf diseases by the classifier, as shown in the Equation (6):

$$P = \frac{T_P}{T_p + F_P} \times 100\% \tag{6}$$

The recall rate (R) represents the proportion of samples correctly detected as mulberry leaf diseases out of all actual mulberry leaf disease samples, as shown in the Equation (7):

$$R = \frac{T_P}{T_p + F_N} \times 100\% \tag{7}$$

The mean Average Precision (mAP) is the mean of the Average Precision (AP), where Average Precision (AP) represents the area under the precision-recall (P-R) curve for a specific mulberry leaf disease, as shown in the Equation (8):

$$\mathrm{m}AP = \frac{1}{N}\sum_{i=1}^{N}\int_{0}^{1} P(R)dR \times 100\% \tag{8}$$

In the equation, N represents the number of categories, T_P denotes the number of correctly detected mulberry leaf diseases, F_P represents the number of images incorrectly classified as a certain type of mulberry leaf disease, and F_N represents the number of actual mulberry leaf diseases in the images that were not correctly detected.

3. Results

3.1. Performance Comparison of Various Object Detection Models

The improved model YOLOv8-RFMD, based on YOLOv8, is compared with mainstream object detection models including YOLOv8n, YOLOv7-tiny, YOLOv5s, Faster R-CNN, SSD, and RetinaNet [33] in this study to demonstrate its effectiveness in object detection tasks, as shown in Table 1. The experiment adopts the same dataset and parameter settings for 400 iterations of training and testing. The table lists the precision, recall rate, mAP values at 50% loU threshold (mAP50), mAP values in the 50-95% loU threshold range (mAP50:95), model size, and floating-point operations for different models.

The comparative experiment results demonstrate that the detection precision, mAP, and mAP:95 of the YOLOv8-RFMD model are the highest compared to other networks. The mAP of the YOLOv8-RFMD model is 2.9%, 2.1%, 2.7%, and 3.9% higher than the other 6 models respectively, while the mAP50:95 is 4.3%, 11.7%, 6.0%, and 8.5% higher than the other 6 models respectively. The improved model in this paper is relatively small, with a size of only 5.45 MB, which is 0.53 MB smaller than the YOLOv8n model, making it the smallest model among all other models except YOLOv5s. The computational resources required for the YOLOv8-RFMD model are also lower, with Floating point of per second (FLOPs) of only 7.8 G, which is 0.3 G less than the already lower YOLOv8n model, making it suitable for deployment on embedded mobile devices. The computational efficiency of the YOLOv8-RFMD model far exceeds that of RetinaNet, SSD, and Faster R-CNN, with RetinaNet having 19 times more FLOPs, SSD 20 times more, and Faster R-CNN even 120 times more, which is far from the computing efficiency of the YOLOv8-RFMD model. In summary, the YOLOv8-RFMD model proposed in this study leads other 6 models in terms of comprehensive detection precision, model size, and computational efficiency. Considering the real-time detection requirements of mulberry leaf diseases, YOLOv7-tiny is inferior to the YOLOv8-RFMD model in terms of precision, model size, and floating-point operations. Although the model size and floating-point operations of YOLOv5s are slightly lower, its mAP50 and mAP50:95 are much higher than those of YOLOv5s. The single-stage detection network SSD has the lowest detection precision, the two-stage detection model Faster R-CNN has too many floating-point operations and the model size is too large, and although the RetinaNet network model has high model size and floating-point operations, the detection precision is still very low. SSD, Faster R-CNN, and RetinaNet cannot meet the real-time detection requirements of practical scenarios for disease detection.

Table 1. Different Model Training Results Comparison.

Model	Precision(%)	Recall(%)	mAP50(%)	mAP50:95(%)	Model size(MB)	FLOPs(G)
YOLOv8-RFMD	92.6	89.5	94.3	67.8	5.45	7.8
YOLOv8n	90.1	84.8	91.4	63.5	5.98	8.1
YOLOv7-tiny	90.8	88.1	92.2	56.1	11.7	13.2
YOLOv5s	90.1	85.1	91.6	61.8	5.04	7.1
Faster R-CNN	79.4	83.2	86.4	59.5	314	954
SSD	59.1	57.3	60.7	43.5	60.3	162
RetinaNet	64.2	61.7	64.5	46.3	338	150

In conclusion, the YOLOv8-RFMD model proposed in this research can guarantee relatively high precision for mulberry leaf disease detection while reducing the introduction of more parameters during inference, improving inference speed. The improved YOLOv8 model has a smaller scale and requires less computational resources, making it suitable for deployment on embedded devices to help detect diseases in mulberry orchards and take measures to prevent further spread of diseases.

3.2. Different Attention module Detection Performance Comparison

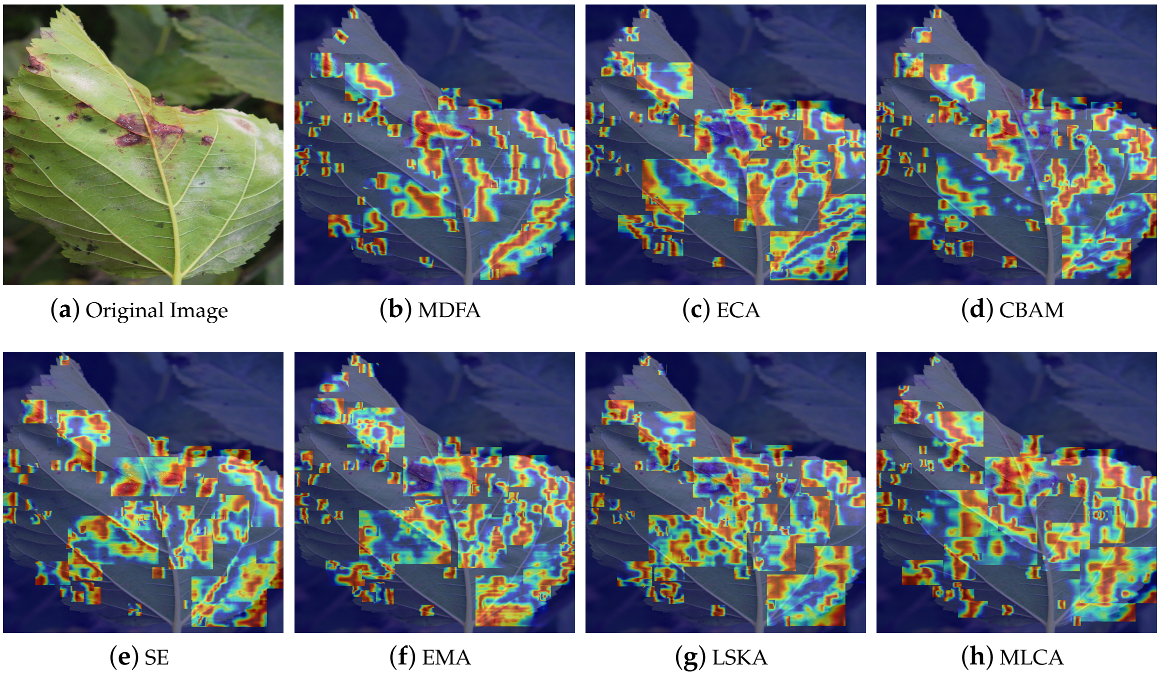
To validate the effectiveness of the MDFA attention module proposed in this paper in improving detection precision, another set of experiments was designed. Five types of attention modules, including Mixed Local Channel Attention (MLCA) [34], Efficient Multi-Scale Attention (EMA) [35], (LSKA) Large Separable Kernel Attention [36], SE, ECA, and CBAM, were respectively added or replaced with the MDFA at the same positions in this model, as shown in Table 2 for comparison.

From Table 2, it can be observed that the size and floating-point operations of the model remain largely unchanged after replacement. The main improvement is seen in the detection precision, where the MDFA attention module outperforms the other attention modules in terms of precision, recall, mAP50, and mAP50:95. Compared to the relatively new attention modules MLCA, EMA, and LSKA, the MDFA attention module improves precision by 1.0, 0.2, and 0.1 percentage points respectively, recall by 0.9, 1.7, and 0.7 percentage points respectively, mAP50 by 0.5, 0.7, and 0.4 percentage points respectively, and mAP50:95 by 1.2, 1.0, and 0.9 percentage points respectively. Compared to the classic attention modules SE, ECA, and CBAM, MDFA improves mAP50 by 0.5, 0.7, and 0.5 percentage points respectively, and mAP50:95 by 1.7, 1.1, and 0.8 percentage points respectively. Therefore, the MDFA attention module exhibits superior feature selection capability, surpassing the compared attention modules in identifying pathological features of mulberry leaves, thereby effectively enhancing the model's precision in disease recognition and detection.

Table 2. Training Results of Different Attention modules.

Attention	Precision(%)	Recall(%)	mAP50(%)	mAP50:95(%)	Model size(MB)	FLOPs(G)
MDFA	92.6	89.5	94.3	67.8	5.45	7.8
MLCA	91.6	88.6	93.8	66.6	5.46	7.8
EMA	92.4	87.8	93.6	66.8	5.49	7.8
LSKA	92.5	88.8	93.9	66.9	5.60	7.8
SE	92.3	88.3	93.8	66.1	5.47	7.8
ECA	92.4	88.0	93.6	66.7	5.45	7.8
CBAM	91.7	88.8	93.8	67.0	5.49	7.8

To further validate the superiority of the proposed MDFA attention module for mulberry leaf disease detection, seven types of attention modules were separately visualized using heat maps to demonstrate the focus of different attention modules on mulberry leaf disease features. Heat maps were generated through RandomCAM and images were re-normalized to obtain the heat maps, as shown in Figure 6. Heat maps are used to display the degree of attention of the neural network to each part of the image, presented in the form of weights. In the heat map, red areas indicate parts of higher attention, while blue areas indicate parts of lower attention.


(**a**) Original Image (**b**) MDFA (**c**) ECA (**d**) CBAM

(**e**) SE (**f**) EMA (**g**) LSKA (**h**) MLCA

Figure 6. Heatmaps of Different Attention modules.

From the figure, it can be observed that MDFA can more precisely locate the positions of different types of diseases and focus on disease areas that closely match the actual size of the diseases compared to other attention modules.

3.3. The results before and after improvement of the YOLOv8 model

3.3.1. The comparison of mAP50 and mAP50:95 before and after improvement

Using the YOLOv8-RFMD algorithm from this experiment and comparing it with the YOLOv8n trained for 400 iterations, the visualization of mAP is shown in Figure 7. From the graph, it can be observed that the mAP of YOLOv8-RFMD is superior to that of the original YOLOv8n. At 200 training iterations, both the mAP50 and mAP50:95 curves continue to rise but tend to plateau.

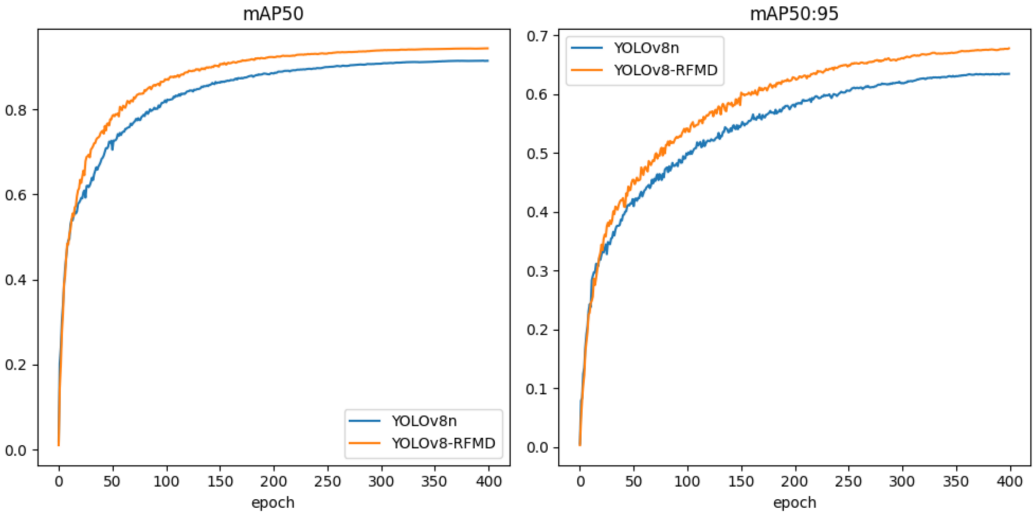
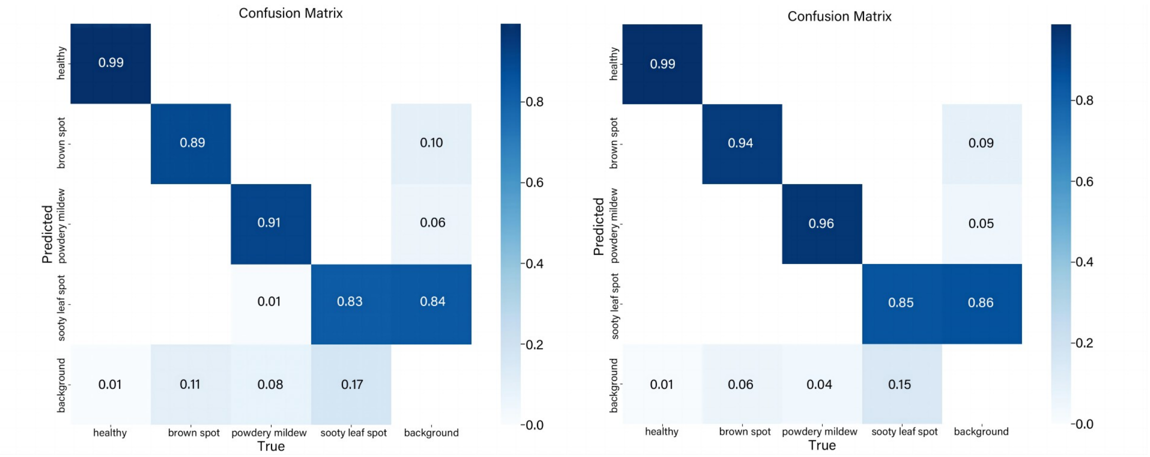


Figure 7. Visualization Comparison of mAP50 and mAP50:95 Before and After Improvement of YOLOv8.

3.3.2. Confusion Matrix

The confusion matrix is a two-dimensional matrix where the rows represent the predicted classes by the model, and the columns represent the actual labels' classes. From the confusion matrix in Figure 8, it can be observed that both before and after improvement, the model achieves the highest precision in detecting healthy mulberry leaves, reaching 99%. After improvement, the detection precision of brown spot disease and powdery mildew disease both exceeds 94%. However, the detection precision of sooty leaf spot disease is slightly lower due to variations in lesion size and the dense pathological features typically present. All images were captured in natural environments, resulting in complex backgrounds, which led to some diseases being missed. However, the improved model effectively reduces the probability of missed detections.



(**a**) YOLOv8n (**b**) YOLOv8-RFMD

Figure 8. Confusion Matrix.

3.4. Performance Comparison of Ablation Experiments

To validate the impact of each improvement on model precision and lightweighting and demonstrate the feasibility of the lightweight optimization strategy proposed in this study, ablation experiments were designed. Table 3 presents the results of the ablation experiments.

Table 3. Different Model Training Results Comparison.

Test	MDFA	RFCA Conv	ADown	NWD Loss	Precision (%)	Recall (%)	mAP50 (%)	mAP50:95 (%)	Model size (MB)	FLOPs (G)
1	-	-	-	-	90.1	84.8	91.4	63.5	5.98	8.1
2	✓	-	-	-	90.7	86.3	92.4	64.3	5.99	8.1
3	-	✓	-	-	90	86.5	92.4	64.9	6.22	8.5
4	✓	✓	-	-	90.5	86.2	92.6	65.4	6.23	8.5
5	-	-	✓	-	90.4	86.2	92.3	64.9	5.20	7.4
6	-	-	-	✓	91.4	86.2	92.4	63.9	5.98	8.1
7	✓	✓	✓	✓	92.6	89.5	94.3	67.8	5.45	7.8

The experiment is based on the YOLOv8n.'✓' denotes addition or improvement, while '-' indicates no change.

*

Experiment 1 represents the YOLOv8n model without any modifications. Experiment 2 adds the MDFA module proposed in this paper to the Bottleneck of the original C2f, focusing on important features at the pixel-level, spatial, and channel dimensions. This contributes to precisely focusing on disease areas, resulting in improvements in mAP50 and mAP50:95 with negligible changes in model size. Experiment 3 replaces the second convolution in the Bottleneck of the original C2f with RFCA Conv, which not only focuses on important local information at the receptive field level but also addresses the problem of parameter sharing in traditional convolutions, leading to more precision localization of disease positions. Despite the increase in floating-point operations and model size, there is a slight increase in detection precision. Experiment 4 combines MDFA and RFCA Conv to form the RFMD Module, further improving mAP50 and mAP50:95 without reducing floating-point operations or model size. In the presence of potential interference from complex environmental backgrounds in mulberry orchards, Experiment 4 demonstrates the performance of adding the RFMD Module to enhance disease detection precision. Experiment 5 introduces the ADown down sampling structure, replacing the CBS modules in the backbone network's P3, P4, and P5, and the CBS module in the neck network. This combines multiple down sampling methods to avoid loss of important feature information during down sampling, significantly reducing model size and floating-point operations while maintaining precision, achieving model lightweighting. Experiment 5 uses NWD Loss to replace the original YOLOv8 loss function, improving the model's ability to detect smaller disease features while maintaining model size and floating-point operations, thereby enhancing detection precision.

Experiment 7 incorporates all improvements into YOLOv8, achieving precision and recall rates of 92.6% and 89.5%, respectively, reaching the highest level among all experiments, indicating that the model achieves a high level of recognition and prediction precision for positive samples.The original YOLOv8n model itself has relatively small model size and floating-point operations. YOLOv8-RFMD further reduces model complexity, with reductions in model size and floating-point operations by 0.53 MB and 0.3 G, respectively, while increasing mAP50 from 91.4% to 94.3% and mAP50:95 from 63.5% to 67.8%. In summary, Experiment 7 achieves a comprehensive improvement in multiple metrics, significantly enhancing detection precision while achieving model lightweighting.

3.5. Different Models Detection Visualization Results Analysis

Based on Table 1, Faster R-CNN, SSD, and RetinaNet models not only have lower mAP but also have larger model sizes and floating-point operations compared to the YOLO series. They cannot meet the requirements for mulberry leaf disease detection and deployment on mobile embedded devices. Therefore, they are not further visualized for validation.

To test the actual effectiveness of the YOLOv8-RFMD model in detecting mulberry leaf diseases, the pre-trained YOLOv8-RFMD, YOLOv8n, YOLOv7-tiny, and YOLOv5s models are used to detect complex diseases on mulberry leaves. Specifically, they are tested for scenarios where brown spot disease and powdery mildew disease coexist, where sooty leaf spot disease and powdery mildew disease coexist, and where all three diseases (sooty leaf spot, brown spot, and powdery mildew) coexist. In the visualizations, blue bounding boxes represent sooty leaf spot disease, red bounding boxes represent brown spot disease, and orange bounding boxes represent powdery mildew disease.

Figure 9 shows the detection of mulberry leaves with both brown spot and powdery mildew by four models. As can be seen from Figure 9a, YOLOv8-RFMD has learned and detected various disease characteristics well, while YOLOv8n had several missed detections, one false detection, and one case of overlapping detection boxes. YOLOv7-tiny had false detections and missed detections, and YOLOv5s had several missed detections and overlapping boxes. This indicates that YOLOv8-RFMD has learned the subtle features of several diseases well, and can still precisely identify them even under complex detection conditions, significantly reducing the occurrences of missed and false detections.

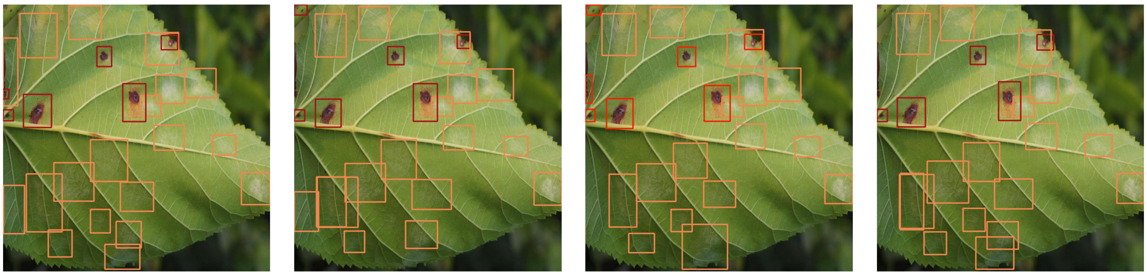

(**a**) YOLOv8-RFMD (**b**) YOLOv8n (**c**) YOLOv7-tiny (**d**) YOLOv5s

Figure 9. Brown spot and Powdery mildew.

Figure 10 shows the detection of mulberry leaves with both powdery mildew and sooty leaf spot by four models. On leaves with both powdery mildew and sooty leaf spot, YOLOv8n, YOLOv7-tiny, and YOLOv5s showed insufficient recognition ability for the small spots of sooty leaf spot, all exhibiting missed detections. YOLOv7-tiny and YOLOv5s missed a case of powdery mildew with less distinct features, YOLOv7-tiny falsely detected a case of powdery mildew, and YOLOv5s failed to properly recognize the characteristics of a case of powdery mildew, resulting in overlapping detection boxes.

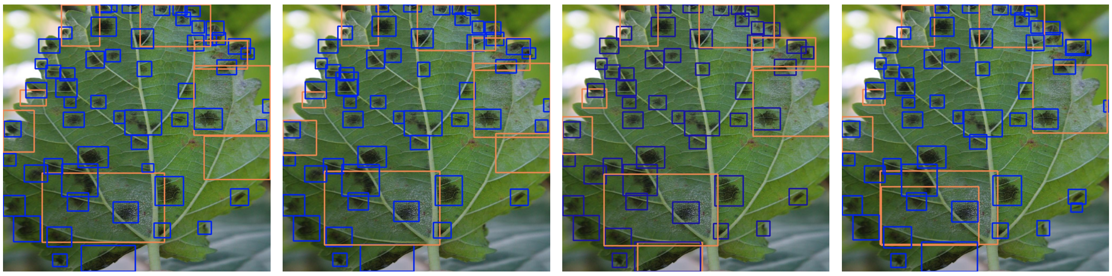
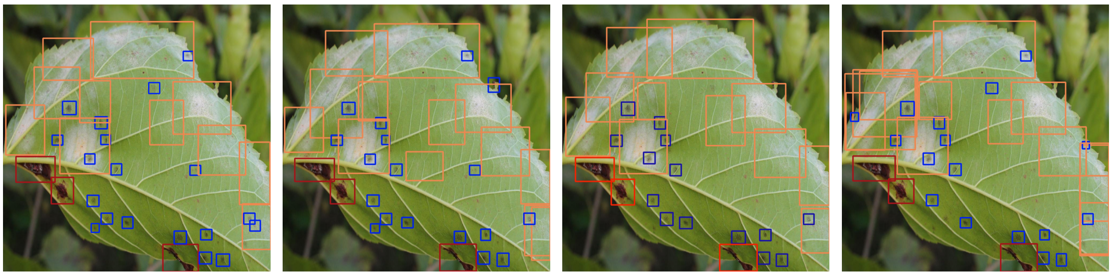

(**a**) YOLOv8-RFMD (**b**) YOLOv8n (**c**) YOLOv7-tiny (**d**) YOLOv5s

Figure 10. Sooty leaf spot and Powdery mildew.

Figure 11 shows the detection of mulberry leaves with three diseases present simultaneously by four models. On leaves with all three diseases present, YOLOv8n exhibited false detections, missed detections, and overlapping detection boxes. YOLOv7-tiny had missed detections, and YOLOv5s had missed detections, false detections, and overlapping detection boxes.



(**a**) YOLOv8-RFMD (**b**) YOLOv8n (**c**) YOLOv7-tiny (**d**) YOLOv5s

Figure 11. Three diseases.

After comprehensive comparison, YOLOv8-RFMD was found to be the most precise in identifying and detecting diseases, with detection boxes that more precisely reflect the actual size of the diseases. It rarely encountered issues with overlapping boxes, missed detections, or false detections, problems which were frequent in the other comparison models. This indicates that YOLOv8-RFMD has effectively learned the characteristics of diseases of various scales and shapes, demonstrating higher confidence in disease recognition and detection on mulberry leaves, more precision target localization, stronger model robustness, and better detection performance. It effectively resolves the issues of poor early disease recognition and inunprecision localization faced by existing models in natural environments.

4. Discussion

Under natural environmental conditions, the intensity of light, weather, and other factors can cause changes in the color and texture of lesions on mulberry leaves. Additionally, at the early stages of disease onset, the size of lesions can vary, and multiple diseases may coexist on the same leaf. Given these complex scenarios, the performance of YOLOv8n is difficult to meet the needs of our subsequent research. Therefore, we have made various improvements to the YOLOv8 model.

Firstly, to improve the detection precision and localization capability of the model for diseases of different sizes, we replaced the Bottleneck in the original model's C2f module with the RFMD Module. The RFMD Module includes the RFCA Conv and the MDFA module. After this replacement, the model's mAP50 and mAP50:95 increased to 92.6% and 65.4%, respectively. However, the model size and the number of floating-point operations increased, necessitating further improvements to optimize the model.

Next, to meet the requirement of deploying the algorithm to mobile embedded devices in the future, we need to make the model more lightweight and simpler. We replaced the CBS modules in the backbone network's P3, P4, and P5 layers and the CBS modules in the neck network with ADown modules. This change enhanced the precision while significantly reducing the model size and the number of floating-point operations. After the replacement, the model size was reduced to 5.20 MB, and the number of floating-point operations decreased to 7.4 G, achieving a lightweight model.

Finally, considering that some lesions on mulberry leaves are small in the early stages of disease onset, and the original model cannot effectively focus on small targets, we replaced the CIOU loss function with the NWD loss function. This effectively improved the detection precision for small targets. Through the above series of improvements, we ultimately established the YOLOv8-RFMD mulberry leaf disease detection model. A comprehensive comparison with various mainstream models showed that YOLOv8-RFMD has an absolute advantage in terms of detection precision and model complexity.

In future research, we will first further improve the quality of the mulberry leaf disease dataset. We will capture and annotate images of other mulberry leaf diseases from different angles, varieties, and weather conditions as much as possible to enhance the generalization ability of this study's mulberry leaf disease detection. Next, we plan to deploy the YOLOv8-RFMD model to mobile embedded devices and test its detection performance. This will provide more reliable technical support for the automated application of pesticides in mulberry plantations. Lan et al. [37] successfully deployed a ginger leaf pest detection model on Jetson Orin NX and tested and analyzed its performance, providing an effective reference for our future implementation. Finally, to further improve practical application capabilities, considering the actual needs of mulberry plantations, we also plan to develop an intelligent mulberry leaf disease monitoring system. This system will be able to call real-time video feeds from surveillance cameras, mobile phones, and drones into the YOLOv8-RFMD algorithm, enabling timely and precision feedback on mulberry leaf disease monitoring to management personnel.

5. Conclusions

This study proposed a target detection model for mulberry leaf diseases in natural mulberry garden environments based on the YOLOv8 model, named YOLOv8-RFMD. It offerd a new approach for more lightweight and precision identification and detection of diseases on mulberry leaves.

(1) This study proposed an MDFA module that selects important feature information from pixel-level dimension, channel dimension and spatial dimension. The feature maps processed by MDFA not only enhanced the extraction of effective information from channels but also contained global and local information on the spatial dimension. The CBS module, RFCA Conv, and MDFA module together formed the RFMD Module, which replaced the Bottleneck in the original YOLOv8's C2f module to create RFMD-C2f. RFMD-C2f was applied to the position of the original model's C2f module, where RFCA Conv focuses on important local features at the receptive field level and can locate disease positions more precisely. The ADown downsampling structure replaced the CBS modules in P3, P4, P5 of the original YOLOv8 backbone network and in the neck network, utilizing various downsampling methods and feature extraction strategies to avoid loss of important information that might be caused by a single downsampling method, and reduced model size and computational complexity. NWD Loss was used to replace the original YOLOv8's CIOU loss function, enhancing detection precision for small disease features through a new measurement method.

(2) The improved lightweight model was experimentally compared with other mainstream detection models, and the results showed that YOLOv8-RFMD increased mAP50 by 2.9% and mAP50:95 by 4.3% relative to the original model, with a reduction in model size by 0.53 MB and FLOPs by 0.3 G. The algorithm model improved in this study is relatively simple, with FLOPs only at 7.8 G, meeting the deployment conditions for mobile embedded devices. It provides technical support for intelligent spraying equipment for mulberry leaves and offers more precise disease diagnosis for mulberry gardens and other professionals.

Author Contributions: methodology, M.Zhang.; validation, X.Q., M.Zhao. and H.L.; investigation, C.Y.; Data curation, C.Y. and H.L.; writing—original draft preparation, M.Zhang.; Writing—review and editing, X.Q.; visualization, M.Zhang. and Q.L.; supervision, Q.L.; Funding acquisition, Q.L. and M.Zhao. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Postgraduate Research & Practice Innovation Program of Jiangsu Province(SJCX24_2495) and The Earmarked Fund for CARS-18

Data Availability Statement: Not applicable.

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

Abbreviations

The following abbreviations are used in this manuscript:

YOLOv8	You Only Look Once version 8
MDFA	Multi Dimension Feature Attention
CBS	Conv-BatchNomalization-SiLU
C2f	Faster Implementation of CSP Bottleneck with 2 convolutions
C3	CSP Bottleneck with 3 convolutions
NWD	Normalized Wasserstein Distance
IoU	Intersection over union
mAP	Mean average precision
mAP50	MAP Values at 50% loU threshold
mAP50:95	MAP Values in the 50-95% loU threshold range
VGG	Visual Geometry Group
CNN	Convolutional neural network
Faster R-CNN	Faster region-based convolutional neural networks
SSP	Spatial pyramid pooling
CSP	Cross Stage Partial
ELAN	Efficient Layer Aggregation Network
SPPF	Spatial pyramid pooling fusion
PAN	Path aggregation network
FPN	Feature pyramid network
DF Loss	Distribution focal loss
CIOU	Complete intersection over union
BCE	Binary cross-entropy
SE	Squeeze-and-excitation
ECA	Efficient Channel Attention
CBAM	Convolutional Block Attention Module
UNAP	Un average pooling
RFCA	Receptive-Field Coordinated Attention
CA	Coordinated Attention
SSD	Single Shot Multibox Detector
MLCA	Mixed Local Channel Attention
EMA	Efficient Multi-Scale Attention
LSKA	Large Separable Kernel Attention
FLOPs	Floating point of per second

References

1. Rohela, G.K.; Shukla, P.; Kumar, R.; Chowdhury, S.R. Mulberry (Morus spp.): An ideal plant for sustainable development. *Trees, Forests and People*, **2020**, *2*, 100011.[CrossRef]
2. Reddy, M.P.; Deeksha, A. Mulberry leaf disease detection using yolo. *International Journal of Advance Research, Ideas and Innovations in Technology*, **2021**, *7*, 3.
3. Gnanesh, B.N.; Arunakumar, G.S.; Tejaswi, A.; Supriya, M.; Pappachan, A.; Harshitha, MM. Molecular Diagnostics of Soil-Borne and Foliar Diseases of Mulberry: Present Trends and Future Perspective. *The Mulberry Genome*, **2023**, 215-241.[CrossRef]
4. Ngugi, H.N.; Ezugwu, A.E.; Akinyelu, A.A.; Abualigah, L. Revolutionizing crop disease detection with computational deep learning: a comprehensive review. *Environmental Monitoring and Assessment*, **2024**, *196*, 3, 302.[CrossRef]
5. Javidan, S.M.; Banakar, A.; Vakilian, K.A.; Ampatzidis, Y. Diagnosis of grape leaf diseases using automatic K-means clustering and machine learning. *Smart Agricultural Technology*, **2023**, *3*, 100081.[CrossRef]
6. Sladojevic, S.; Arsenovic, M.; Anderla, A.; Culibrk, D.; Stefanovic, D. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, **2016**, *2016*, 1, 3289801.[CrossRef]
7. Krizhevsky, A.; Sutskever, I.; Hinton, G.E.; ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, **2017**, *60*, 6, 84-90.[CrossRef]
8. Rangarajan, A.K.; Purushothaman, R.; Ramesh, A. Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia computer science*, **2018**, *133*, 1040-1047.[CrossRef]

9. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
10. Nahiduzzaman, M.; Chowdhury, M.E.H.; Salam A.; Nahid, E.; Ahmed, F.; AL-Emadi, N.; Ayari, M.A.; Khandakar, A.; Haider, J. Explainable deep learning model for automatic mulberry leaf disease classification. *Frontiers in Plant Science*, **2023**, *14*, 1175515.[CrossRef]
11. Waheed, A.; Goyal, M.; Gupta, D.; Khanna, A.; Hassanien, A.E.; Pandey, H.M. An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. *Computers and Electronics in Agriculture*, **2020**, *175*, 105456.[CrossRef]
12. Wen, C.; He, W.; Wu, W.; Liang, X.; Yang, J.; Nong, H.; lAN, Z. Recognition of mulberry leaf diseases based on multi-scale residual network fusion SENet. *Plos one*, **2024**, *19*, 2, e0298700.[CrossRef]
13. Xue, Z.; Xu, R.; Bai, D.; Lin, H. YOLO-tea: A tea disease detection model improved by YOLOv5. *Forests*, **2023**, *14*, 2, 415. [CrossRef]
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2017**, *39*, 6, 1137-1149. [CrossRef]
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part I 14.
16. Li, Y.; Sun, S.; Zhang, C.; Yang, G.; Ye, Q. One-stage disease detection method for maize leaf based on multi-scale feature fusion. *Applied Sciences*, **2022**, *12*, 16, 7960. [CrossRef]
17. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Nashville, TN, USA, 20-25 June 2021.
18. Nie, X.; Wang, L.; Ding, H.; Xu, M. Strawberry verticillium wilt detection network based on multi-task learning and attention. *IEEE access*, **2019**, *7*, 170003-170011. [CrossRef]
19. Dwivedi, R.; Dey, S.; Chakraborty, C.; Tiwari, S. Grape disease detection network based on multi-task learning and attention features. *IEEE Sensors Journal*, **2021**, *21*, 16, 17573-17580. [CrossRef]
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
21. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Vancouver, BC, Canada, 17-24 June 2023.
22. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
23. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
24. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21-26 July 2017.
25. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI conference on artificial intelligence, New York Hilton Midtown, New York, New York, USA, 7–12, February 2020.
26. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18-23 June 2018.
27. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Seattle, WA, USA, 13-19 June 2020.
28. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018,.
29. Yang, L.; Zhang, R.Y.; Li, L.; Xie X. Simam: A simple, parameter-free attention module for convolutional neural networks. In Proceedings of the 38th International conference on machine learning, Virtual Only, 18-24 July 2021.
30. Zhang, X.; Liu, C.; Yang, D.; Song, T.; Ye, Y.; Li, K.; Song, Y. Rfaconv: Innovating spatial attention and standard convolutional operation. *arXiv* **2023**, arXiv:2304.03198.

31. Wang, C.Y.; Yeh, I.H.; Liao, H.Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
32. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2021**, arXiv:2110.13389.
33. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, Venice, Italy, 22-29 October 2017.
34. Wan, D.; Lu, R.; Shen, S.; Xu, T.; Lang, X.; Ren, Z. Mixed local channel attention for object detection. *Engineering Applications of Artificial Intelligence*, **2023**, *123*, 106442. [CrossRef]
35. 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, 04-10 June 2023.
36. Lau, K.W.; Po, L.M.; Rehman, Y.A.U. Large separable kernel attention: Rethinking the large kernel attention design in cnn. *Expert Systems with Applications*. **2024**, *236*, 121352. [CrossRef]
37. Lan, Y.; Sun, B.;Zhang, L.; Zhao, D. Identifying diseases and pests in ginger leaf under natural scenes using improved YOLOv5s. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*. **2024**, *40*, 1, 210-246.

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.