applied sciences

MDPI

Article

CEMLB-YOLO: Efficient Detection Model of Maize Leaf Blight in Complex Field Environments

Shengjie Leng, Yasenjiang Musha *, Yulin Yang and Guowei Feng

School of Mechanical Engineering, Xinjiang University, Urumqi 830046, China; xjulsj@outlook.com (S.L.)
* Correspondence: yasenjiangmusha@163.com

Abstract: Northern corn leaf blight is a severe fungal disease that adversely affects the health of maize crops. In order to prevent maize yield decline caused by leaf blight, we propose the YOLOv5-based object detection lightweight models to rapidly detect maize leaf blight disease in complex scenarios. Firstly, the Crucial Information Position Attention Mechanism (CIPAM) enables the model to focus on retaining critical information during downsampling to reduce information loss. We introduce the Feature Restructuring and Fusion Module (FRAFM) to extract deep semantic information and make the feature map fusion across maps at different scales more effective. Thirdly, we add the Mobile Bi-Level Transformer (MobileBit) to the feature extraction network to help the model understand complex scenes more effectively and cost-effectively. The experimental results demonstrate that the proposed model achieves 87.5% mAP@0.5 accuracy on the NLB dataset, which is 5.4% higher than the original model.

Keywords: attention mechanism; cross-scale fusion; lightweight; maize leaf blight

Citation: Leng, S.; Musha, Y.; Yang, Y.; Feng, G. CEMLB-YOLO: Efficient Detection Model of Maize Leaf Blight in Complex Field Environments. *Appl. Sci.* **2023**, *13*, 9285. https://doi.org/10.3390/app13169285

Academic Editor: Stéfano Frizzo Stefenon

Received: 20 July 2023
Revised: 14 August 2023
Accepted: 14 August 2023
Published: 16 August 2023

1. Introduction

Corn is one of the world's major cereal crops, second only to wheat and rice in terms of cultivation area, and it serves a vital role as an essential feed and industrial raw material [1]. Northern maize leaf blight (NLB), caused by the phytopathogenic fungus Setosphaeria turcica, occurs frequently in northern China and greatly restricts photosynthesis and the transport of nutrients in the maize leaves, seriously affecting the yield and quality of the maize. As a result, the most critical task for maize producers is to detect whether maize is contaminated with NLB in a timely and accurate manner, thereby preventing the spread of the disease and the resulting decrease in maize production.

Currently, the primary method for detecting NLB is still visual identification, but it is difficult for inexperienced growers to identify similar diseases with the naked eye, leading to inappropriate pesticide applications that affect maize yield and quality, while relying on plant pathologists to identify disease types on site is time-consuming, inefficient and prone to subjective errors, especially in large field environments, significantly increasing labour costs. Many researchers have increasingly utilized machine vision and image-processing techniques to overcome the limitations of manual detection [2,3]. The idea of these studies is often based on the analysis of the colour, texture and spatial structure of the image, using edge arithmetic, threshold segmentation clustering and other methods [4–7], but it is difficult to meet the natural conditions of complex background images; there are poor adaptability, weak anti-interference ability and other problems, leading to serious limitations in the practical application [8].

Compared to the detection of other crop diseases, the small size of the disease area spots on the leaves of maize leaf blight in the early stages of the pathology, coupled with interfering factors such as the growth chain, lighting, climatic conditions and shading, poses a huge challenge to the visual detection of maize leaf blight. This requires that the algorithm model should have the ability to accurately detect small targets and understand

complex scenes. To overcome these challenges, we explore the visual detection of maize leaf blight from the perspective of digital image analysis, proposing the lightweight, robust model named CEMLB-YOLO based on YOLOV5. The specific contributions of the research are as follows:

1. We introduced a key information position attention mechanism into our model to enhance critical information representation in the feature map, reducing information loss during the downsampling process.
2. To aggregate global context data more effectively and affordably, the MobileBit is added to the feature extraction network to improve the model's ability to understand complex scenarios.
3. To exploit the deep feature map's potential for semantic information, FRAFM is incorporated into the model to reorganize and up-sample the semantic information of the deep feature map while adaptively adjusting the proportion of cross-scale feature map information for efficient feature aggregation.

2. Related Work

2.1. Object Detection Algorithm Based on CNN

In 2012, Krizhevsky et al. proposed AlexNet [9], a deep convolutional neural network-based image classification system. AlexNet achieved remarkable results in the ImageNet image classification competition, causing CNNs to gain significant attention. Object detection algorithms based on deep neural networks have advanced rapidly since then.

There are two types of CNN-based object detection algorithms: two-stage detection based on candidate regions and one-stage detection based on regression. The R-NN (R-CNN [10], Fast R-CNN [11], Faster R-CNN [12]) series is a representative two-stage algorithm series that can achieve better detection accuracy but is far from real-time in terms of speed. Single-stage algorithms represented by the SSD series [13–15] and YOLO series [16–18] have comparatively fewer parameters and superior real-time performance but inferior detection accuracy.

2.2. Plant Disease Detection Based on Convolutional Networks

As an effective feature extraction tool, convolutional networks have a broad range of applications in crop disease detection. Liao [19] combined the preliminary feature information obtained from manually extracted texture features and colour features with the high-level semantic information extracted with ResNeXt through a graph attention mechanism to achieve strawberry disease type classification. Xie [20] introduced the Inception module and SE module to modify the backbone network of Faster R-CNN and designed a bidirectional region candidate structure to locate grape disease lesion spots. Liu [21] proposed a lightweight model for the real-time detection of tomato leaf diseases by combining YOLOV3 with MobileNetV2; the proposed model accurately detects various types of tomato leaf diseases while maintaining a fast processing speed. Zhao [22] introduced the CBAM attention mechanism and adopted the pyramid structure to construct a multi-scale Faster R-CNN for detecting common strawberry diseases in natural environments. The multi-scale structure enables the network to effectively detect small and large strawberry lesions. Lv [23] developed the DMS-Robust AlexNet model by incorporating cavity convolution and multi-scale convolution into the AlexNet architecture, enhancing the model's feature extraction capabilities and showing strong robustness when detecting maize disease in the natural environment. Afzaal [24] constructed a Mask R-CNN architecture for detecting strawberry diseases using ResNet-101 as the model backbone, providing a foundation for future research in this field. Albattah [25] proposed an improved CenterNet algorithm to identify diseased and healthy leaves of tomatoes, using the Plant Village Kaggle database as the main data source and DenseNet-77 as the base network for deep-level key point extraction.

3. CEMLB-YOLO Network Model

YOLO is an end-to-end target detection algorithm proposed by Joseph Redmon, which divides the image into a number of $S \times S$ grids and predicts the bounding box and species probabilities for each grid cell. Compared to other object detection models, the YOLO series is more capable of meeting various conditions in industrial applications, which has led to widespread attention. YOLOv5 [26], the most widely used version of the YOLO series, has the advantages of fast detection and strong generalization ability; it is widely used, and, in recent years, scholars have proposed YOLOv7 [27], YOLOv8 [28] and more excellent YOLO series of detection models. YOLOv5, YOLOv7 and YOLOv8 can generate different variants, such as YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x, YOLOv7, YOLOv7x, YOLOv8n, YOLOv8s, YOLOv8m and YOLOv8l, by adjusting the width and depth multipliers of the network model.

However, YOLO as a single-stage algorithm. There is still potential for improvement, and many scholars have proposed improvements based on the YOLO series of detection algorithms. Souza [29] and Stefenon [30] proposed a hybrid architecture YOLO, which first detects defective insulators in transmission lines through the YOLO detection algorithm, then slices the defective insulators out of the picture and adds a new convolutional network for secondary classification to achieve higher accuracy than simple YOLO. Yao [31] proposed an adaptive feature fusion pyramid, which can better achieve cross-scale feature fusion and added multi-branch cavity convolution, which improves the model's long-range sensing ability. Xu [32] introduced the coordinate attention mechanism in YOLOv5 to increase the model's ability to detect small targets and, secondly, to improve the model's feature extraction ability by replacing the model loss function.

3.1. Architectureof CEMLB-YOLO

In this study, we aim to utilize YOLOV5 to create a lightweight model for identifying maize leaf blight to reduce the model complexity and enhance detection speed; we employ MobileNetV3 [33] as the backbone network for feature extraction. FRAFM is based on the idea of CARAFE [34]; it performs up-sampling by extracting the potential semantic information of a high-level feature map and reassembling the feature information. It adaptively adjusts the proportion of information at different scales in the feature map, achieving more effective cross-scale feature information fusion. MobileBit combines the advantages of inductive bias in CNNs and long-range perception in Vision Transformers [35] (VIT), enabling the model to balance the processing of local detail information and long-range information modelling capabilities. CIPAM first uses self-attention to fuse feature information from multiple channels, enhancing the representation ability of key features. Then it uses two-directional, one-dimensional pooling layers to encode the spatial position of key features to improve the model's ability to perceive their spatial locations. The overall architecture of the model is shown in Figure 1.

3.2. Mobile Bi-Level Vision Transformer

In complex and varied maize planting areas, it is essential for the model to capture global feature information to comprehend the whole scene. Such as, VIT divides the image into patches, and each patch calculates the affinity with other patches enabling the model to capture long-range dependencies effectively. However, this leads to higher model complexity and incurs heavy memory footprints, which is not conducive to model deployment for edge devices. In addition, VIT requires a larger amount of training data and longer training time due to the lack of convolutional inductive bias characteristics [36].

To solve the above problems, we propose a lightweight hybrid architecture that combines the convolution and transformer, which can effectively model both local and global information simultaneously and is easier deploy for the edge devices. The overall architecture of the MobileBit is shown in Figure 2.

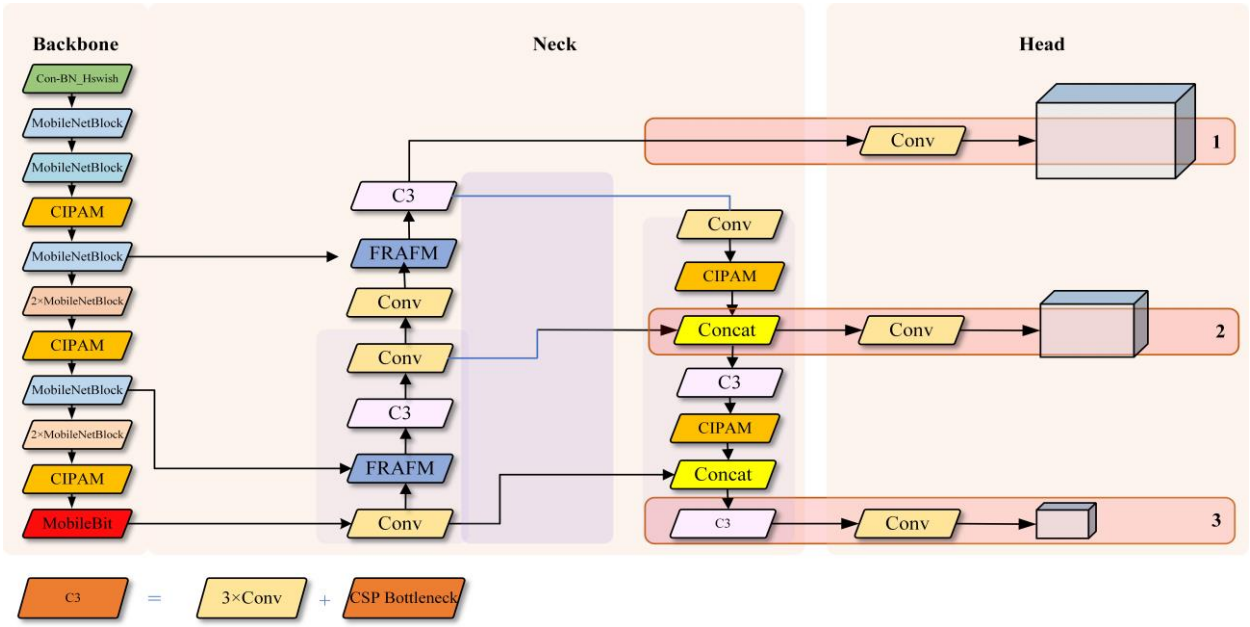


Figure 1. The overall architecture of the proposed model.

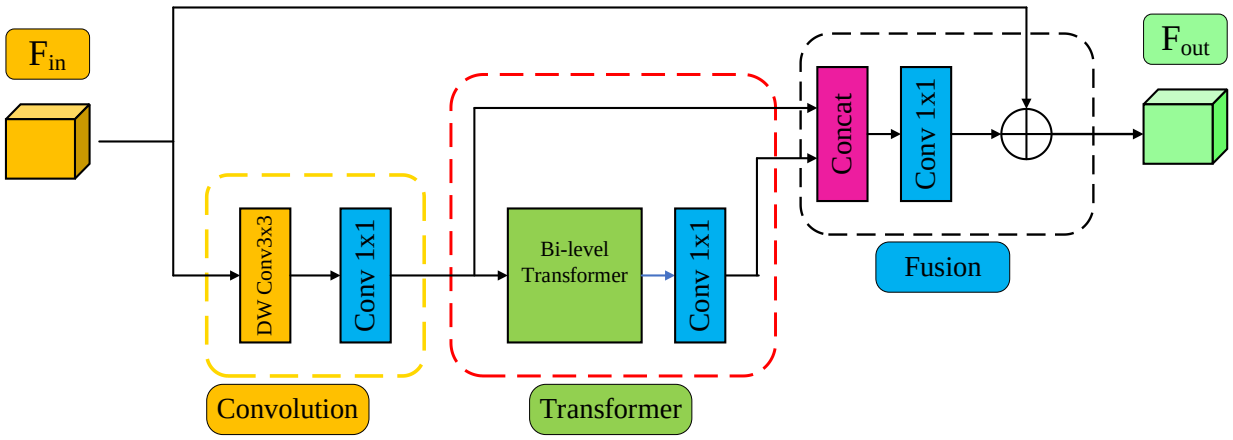


Figure 2. The overall architecture of the MobileBit.

MobileBit is divided into three sections: the Convolution section, Transformer section and Fusion section. In the Convolution section, we first use depth-wise separable convolution to encode the spatial information in the image and model the local features, adjusting the channel dimension of the feature map by 1×1 convolution to reduce the operation of the transformer. In the Transformer section, we use a bi-level transformer [37] to segment the image into several non-overlapping regions. For each region, only the most relevant K subregions are preserved for the execution of the self-attention mechanism. This selective approach not only enables the model to comprehend long-range perceptual correlations amongst non-overlapping regions but also significantly reduces the model's complexity. In the Fusion section, the local modelling information is concatenated with the global modelling information, then through 1×1 convolution to fuse the information.

Bi-Level Transformer

The bi-Level transformer first constructs a coarse-grained affinity graph of query-keys and performs pruning at the coarse-grained region level instead of directly at the fine-grained token level, retaining the most critical part for token–token attention, as shown in Figure 3. The bi-Level transformer divides the input feature map $X \in R^{H \times W \times C}$ into

non-overlapped areas and reshapes X to $X^r \in R^{S^2 \times \frac{HW}{S^2} \times C}$, then with linear projections to obtain $Q, K, V \in R^{S^2 \times \frac{HW}{S^2} \times C}$:

$$Q = X^r W^q, K = X^r W^k, V = X^r W^v \tag{1}$$

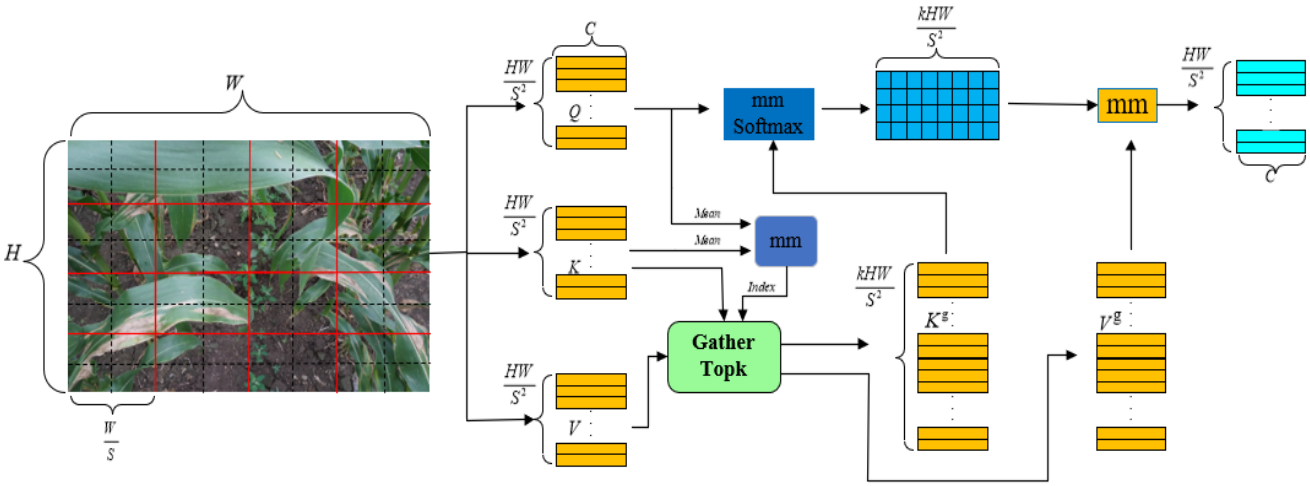


Figure 3. The overall architecture of the bi-level transformer with a coarse-grained relationship graph to filter the most relevant k candidate patches for each patch; then fine-grained token-to-token attention are applied to candidate patches.

W^q, W^k, W^v are projection weights for the query, key and value, respectively.

Then, the bi-Level transformer calculates the mean value Q, K of each patch to obtain the region-level $Q^r, K^r \in R^{S^2 \times C}$ and performs matrix multiplication between Q^r and the transpose K^r to derive the region-to-region affinity adjacency matrix $A^r \in R^{S^2 \times S^2}$:

$$A^r = Q^r (K^r)^T \tag{2}$$

A^r Indicates the degree of semantic information associated between the two regions. Next, only retain the k highest associated regions for each region, trimming A^r to obtain the region of interest index matrix $I^r \in R^{S^2 \times k}$. Finally, using the index matrix I^r to obtain the key-value pairs of the K most relevant regions associated with the i^{th} region and apply self-attention to the gathered key, the values are as follows:

$$I^r = \text{TopK}(A^r) \tag{3}$$

$$K^g = gather(A^r, I^r) \tag{4}$$

$$V^g = gather(A^r, I^r) \tag{5}$$

$$\text{Attention}(Q, K^g, V^g) = Softmax(\frac{Q(K^g)^T}{\sqrt{d_k}})V^g \tag{6}$$

The ith row of I^r indicates the k regions that are most relevant to the ith region. $K^g, V^g \in R^{S^2 \times \frac{kHW}{S^2} \times C}$ is the key-value pair tensor for each region token–token. $\sqrt{d_k}$ is used to avoid concentrated weights and gradient vanishing.

3.3. Feature Restructuring and Fusion Module

Multi-scale fusion features can improve the detection ability of the model, but the deep feature maps are often up-sampled by interpolation methods with a small sense field, which does not fully use the semantic information in the deep feature maps. Second, the information fusion ratio of different feature maps is 1:1, which cannot adjust the proportion of information in the feature maps.

To achieve more effective cross-scale fusion, we propose FRAFM, as shown in Figure 4. FRAFM first employs CARAFE to up-sample the deep feature map to preserve the intricate details embedded in the deep features. We use a Spatial Attention Mechanism (SAM) for shallow feature maps and a Channel Attention Mechanism(CAM) for deep feature maps to better highlight important information in feature maps at different scales. In addition, we concatenate shallow and deep feature maps, then pass them through a 3×3 convolutional layer, a Batch Normalization (BN) layer and a Sigmoid activation function to generate learnable weights, which are used to adjust the ratio of information contributed by feature maps of different scales during the fusion process. In the following subsections, we will delve into a detailed exploration of the CARAFE, CAM and SAM.

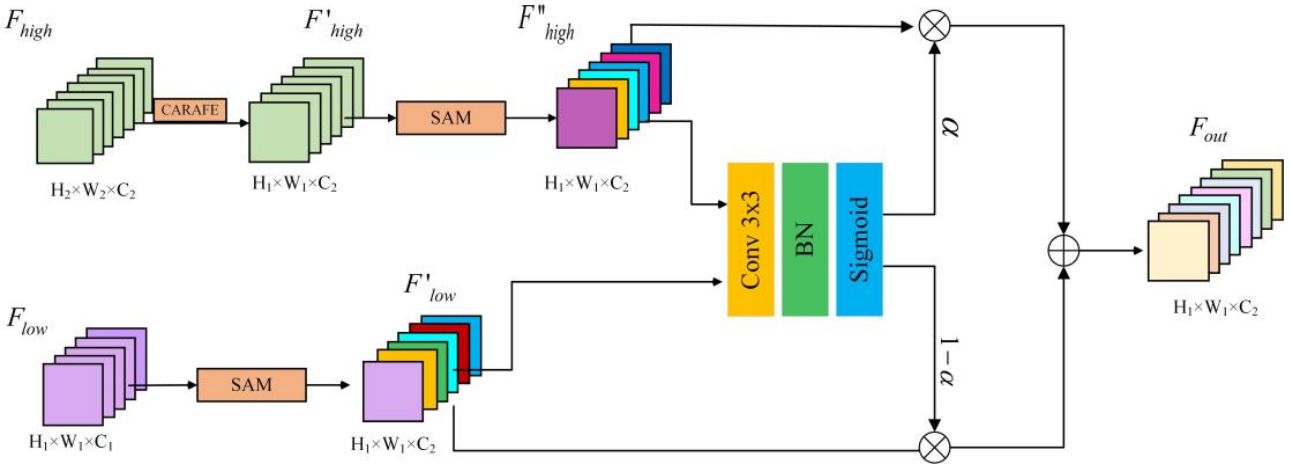


Figure 4. The overall architecture of FRAFM. The F_{low}, F'_{low} represents low feature maps, and the $F_{high}, F'_{high}, F''_{high}$ represents high feature maps, where $\alpha, 1-\alpha \in R^{1 \times H_1 \times W_1}$.

3.3.1. CARAFE

CARAFE consists of kernel prediction and content-aware reassembly modules, as shown in Figure 5. The kernel prediction module generates a reassembly kernel in a content-aware manner for the input feature map $\chi \in R^{C \times H \times W}$, using $k_{\text{encoder}} \times k_{\text{encoder}}$ convolution to generate a reassembly kernel $W_{l\prime}$ for each position in the target feature map $\chi\prime \in R^{C \times \sigma H \times \sigma W}$ based on $\chi \in R^{C \times H \times W}$; finally, it uses the Softmax function to normalize so that the sum of the weights of each convolution kernel is 1. For $l = (i\prime, j\prime)$ in $\chi \in R^{C \times H \times W}$, the content-aware reassembly module performs a dot product operation between square region N = (k_{up}, k_{up}) centred at $l = (i, j)$ in $\chi \in R^{C \times H \times W}$ and $W_{l\prime}$. The mathematical formula for CARAFE is expressed as shown in Equations (7) and (8), where $r = k_{up}/2$:

$$W_{l\prime} = \phi(X, k_{encoder}) \tag{7}$$

$$X = \sum_{n=-r}^{r} \sum_{m=-r}^{r} W_{l\prime(n,m)} \cdot X_{(i+n,j+m)} \tag{8}$$

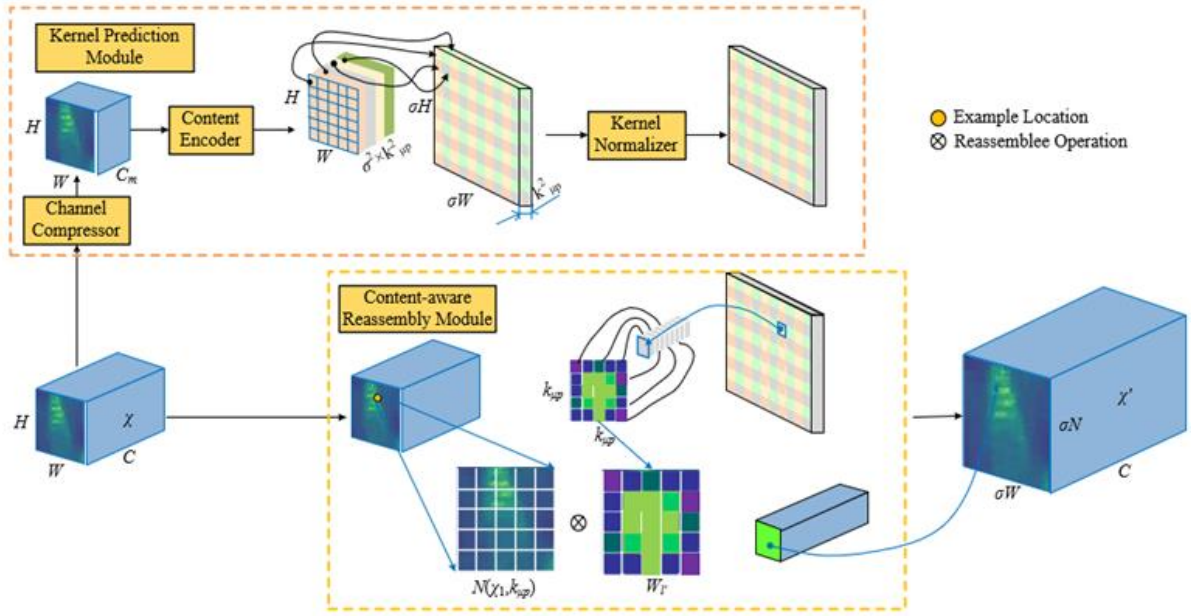


Figure 5. Schematic representation of the CARAFE up-sampling operator structure.

3.3.2. CAM and SAM

The CAM and SAM refer to the attention mechanism in CBAM [38], as shown in Figure 6. For deep feature maps F'_{high}, we use CAM to obtain key information and ignore redundant information. Specifically, CAM uses a pooling layer to compress the spatial dimension to obtain two features of dimension $C \times 1 \times 1$; then, through multi-layer perceptron, it determines the weights of each channel and, finally, multiplies the weights with F'_{high} to obtain F''_{high}:

$$F''_{high} = \sigma(MLP(Maxpool(F'_{high}) + MLP(Avgpool(F'_{high}))) \times F'_{high} \quad (9)$$

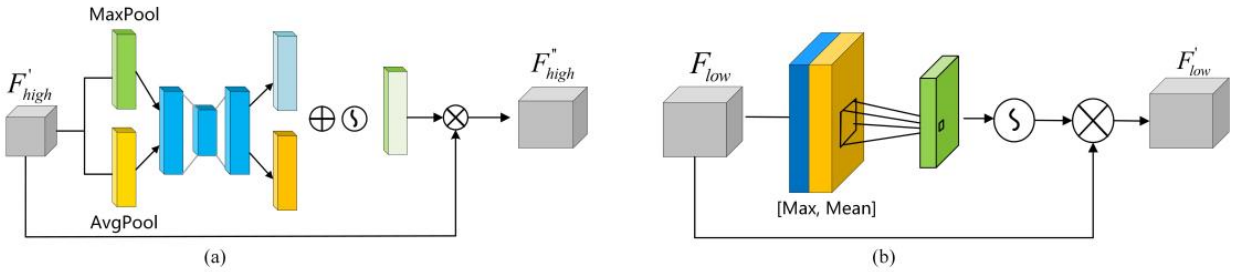


Figure 6. The overall architecture of the CAM and SAM. (**a**) is CAM that is used for F_H. (**b**) is SAM used for F_L. The F_L, F_H; respectively, they represent low feature maps and deep feature maps.

Spatial attention maintains the spatial dimension and compresses the channel dimension. For shallow feature map F_{low}, we use SAM to locate the location information of the target:

$$F'_{low} = \sigma(Conv_{7\times7}[Max(F_{low}); Mean(F_{low})]) \times F_{low} \quad (10)$$

3.4. Crucial Information Position Attention Mechanism

Maize leaf blight occurs in small and dense areas; some critical information is lost or blurred during the image feature extraction model's down-sampling procedure, impairing the model's detection capability.

To address this issue, we propose a Crucial Information Position Attention Mechanism (CIPAM) that helps the model to be able to focus on specific regions of important details and highlight the most informative regions. The model can retain and utilise the most

essential parts of the image even during the down-sampling process, reducing the potential loss of critical information. The structure is shown in Figure 7.

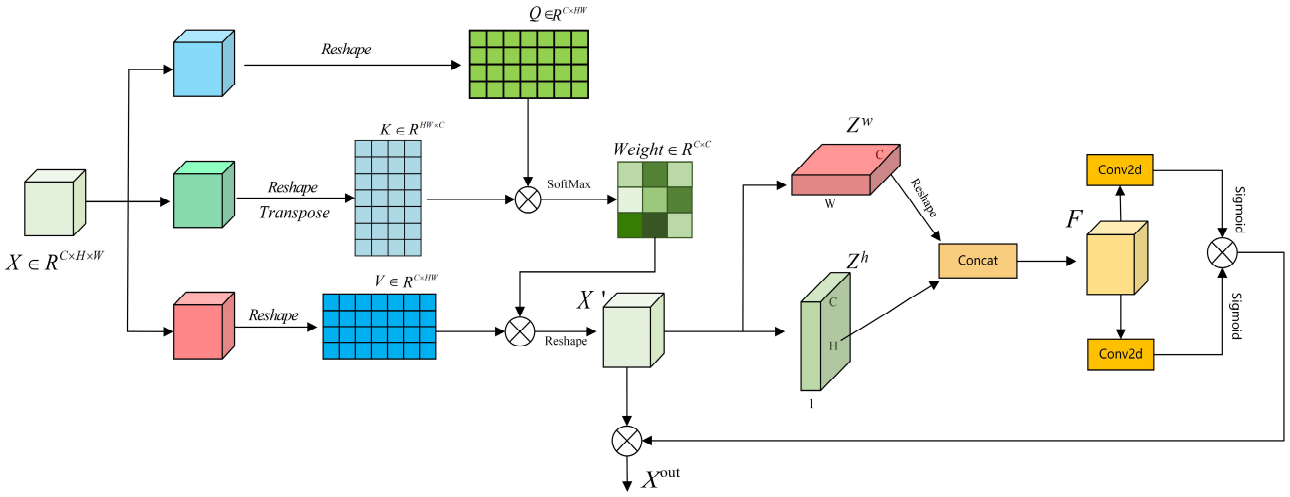


Figure 7. CIPAM structure. Each channel feature map is regenerated and then uses horizontal and vertical pooling layers to capture the generated key position information.

For the input feature map $X \in R^{C\times H\times W}$, CIPAM first constructs interdependencies between channels of the feature map using a self-attention approach. Specifically, the feature map of the i^{th} channel is then reweighted and fused with the feature maps of other channels based on their correlation coefficients to enable information interaction between different channels and enhance the representation of crucial information. For the i^{th} channel of the feature map, it is as follows:

$$Weight_{ij} = \frac{\exp(Q_i \cdot K_j{}^T)}{\sum_{i=1}^{C} \exp(Q_i^T \cdot K_j)} \tag{11}$$

$$X_i' = \sum_{j=1}^{c} Weight_{ij} \cdot V_j \tag{12}$$

$Weight_{ij}$ denotes the correlation of the i^{th} channel with the j-th channel.

More importantly, in the complex and ever-changing natural environment, accurately pinpointing the location of plant diseases is crucial for enhancing the model's performance. In this article, we use CA [39] to capture the exact position. Specifically, for input feature maps $X \in R^{C\times H\times W}$, two one-dimensional pooling layers are used along the horizontal and vertical directions, respectively, to obtain Z^h, Z^w. The Z^h, Z^w captures both long-range dependencies and retains precise positional information of crucial information. Z^h, Z^w can be expressed mathematically as follows in Equations (13) and (14):

$$Z_c^w = \frac{1}{H}\sum\nolimits_{0\le i\le H} X_c'(i,w) \tag{13}$$

$$Z_c^h = \frac{1}{W}\sum_{0\le i\le W} X_c'(h,i) \tag{14}$$

The two directional feature maps are concatenated and passed through a 1×1 convolution layer to obtain the feature map F, representing the interaction between the height and width directions. After applying batch normalization and a non-linear activation function, the feature maps are split into two directional feature maps f^w, f^h. Then using the Sigmoid function to obtain the weights g^w, g^h of the feature maps in height and width. Finally, the

weights are multiplied by X'. The process uses mathematical expressions as shown in Equations (15)–(18):

$$F = \sigma(Conv(\left[Z^h, Z^w\right])) \tag{15}$$

$$g^h = \sigma(Conv(f^h)) \tag{16}$$

$$g^w = \sigma(Conv(f^w)) \tag{17}$$

$$X_c^{out} = X_c'(i,j) \times g^h(i) \times g^w(j) \tag{18}$$

The resulting feature map X^{out} significantly enhances the representation of crucial information in the feature map and accurately captures the location of such information. Our experiments demonstrate that our proposed strategy focuses better than previous attention techniques on the location of disease occurrence in complex field environments.

3.5. The Loss Function of CEMLB-Yolov5

The regression loss in YOLOV5 adjusts the position of the predicted bounding box by calculating the intersection over the union ratio between the ground truth box and the predicted box, as demonstrated by Equation (19). The continued research on loss functions, GIOU [40], DIOU [41] and CIOU [42], have been proposed. CIOU regression loss converges faster than other alternative regression losses. This paper adopts CIOU as the model's regression loss, with its expression shown in Equation (20):

$$Loss_{iou} = 1 - \frac{A \cap B}{A \cup B} \tag{19}$$

$$Loss_{CIOU} = 1 - IOU + \frac{\rho^2(b, b^{gt})}{d^2} + av \tag{20}$$

where A and B respectively, denote the area of the ground truth-bounding boxes and the predicted boxes, $\rho(\cdot)$ indicates the Euclidean distance between the predicted and true box centroids and d represents the diagonal distance between the smallest closed regions. α indicates the trade-off indicator. The value of v describes the similarity of the ground truth and bounding box shapes. α, v can be expressed mathematically as follows:

$$v = \frac{4}{\pi}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{21}$$

$$\alpha = \frac{v}{(1 - IOU) + v} \tag{22}$$

Generally, the distance between the centre of the predicted box and the true box increases as the size of an object increases. The larger the object, the more significant its contribution to the loss function, which can reduce the model's ability to detect smaller defects and result in false negatives. Therefore, in order to balance this difference, this paper takes the square root of the numerator when calculating CIOU. The enhanced expression of the CIOU function is as follows:

$$Loss_{CIOU} = 1 - IOU + \frac{\sqrt{\rho^2(b, b^{gt})}}{d^2} + av \tag{23}$$

4. Experimental Results and Analyses

4.1. Data

The NLB dataset [43] was created for detecting maize leaf blight disease and is the largest dataset of its kind, with each image annotated by one of two anthropologists. The

dataset was divided into three parts: the first was taken with a handheld camera device, the second part was taken by mounting the camera on a 5 m long boom and the third part was taken with a DJI Matrice 600 sUAS camera on board, flying at an altitude of 6 m and a speed of 1 m/s, capturing images every two seconds. The handheld datasets, which include 1019 images with different angles and backgrounds and 7669 annotations, are used in this research study because it offers clear and training-friendly images. Figure 8 displays the dataset example image.

Figure 8. Example image from handheld dataset.

We apply data augmentation techniques, such as overexposure, haze, rain, random rotation and random cropping, to the original dataset to mitigate the effects of a small dataset on the model training. This technique is randomly combined to generate a total of 9070 images. The dataset is split into training and validation sets at 7:3 ratio.

4.2. Experimental Configuration

The experiments were conducted under Windows 10 with the PyTorch deep learning framework, CUDA version 11.1, NVIDIA GeForce GTX3060 graphics card, 12 GB of video memory and a 12-core Intel(R) Xeon(R) Platinum 8255C CPU @ 2.50 GHz. The initial learning rate was set to 0.01, the optimizer was chosen from SGD [44], the momentum decay was set to 0.937, the weight decay was set to 0.0005, the epoch was set to 300 and the batch size was set to 36. During the experiment, the learning rate will be adjusted according to the cosine annealing strategy during the training process.

4.3. Model Evaluation Indicators

This study used average precision (AP) to evaluate the detection model's performance. AP uses a combination of Precision (P) and Recall (R) to evaluate the model's performance in detecting a particular class. AP evaluates the model's performance in detecting a specific class using a combination of Precision and Recall. Mean average-precision (mAP) is the average of the AP of multiple categories. mAP@0.5 is the average of the AP calculated for all categories when the IOU threshold between the predicted and true boxes is set to 0.5. In this paper, mAP@0.5 is used as the evaluation criterion for the model. The expressions that calculate P, R, AP and mAP are shown in Equations (24)–(27):

$$P = \frac{TP}{TP + FP} \tag{24}$$

$$R = \frac{TP}{TP + FN} \tag{25}$$

$$AP = \int_0^1 P(R)dR \tag{26}$$

$$mAP = \frac{1}{N}\sum AP \tag{27}$$

P is the proportion of targets detected by the model that actually belong to the target category; R is the proportion of the actual target that is detected. True Positive (TP) is the proportion of positive samples that are correctly predicted by the model as positive. False Positive (FP) is the number of false positive samples detected as negative by the model. False Negative (FN) is the number of false positive samples detected as negative samples quantity. N is the number of species in the detection dataset.

4.4. Analysis and Comparison of Experimental Results

We evaluate the superiority of the proposed CEMLB-YOLO algorithm in detecting maize leaf blight in complex field environments using other object detection algorithms as comparison experiments. Table 1 compares the detection effectiveness of CEMLB-YOLO with other models. In this paper, we replaced the backbone network in YOLOV5 with MobileNetv3 as the original model.

Table 1. Comparison experiments with other models.

Model	Imgsize	mAP	FPS	Parameters (m)	GFLOPs
Origin Model	640 × 640	82.1%	84	6.2	6.6
YOLO v3-tiny	640 × 640	76.2%	65	8.6	12.9
YOLOv7-tiny	640 × 640	81.3%	64	6.02	13.16
Faster R-CNN	640 × 640	92.4%	32	41.13	78.1
RetinaNet	640 × 640	91.6%	36	36.3	82
YOLOv8s	640 × 640	89.4%	69	11.1	28.06
YOLOV8n	640 × 640	78.7%	77	3.01	8.2
YOLOX [45]	640 × 640	91%	53	8.94	26.64
Song [46]	512 × 512	82.1	39	40.9	/
Sun [47]	512 × 512	91.8	28	/	/
Ours	640 × 640	87.5%	62	4.54	9.4

Based on the results of the comparative experiments in this paper, the proposed model performs well in detecting maize leaf blight in complex environments in the field. Compared to YOLOv3-tiny and YOLOv7-tiny, which have lower parameter quantities and complexity, our model proposed in this paper has lower parameter and model complexity but higher accuracy. Compared to YOLOv3-tiny, the accuracy of the model is improved by 11.3%, and the model and parameter amount is reduced by 4 Million, 2.5 GFLOPs.

The accuracy of our model is 6.2% higher than YOLOv7-tiny, and the model parameter amount and complexity are reduced by 1.5 Million, 3.7 GFLOPs. During the comparison experiments, we chose Resnet50 as the backbone network of Faster R-CNN and RetinaNet, which results in a higher complexity and a higher number of parameters of the model. The number of parameters and complexity of our model is one-tenth of the Faster R-CNN, RetinaNet, and the accuracy has been reduced by 4.9% and 4.1%. Compared with the emerging YOLOv8s and YOLOX, our model complexity is reduced by nearly 2/3, and the accuracy is only 1.9% less than YOLOv8s and 3.5% less than YOLOX. Compared to YOLOV8n, which has a similar number of model parameters and complexity, the accuracy of our model increases by 8.8%, while the number of parameters increases by only 150 w and 1.2 GFLOPs. We also compared our approach with other researchers, and our model is 5.4% more accurate than Song's and 4.3% less accurate than Sun's approach.

In this paper, FPS is introduced to evaluate the impact of different methods on the detection speed. From the table, we can see that our proposed model's detection speed is lower than YOLOv3-tiny, YOLOv7-tiny, YOLOV8n and YOLOv8s, which we analysed due to the introduction of MobileBit. Our model's detection speed is faster than YOLOX, RetinaNet, YOLOv7 and Faster RCNN models; particularly, the detection speed of our model is roughly twice as fast as the detection speed of Faster RCNN, RetinaNet. The detection speed of our model also shows a superior performance when compared to the detection speed of other researchers' methods.

Overall, our model achieves a better balance between detection speed and detection accuracy, and our model is more suitable for detecting maize leaf blight in complex environments.

4.5. Ablation Experiment

In this section, we have conducted a series of experiments to verify the validity of our proposed method. The results of the ablation experiments are shown in Table 2, and Figure 9 shows the mAP@0.5 curves of the ablation experiment.

Table 2. CEMLB-YOLO ablation experiment.

Method	P	R	mAP	GFLOPs	Size (MB)	FPS
Origin Model	88.2%	71%	82.1%	6.3	6.2	84
Origin Model + FRAFM	91.2%	76.2%	86.1%	6.7	7.9	78
Origin Model + FRAFM + MobileBit	92.5%	77.3%	86.7%	8.9	8.1	69
Origin Model + MobileBit	91%	76.3%	85.4%	8.5	7.4	71
Origin Model + CIPAM	88.6%	74.2%	83.4%	6.9	7.5	73
Origin Model + MobileBit + CIPAM	92.1%	76.1%	86.4%	9.1	8.25	69
Origin Model + FRAFM + CIPAM + MobileBit	93.4%	79.3%	87.5%	9.4	8.4	62

As can be seen from Figure 9, the number of convergence iterations of MobileBit is reduced by 50 rounds compared to the original model, which can effectively shorten the training time of the model; in addition, through the experimental ablation curve, it is observed that CIPAM can also accelerate the training of the model, proving that the model pays more attention to the location of disease occurrence and ignores irrelevant information; the combination of CIPAM + MobileBit can effectively accelerate the training of the model, as can be seen in the figure. It converges to around 120 fewer iterations than the original model; by adding the FRAFM module to alleviate the aliasing effects caused by feature fusion, the training time of the model can be accelerated even further.

From the table, it can be seen that all three methods proposed in this paper improve the model's detection capability. Specifically, the FRAFM module is the greatest improvement among the individual methods, resulting in a 4% increase in model accuracy, the combination of FRAFM and MobileBit achieves the highest accuracy improvement of 4.6% among the two-method combinations. The MobileBit+CIPAM combination improves the average precision by 4.3%, but it also introduces the highest increase in the number of model parameters and GFLOPs among all the methods. We observe that adding MobileBit

or CIPAM increases more GFLOPs due to the incorporation of self-attentive computation, but this increase in GFLOPs is within acceptable limits considering the significant improvement in model performance they provide. In this paper, FPS is introduced to evaluate the impact of different methods on the detection speed. It can be seen from the table results that the improved model still achieves real-time detection compared with the original detection method.

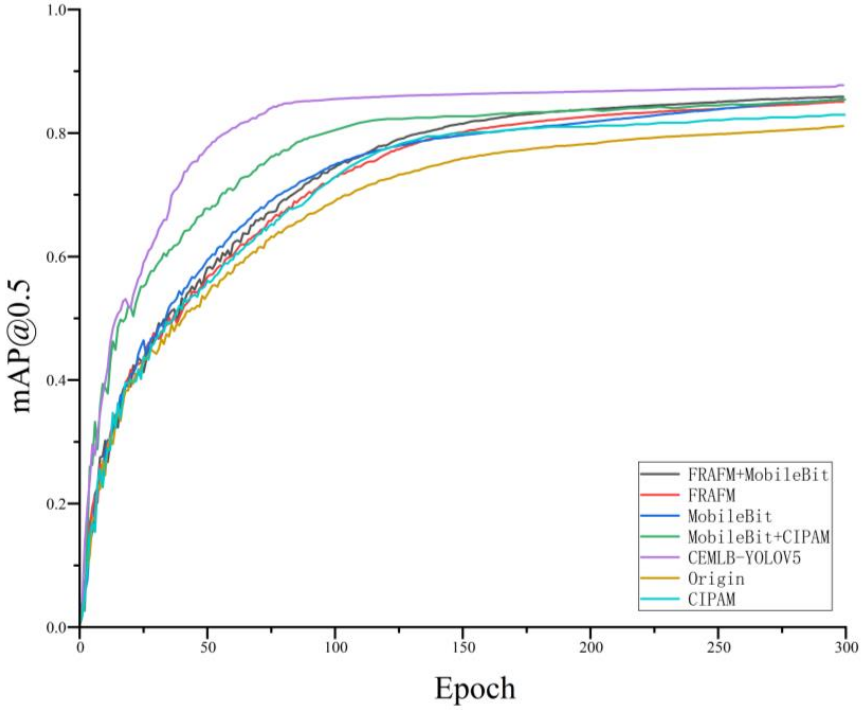
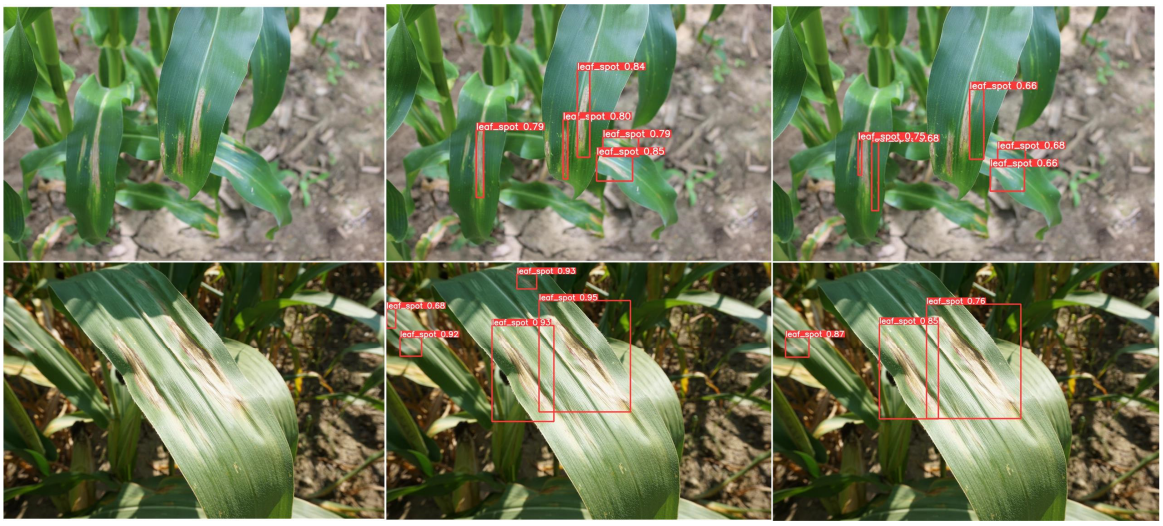


Figure 9. The mAP@0.5 curves of the ablation experiment.

In Figure 10, we show the impact of different improvement methods on the detection capability of the model. In Figure 10, the third column represents the detection results of the original model, and the middle represents the detection results of the different improvement methods.

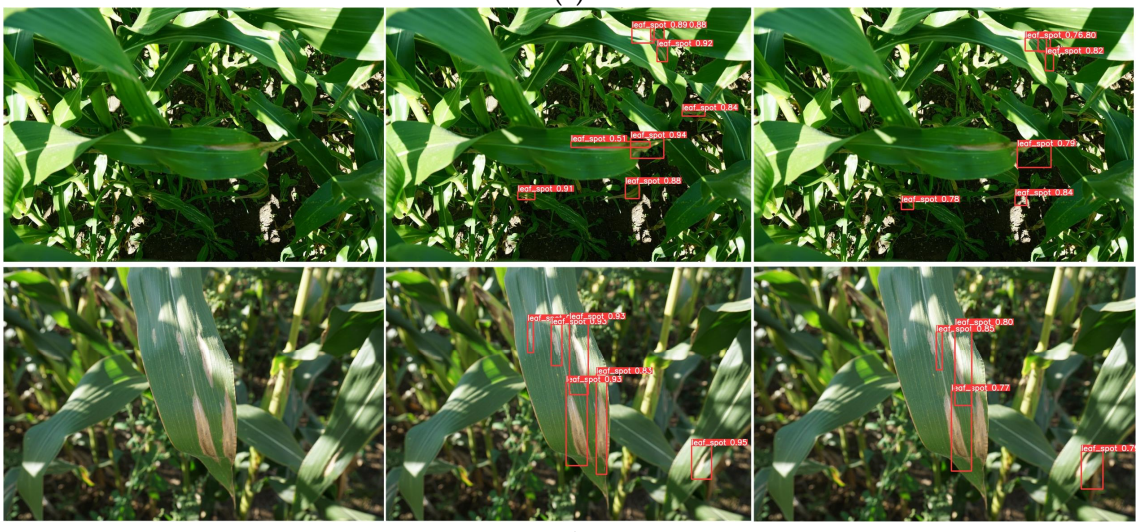


(**a**)

Figure 10. *Cont.*

(**b**)

(**c**)



(**d**)

Figure 10. *Cont.*

(**e**)

Figure 10. Comparison of detection results before and after improvement. (**a**) Original model + CIPAM detection results vs. original model detection results. (**b**) Original model + FRAFM detection results vs. original model detection results. (**c**) Original model + MobileBit detection results vs. original model detection results. (**d**) Original model +CIPAM +MobileBit detection results vs. original model detection results. (**e**) Original model + MobileBit + FRAFM detection vs. original model detection results.

Figure 10a shows that the original model has the problem of inaccurate detection of small-area diseases; adding CIPAM can help the model increase its ability to locate small-area diseases. In Figure 10c, the original model has a more serious leakage detection in complex scenes; adding MobileBit can increase the detection ability of the model in complex scenes and reduce leakage detection. In Figure 10d, the model's original detection results suffered from inaccurate localization and missed detection when the background information is complex and the disease location occurs in a small area. MobileBit + CIPAM method can effectively detect the location of the maize leaf blight disease in a complex background.

4.6. Visualization of Results

To further validate the effectiveness of our proposed CIPAM in focusing more effectively on the location of leaf blight disease in complex environments compared to other attention mechanisms, we used the Grad-CAM [48] method to visualize the detection results to see which part of the image the model is most concerned with to make a judgement. Some example images are presented in Figure 11. From the figure, CIPAM can effectively focus more on the location of disease occurrence, even in small and complex scenarios, proving the effectiveness of the CIPAM method compared to other methods.

During model training, we trained the model in the handheld portion of the dataset because this portion of the dataset is clearer, has more distinct disease features and is more friendly for model training. In addition, we tested the model in the other two parts of the dataset to verify the generalisation ability of our proposed model. The accuracy of our proposed model on the three partial datasets is shown in Table 3.

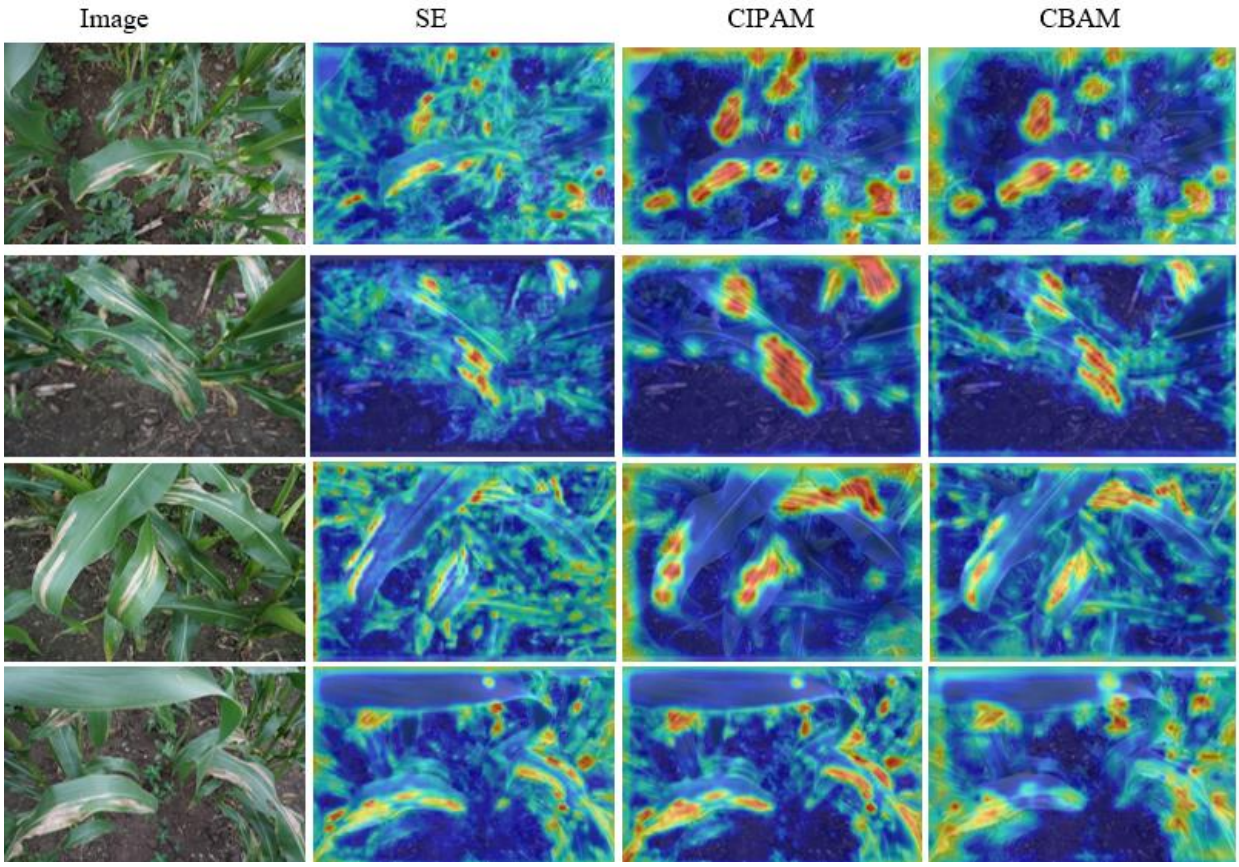


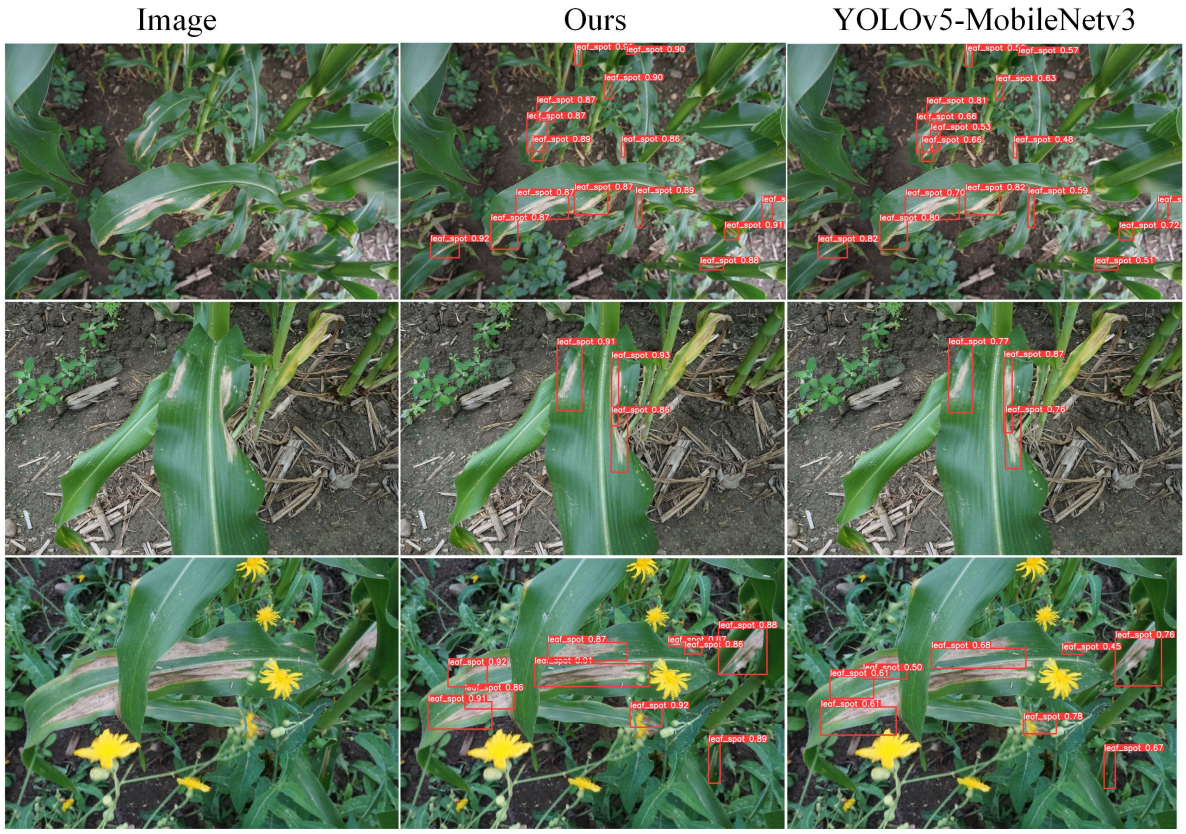
Figure 11. Visualization results of different methods. Experimental comparison group SE [49], CBAM, CIPAM can locate disease more accurately than other attention mechanisms, while SE and CBAM are sensitive to the approximate extent of disease location.

Table 3. Accuracy of the three-part dataset.

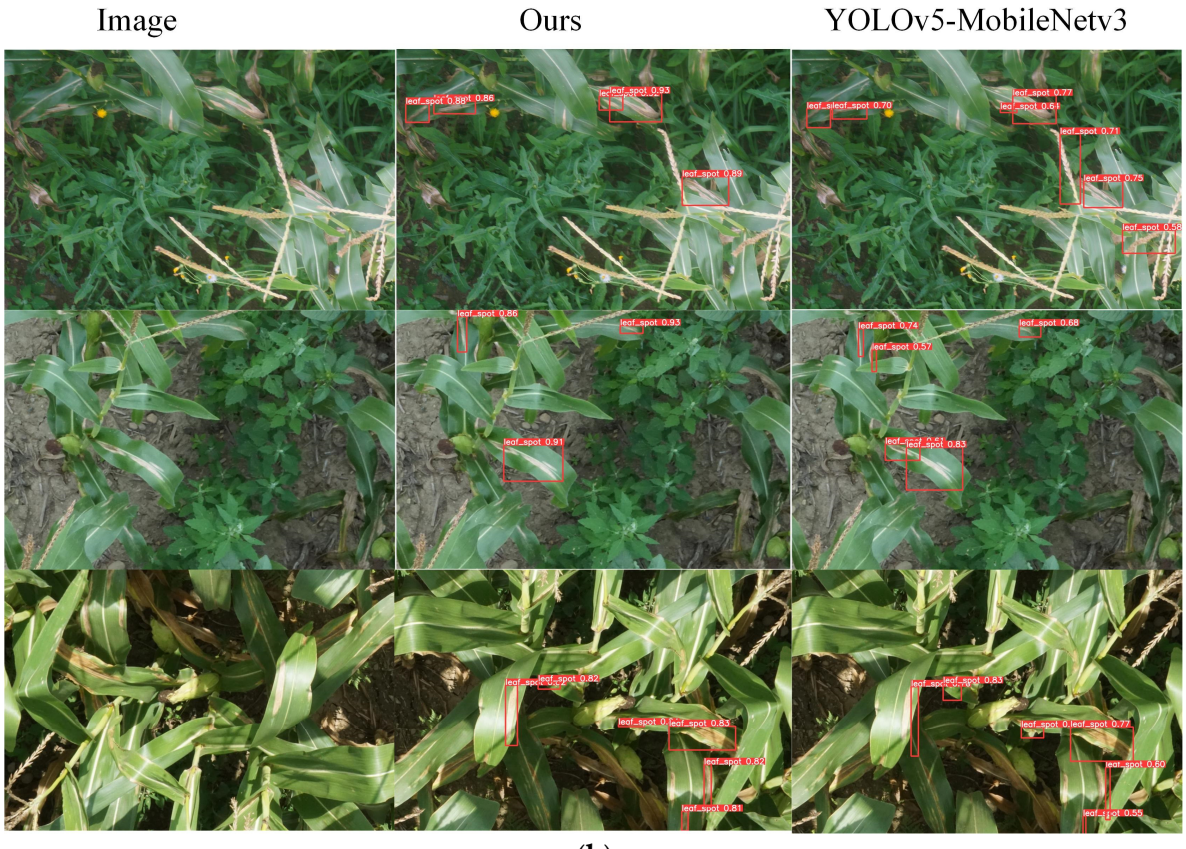
Method	Boom Set	Drone Set	Handheld Set
Origin Model	69.6%	71.3%	82.1%
Faster R-CNN	79.1%	75.7%	92.4%
Ours	84.3%	83.6%	87.5%

From the table, we can see that, although the Faster R-CNN model can achieve the highest accuracy in the handheld set, the detection accuracy in the boom set and the drone set decreases significantly by 13.3% and 16.7%. The detection accuracy of the original model in the two parts of the dataset decreases by 13.5% and 10.8%. We analyse that there are large differences in the shooting angle, background and illumination of the three datasets, which cause the model to fail to extract the disease features well.

Although the accuracy of the model proposed in this paper is not as good as Faster R-CNN in the handheld part of the dataset, the accuracy in the boom set, drone set part of the dataset only decreases by 3.2% and 3.9, which suggests that our proposed model can focus on the location of the disease occurrence more efficiently and extract the features of the disease effectively. It suggests that our proposed model can handle the effect of environmental factors on model performance more effectively and has a stronger generalisation ability. We selected some sample images to show the detection effect on the three datasets, as shown in Figure 12.



(**a**)



(**b**)

Figure 12. *Cont.*

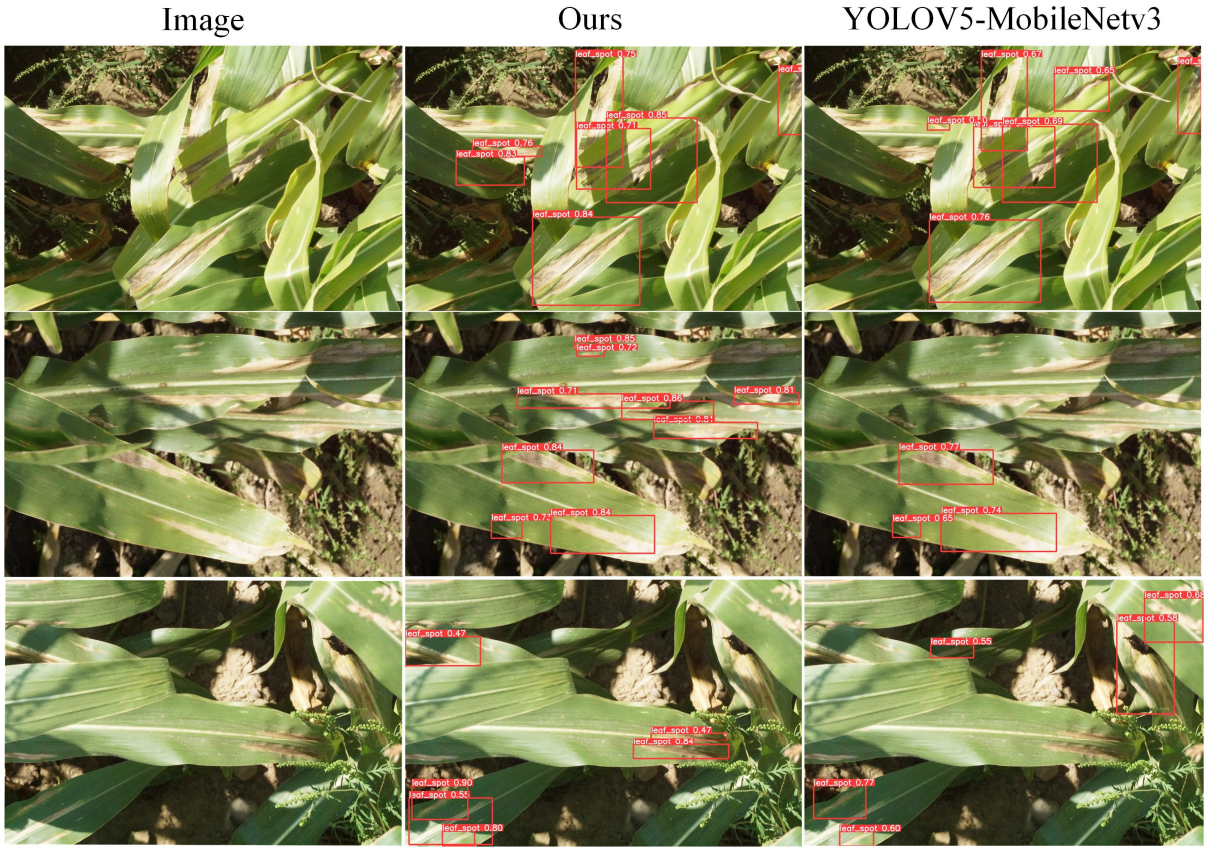


Figure 12. CEMLB-YOLOV5 results for the handheld, drone and boom group datasets. (**a**) Visualization of CEMLB-YOLO detection results on handheld partial datasets. (**b**) Visualization of CEMLB-YOLO detection results on boom partial datasets. (**c**) Visualization of CEMLB-YOLO detection results on drone partial datasets.

As shown in Figure 12, due to the strong illumination of the images taken by the UAV, the original model has a poor detection capability in the drone part of the dataset, creating a missed detection problem. The original model generates false detections in complex scenarios in the boom part of the dataset. However, CEMLB-YOLOv5 detects the drone and boom portion of the NLB dataset significantly better than the original model.

5. Conclusions

This paper proposes the CEMLB-YOLO maize leaf blight detection algorithm based on YOLOv5 to address the challenge of balancing accuracy and detection speed when detecting maize leaf blight in complex scenarios. CIPAM enhances the feature representation of key information more effectively than other attention mechanisms, enabling the model to focus more precisely on the disease's location and ignore irrelevant information in complex environments. MobileBit uses a combination of convolution and transformer architectures to enable the model to efficiently sense long-distance dependencies while at the same time having the inductive bias of convolution, which greatly reduces training time and model complexity compared to standard vision transformers. FRAFM makes full use of the important information in feature maps of different scales and introduces learnable parameters to control the proportion of information in the fusion process of deep and shallow feature maps to achieve more effective cross-scale fusion. The experiments demonstrate that the method proposed in this paper has fewer parameters and lower complexity than other models, which is more suitable for deployment on edge devices and can replace human experts for field identification. However, one limitation of our

current study is the lack of evaluation of the model's robustness under specific weather conditions, such as rain, snow and fog. Our future research will be focused on enhancing the model's ability to detect maize leaf blight under more complex weather conditions. We will remain committed to improving the precision and robustness of our model in the face of environmental variables.

Author Contributions: S.L.; methodology, S.L. and Y.M.; formal analysis, G.F.; investigation, Y.Y. and G.F.; resources, Y.Y. and G.F.; writing—original draft preparation, S.L.; writing—review and editing, Y.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Xinjiang Uygur Autonomous Region Education Department (grant number XJEDU2017M009). Xinjiang University Natural Science Foundation Project (grant number BS180264).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data described in this data note can be freely and openly accessed via a repository on the Open Science Framework (https://osf.io/p67rz/, accessed on 15 June 2023).

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

References

1. Wang, Q.; Chen, Y. Advantages Analysis of Corn Planting in China. *J. Agric. Sci. Tech. China* **2018**, *20*, 1–9.
2. Zhang, M.; Wang, T.; Li, P.; Deng, L.; Zheng, Y.; Yi, S.; Lv, Q.; Sun, R. Surface defect detection of navel orange based on region adaptive brightness correction algorithm. *Sci. Agric. Sin.* **2020**, *53*, 2360–2370.
3. Zhang, F.; Wang, L.; Fu, L.; Tian, Y. Recognition of cucumber leaf disease based on support vector machine. *J. Shenyang Agric. Univ.* **2014**, *45*, 457–462.
4. Lai, J.; Li, S.; Ming, B.; Wang, N.; Wang, K.; Xie, R.; Gao, S. Advances in research on computer-vision diagnosis of crop diseases. *Sci. Agric. Sin.* **2009**, *42*, 1215–1221.
5. Khirade, S.D.; Patil, A. Plant disease detection using image processing. In Proceedings of the 2015 International Conference on Computing Communication Control and Automation, Pune, India, 26–27 February 2015; IEEE: Washington, DC, USA, 2015.
6. Liu, T.; Zhong, X.; Sun, C.; Guo, W.; Chen, Y.; Sun, J. Recognition of rice leaf diseases based on computer vision. *Sci. Agric. Sin.* **2014**, *47*, 664–674.
7. Dang, M.; Meng, Q.; Gu, F.; Gu, B.; Hu, Y. Rapid recognition of potato late blight based on machine vision. *Trans. Chin. Soc. Agric. Eng.* **2020**, *36*, 193–200.
8. Zhao, L.; Hou, F.; Lu, Z.; Zhu, H.; Ding, X. Image recognition of cotton leaf diseases and pests based on transfer learning. *Trans. Chin. Soc. Agric. Eng.* **2020**, *36*, 184–191.
9. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
10. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
11. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
12. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
13. 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 Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14. Springer: Berlin/Heidelberg, Germany, 2016.
14. Jeong, J.; Park, H.; Kwak, N. Enhancement of SSD by concatenating feature maps for object detection. *arXiv* **2017**, arXiv:1705.09587.
15. Li, Z.; Zhou, F. FSSD: Feature fusion single shot multibox detector. *arXiv* **2017**, arXiv:1712.00960.
16. 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, 26 June–1 July 2016.
17. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
18. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
19. Liao, T.; Yang, R.; Zhao, P.; Zhou, W.; He, M.; Li, L. MDAM-DRNet: Dual Channel Residual Network with Multi-Directional Attention Mechanism in Strawberry Leaf Diseases Detection. *Front. Plant Sci.* **2022**, *13*, 869524. [CrossRef] [PubMed]

20. Xie, X.; Ma, Y.; Liu, B.; He, J.; Li, S.; Wang, H. A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. *Front. Plant Sci.* **2020**, *11*, 751. [CrossRef] [PubMed]
21. Liu, J.; Wang, X. Early recognition of tomato gray leaf spot disease based on MobileNetv2-YOLOv3 model. *Plant Methods* **2020**, *16*, 1–16. [CrossRef] [PubMed]
22. Zhao, S.; Liu, J.; Wu, S. Multiple disease detection method for greenhouse-cultivated strawberry based on multiscale feature fusion Faster R_CNN. *Comput. Electron. Agric.* **2022**, *199*, 107176. [CrossRef]
23. Lv, M.; Zhou, G.; He, M.; Chen, A.; Zhang, W.; Hu, Y. Maize leaf disease identification based on feature enhancement and DMS-robust alexnet. *IEEE Access* **2020**, *8*, 57952–57966. [CrossRef]
24. Afzaal, U.; Bhattarai, B.; Pandeya, Y.R.; Lee, J. An instance segmentation model for strawberry diseases based on mask R-CNN. *Sensors* **2021**, *21*, 6565. [CrossRef]
25. Albahli, S.; Nawaz, M. DCNet: DenseNet-77-based CornerNet model for the tomato plant leaf disease detection and classification. *Front. Plant Sci.* **2022**, *13*, 957961. [CrossRef]
26. Jocher, G.; Stoken, A.; Borovec, J.; Changyu, L.; Hogan, A.; Diaconu, L.; Poznanski, J.; Yu, L.; Rai, P.; Ferriday, R. *ultralytics/yolov5: v3.0*; Zenodo: Geneva, Switzerland, 2020.
27. 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, 18–22 June 2023.
28. Jocher, G.; Chaurasia, A.; Qiu, J. YOLO by Ultralytics. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 15 June 2023).
29. Souza, B.J.; Stefenon, S.F.; Singh, G.; Freire, R.Z. Hybrid-YOLO for classification of insulators defects in transmission lines based on UAV. *Int. J. Electr. Power Energy Syst.* **2023**, *148*, 108982. [CrossRef]
30. Stefenon, S.F.; Singh, G.; Souza, B.J.; Freire, R.Z.; Yow, K.C. Optimized hybrid YOLOu-Quasi-ProtoPNet for insulators classification. *IET Gener. Transm. Distrib.* **2023**. [CrossRef]
31. Yao, Y.; Han, L.; Du, C.; Xu, X.; Jiang, X. Traffic sign detection algorithm based on improved YOLOv4-Tiny. *Signal Process. Image Commun.* **2022**, *107*, 116783. [CrossRef]
32. Xu, L.; Dong, S.; Wei, H.; Ren, Q.; Huang, J.; Liu, J. Defect signal intelligent recognition of weld radiographs based on YOLO V5-IMPROVEMENT. *J. Manuf. Process.* **2023**, *99*, 373–381. [CrossRef]
33. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
34. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. Carafe: Content-aware reassembly of features. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
35. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
36. Xiao, T.; Singh, M.; Mintun, E.; Darrell, T.; Dollár, P.; Girshick, R. Early convolutions help transformers see better. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 30392–30400.
37. Zhu, L.; Wang, X.; Ke, Z.; Zhang, W.; Lau, R.W. BiFormer: Vision Transformer with Bi-Level Routing Attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
38. 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.
39. 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.
40. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–16 June 2019.
41. 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, NY, USA, 7–12 February 2020.
42. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef]
43. Wiesner-Hanks, T.; Stewart, E.L.; Kaczmar, N.; DeChant, C.; Wu, H.; Nelson, R.J.; Lipson, H.; Gore, M.A. Image set for deep learning: Field images of maize annotated with disease symptoms. *BMC Res. Notes* **2018**, *11*, 1–3. [CrossRef] [PubMed]
44. Ruder, S. An overview of gradient descent optimization algorithms. *arXiv* **2016**, arXiv:1609.04747.
45. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.
46. Song, B.; Lee, J. Detection of Northern Corn Leaf Blight Disease in Real Environment Using Optimized YOLOv3. In Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 26–29 January 2022; IEEE: New York, NY, USA, 2022.
47. Sun, J.; Yang, Y.; He, X.; Wu, X. Northern maize leaf blight detection under complex field environment based on deep learning. *IEEE Access* **2020**, *8*, 33679–33688. [CrossRef]

48. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
49. 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.