symmetry

MDPI

Article

Critical Information Mining Network: Identifying Crop Diseases in Noisy Environments

Yi Shao [1], Wenzhong Yang [1,2,*], Zhifeng Lu [3,*], Haokun Geng [1] and Danny Chen [1]

[1] School of Computer Science and Technology, Xinjiang University, Urumqi 830017, China; 107552204060@stu.xju.edu.cn (Y.S.); 107552203997@stu.xju.edu.cn (H.G.)
[2] Xinjiang Key Laboratory of Multilingual Information Technology, Xinjiang University, Urumqi 830017, China
[3] School of Information Science and Technology, Xinjiang Teacher's College, Urumqi 830043, China
* Correspondence: yangwenzhong@xju.edu.cn (W.Y.); xjdxsylb@xju.edu.cn (Z.L.)

Abstract: When agricultural experts explore the use of artificial intelligence technology to identify and detect crop diseases, they mainly focus on the research of a stable environment, but ignore the problem of noise in the process of image acquisition in real situations. To solve this problem, we propose an innovative solution called the Critical Information Mining Network (CIMNet). Compared with traditional models, CIMNet has higher recognition accuracy and wider application scenarios. The network has a good effect on crop disease recognition under noisy environments, and can effectively deal with the interference of noise to the recognition effect in actual farmland scenes. Consider that the shape of the leaves can be symmetrical or asymmetrical.First, we introduce the Non-Local Attention Module (Non-Local), which uses a unique self-attention mechanism to fully capture the context information of the image. The module overcomes the limitation of traditional convolutional neural networks that only rely on local features and ignore global features. Global features are particularly important when the image is disturbed by noise. Non-Local improves a more comprehensive visual understanding of crop disease recognition. Secondly, we have innovatively designed a Multi-scale Critical Information Fusion Module (MSCM). The module uses the Key Information Extraction Module (KIB) to dig into the shallow key features in the network deeply. The shallow key features strengthen the feature perception of the model to the noise image through texture and contour information, and then the shallow key features and deep features are fused to enrich the original deep feature information of the network. Finally, we conducted experiments on two public datasets, and the results showed that the accuracy of our model in crop disease identification under a noisy environment was significantly improved. At the same time, our model also showed excellent performance under stable conditions. The results of this study provide favorable support for the improvement of crop production efficiency.

Keywords: CNN; CIMNet; crop disease identification; leaf diagnosis; artificial intelligence (AI)

check for updates

Citation: Shao, Y.; Yang, W.; Lu, Z.; Geng, H.; Chen, D. Critical Information Mining Network: Identifying Crop Diseases in Noisy Environments. *Symmetry* **2024**, *16*, 652. https://doi.org/10.3390/sym16060652

Academic Editors: Theodore E. Simos and Lorentz Jäntschi

Received: 6 April 2024
Revised: 6 May 2024
Accepted: 23 May 2024
Published: 24 May 2024

1. Introduction

Recently, plant diseases have become a major obstacle to the development of agricultural production. Rapid and accurate identification of crop diseases has become an important means to solve the problem of crop yield at this stage [1]. Traditional crop disease identification methods require agricultural producers to have professional agronomic knowledge, and at the same time cost a lot of money and time, often resulting in infected crops not being timely and effectively identified and treated, resulting in huge economic losses [2]. Therefore, to improve the efficiency of agricultural production, more efficient and accurate crop disease identification methods must be adopted.

In recent years, the application of artificial intelligence technology in agriculture has been increasing, providing a large number of methods to solve the problems of agricultural production and making great contributions to the sustainable development of agriculture [3]. The development of the Internet of Things (IoT) fully combines agriculture and the Internet. Supervised and unsupervised machine learning techniques are the main components of intelligent data analysis in the Internet of Things [4]. The pathological manifestations of disease in plants mainly lie in leaves, rhizomes, and fruits [5]. Generally, the type and degree of plant disease are most significant in leaves, so the pathological characteristics of leaves will be the main information source for plant disease identification [6]. In the study on leaves as pathological features, Devi N et al. [7] used a hybrid learning model to identify and classify crop diseases. The hybrid model first uses K-means clustering to detect disease areas in leaves, and then uses convolutional neural networks (CNNs) for feature extraction to achieve the goal of classifying diseases, achieving good recognition results on public datasets. Nandhini et al. [8] combined Inception V3 and Vgg16 convolutional neural networks to solve the time-consuming problem of leaf feature extraction by shallow machine learning architecture and achieved good recognition results on tomato leaf datasets. At the same time, to solve the problem of subtle inter-class differences and large intra-class variation in disease symptoms, Zeng T et al. [9] proposed a GMA-Net network to significantly improve the recognition accuracy of rubber leaf disease by using a multi-scale feature extraction module. M Aggarwal et al. [10] used deep learning and machine learning techniques to identify rice leaf diseases. After analyzing the experimental results, it was found that deep learning technology had better recognition performance. Peng J et al. [11] proposed the RiceDRA-Net model, which uses a 3 × 3 convolution kernel and more dense connections to reduce information loss and achieves high accuracy on rice leaf disease datasets in complex scenarios. M Aggarwal et al. [12] proposed a framework called Federated Transfer Learning (F-TL) for rice leaf disease classification across multiple clients and databases. The framework adopts federated learning and can train shared models on distributed devices or servers without directly transmitting or concentrating raw data.

However, the existing research has neglected that in actual production, equipment such as cameras may be affected by various signal interference, equipment aging, and external factors (such as rain and dust), thus introducing image noise, which will affect the accuracy of plant disease recognition models. Leaves in noise-polluted images lose a high degree of symmetry, which can seriously interfere with the identification of diseases and thus reduce the identification accuracy. This problem is very common in actual production and daily life. Therefore, the recognition and detection of plant diseases not only need to have a high recognition accuracy but also need to improve in their tolerance of image noise. In recent years, whether artificial intelligence technology is model-centric or data-centric has become a widely discussed topic. Hamid et al. [13,14] compared the characteristics of model-centered artificial intelligence and data-centered artificial intelligence, analyzed the limitations of model-centered artificial intelligence, proposed the advantages of data-centered artificial intelligence, and emphasized that we should combine the two, rather than just focusing on one. Only by jointly developing the two can we make the current artificial intelligence more robust and powerful. Ng, A. et al. [15] believed that shifting from big data to high-quality data was an important direction for the development of AI. Jarrahi et al. [16] emphasized the crucial role of data quality in the performance of AI systems and proposed the concept of Data Center AI (DCAI). In this study, we acknowledge the importance of datasets but still focus on convolutional neural networks. We achieve a high recognition rate of crop diseases in noisy environments by deeply optimizing the ResNet network.

Therefore, we propose a solution called Critical Information Mining Network (CIMNet) to solve this problem. Unlike the way V. Gautam et al. [17] used semantic segmentation to extract lesion sites and then trained segmented images for classification, this model takes ResNet [18] as the basic backbone, combining a Non-Local Attention Module (Non-Local) and a Multi-scale Critical Information Fusion Module (MSCM). Non-Local introduces a special self-attention mechanism, which makes up for the lack of global feature utilization in traditional convolutional networks by extracting the long-term dependency between pixels in the image. MSCM uses the Key Information Extraction Module (KIB) to dig deep into the effective parts of shallow features. Then, through the innovative multi-scale architecture, the shallow information and deep information are integrated to enrich the deep feature information of the network and improve the image recognition ability of the model in noisy images. Experiments on two common plant disease datasets show that our CIMNet offers significant advantages over previous networks. It can not only effectively solve the problem caused by image noise, but also improve the performance of plant disease recognition. The main contributions of this study are as follows:

- In this study, we propose a model called CIMNet, which can accurately identify crop diseases in noisy environments.
- We propose an MSCM, which fuses shallow key features with deep features to help the model focus on multi-scale key features and reduce noise interference to image recognition.
- We conducted experiments on two common plant disease datasets in noisy and stable environments, respectively, and compared them with existing methods. The experimental results show that CIMNet can effectively deal with the problems caused by noise.

The structure of the subsequent sections is as follows: Section 2 introduces related work; Section 3 provides an overview of the materials and methods used in this study; Section 4 presents the experimental results and discussion; and Section 5 provides conclusions.

2. Related Work

In recent years, a large number of excellent deep learning methods have been used in the field of plant disease recognition, which can be mainly divided into two categories: convolutional neural network (CNN) and Vision Transformer architecture. Therefore, agricultural experts use these two types of network architecture as the backbone network and then optimize based on the backbone network, so that the network can achieve excellent results in plant disease recognition.

The method based on a convolutional neural network mainly extracts image features through multi-layer convolution and pooling operations and then classifies them through fully connected layers. Since the advent of AlexNet [19] in 2012, convolutional neural networks have been widely used in academia and industry, such as face detection in dangerous situations. In the field of plant disease recognition, Zhe Tang et al. [20] has proposed a lightweight convolutional neural network model to diagnose grape diseases, including black rot, black measles, and leaf blight, which has added the SE mechanism to ShuffleNet and achieved 99.14% accuracy on the Plant Village dataset. De Ocampo et al. [21] proposed a MobileNet that could be deployed on mobile smart devices. This model combined deep convolution and point-to-point convolution and achieved 89% classification accuracy on six randomly selected plant disease datasets. To solve the problem that large-scale architecture is not suitable for mobile devices, Rahman et al. [22] proposed a two-stage small-scale CNN architecture and compared it with advanced memory-efficient CNN architectures such as MobileNet, NasNet mobile, and SqueezeNet. The accuracy of the rice disease dataset was 93.3%. Alfarisy et al. [23] has collected 4511 plant disease images using search engines and enhanced them to develop a diverse dataset, which has been tested with the CaffeNet model and achieved an accuracy of 87%. M Aggarwal et al. [24] reviewed the methods of using machine learning and deep learning techniques for rice leaf disease recognition. Convolutional neural networks, VGG, and other models were used to significantly improve the accuracy of rice leaf disease recognition, and the advantages and disadvantages of various technical routes were summarized. AM Mishra et al. [25] proposed a deep learning-based weed growth estimation method to address the impact of weeds on crops through automated agriculture. Zhang et al. [26], starting from the fact that the color of plant disease leaves is the main basis for disease recognition, used color information and combined three color components to build a three-channel convolutional neural network TCCNN model. Each channel in this model has one of the three color components of RGB images as input and carries out feature extraction through convolution operation and pooling. Finally, the fully connected layer is integrated to obtain the deep disease feature vector. The experimental results show that the model can automatically learn representative features from complex diseased leaf images and effectively identify vegetable diseases, and the accuracy rate is better than the most advanced methods. M Aggarwal et al. [27] collected 551 images of rice leaf diseases, divided them into three categories, and then used various pre-trained deep-learning models for feature extraction. Then, machine learning and ensemble learning classifiers were used for classification. The experimental results showed that EfficientNetV2 achieved an accuracy of 94% on the test set, proving that the combination of feature extraction and classification can effectively identify and classify rice leaf diseases. Kanna et al. [28] used advanced deep learning techniques, especially transfer learning techniques, to perform early disease detection on cauliflower plants. Among multiple models, EfficientNetB1 achieved the highest validation accuracy of 99.9% and the lowest loss of 0.16. The experimental results indicated that advanced deep learning models played a crucial role in automated cauliflower disease detection. Dhaka et al. [29] reviewed the research on using deep convolutional neural networks (DCNNs) for plant leaf disease recognition. They summarized the application cases of deep learning models in various plant leaf disease recognition, demonstrating their accuracy and efficiency. Kundu et al. [30] developed a framework called AIDCC using the Internet of Things (IoT) and interpretable deep transfer learning techniques for detecting and classifying rust and rice blast diseases in pearl millet. The experimental results indicated that the Custom-Net model can effectively extract relevant features.

The method based on Vision Transformer [31] mainly extracts image features through multiple self-attention layers and feedforward neural network layers to distinguish whether plants are diseased. Therefore, the self-attention mechanism is the core of Transformer architecture, and network models utilizing the self-attention mechanism are also considered to be part of Vision Transformer. The disease area of crop leaves is small, and the contrast between the disease area and the background is small, which is easy to confuse. Zeng W et al. [32] paid attention to important areas of images by using the self-attention mechanism and proposed a self-attention convolutional neural network SACNN to extract effective features of plant disease spots to identify crop diseases. SACNN's recognition accuracy on MK-D2 was 2.9% higher than in advanced methods. Wang Y. et al. [33] proposed a deep separable neural network based on Bayesian optimization for disease detection and classification of rice leaf images, realizing the purpose of rapid disease recognition. Lee et al. [34] developed a method based on a recurrent neural network through a large number of experiments to automatically locate the infected area and extract relevant features for disease classification, which solved the problem that the CNN model does not need to pay attention to the visible part of the impact of plant diseases, but can make use of irrelevant background and healthy plant parts for classification. Wang P. et al. [35] proposed a deep learning model to coordinate attention efficiency networks to identify different apple diseases. The model achieved a high recognition rate of 98.9% on the apple disease dataset by integrating a coordinate attention module into EfficientNet.

3. Materials and Methods

3.1. Datasets

In this study, two datasets of potato leaf disease and tomato leaf disease were used to test the performance of the model. Table 1 describes the distribution of the datasets.

Table 1. Dataset image information and statistics.

Class	Name	Number
	Potato disease leaf dataset	
00	Early Blight	1628
01	Late Blight	1414
02	Healthy	1020
	Tomato subset of Plant Village	
00	Bacterial Spot	391
01	Early Blight	392
02	Late Blight	423
03	Leaf Mold	403
04	Septoria Leaf Spot	377
05	Spider Mites	415
06	Target Spot	413
07	Yellow Leaf Curl Virus	393
08	Mosaic Virus	393
09	Healthy	421

3.1.1. Potato Datasets

In this study, we used the potato leaf disease dataset [36], which marked three diseases (early blight, late blight, and healthy) in detail. The dataset image consists of a detected leaf and monochromatic background with a resolution of 256 × 256 × 3 pixels per photo. The public dataset contains 3241 training set images, 416 validation set images, and 405 test set images.

3.1.2. Tomato Datasets

In this study, we used a subset of tomatoes from Plant Village [37] for model evaluation. This subset has detailed labeling of ten disease statuses. They are bacterial spot, early blight, late blight, leaf mold, septoria leaf spot, spider mites, target spot, yellow leaf curl virus, mosaic virus, and healthy. The dataset image consists of detected leaves and similar backgrounds. Each image has a resolution of 256 × 256 × 3 pixels. We manually divided the dataset into 3217 training set pictures, 402 validation set pictures, and 402 test set pictures at a ratio of 8:1:1.

3.2. Image Preprocessing

3.2.1. Image Resizing

To ensure that the dataset images used in this study can be successfully input into various network models during subsequent comparison experiments (networks such as Swin Transformer have specific requirements on the input image size), the bilinear difference method [38] is used to adjust the image size. The bilinear interpolation is calculated as follows: We want to bilinearly differ 2 × 2 pixels to 1 pixel point. Suppose that the coordinates of this one pixel are (a,b), and the four pixels of 2 × 2 are $Q_{11}Q_{12}Q_{21}Q_{22}$. First, we calculate the linear difference twice in the X-axis direction, and the equation is described as follows.

$$f(\mathrm{a},\mathrm{b}_1) \approx \frac{\mathrm{a}-\mathrm{b}}{\mathrm{a}_2-\mathrm{a}_1} f(Q_{11}) + \frac{\mathrm{a}-\mathrm{a}_1}{\mathrm{a}_2-\mathrm{a}_1} f(Q_{21}) \tag{1}$$

$$f(\mathrm{a},\mathrm{b}_2) \approx \frac{\mathrm{a}_2-\mathrm{a}}{\mathrm{a}_2-\mathrm{a}_1} f(Q_{12}) + \frac{\mathrm{a}-\mathrm{a}_1}{\mathrm{a}_2-\mathrm{a}_1} f(Q_{22}) \tag{2}$$

Then, we make a linear difference in the Y-axis and we obtain $f(\mathrm{a},\mathrm{b})$. The equation is described below:

$$f(\mathrm{a},\mathrm{b}) \approx \frac{\mathrm{b}_2-\mathrm{b}}{\mathrm{b}_2-\mathrm{b}_1} f(Q_{12}) + \frac{\mathrm{a}-\mathrm{a}_1}{\mathrm{a}_2-\mathrm{a}_1} f(Q_{22}) \tag{3}$$

The original image size of the dataset used in this study was 256 × 256 × 3. After the above bilinear interpolation operation, the image size was changed to 224 × 224 × 3, and the number of channels was still three channels (red, green, and blue).

3.2.2. Data Augmentation

Agricultural datasets generally have insufficient training samples, so there will be overfitting and underfitting phenomena when training deep learning models. This study uses two methods to avoid these problems. Firstly, the number of training samples is increased by data enhancement techniques. We use two data enhancement methods of random horizontal and vertical flipping to increase the number of images. We set the frequency at which the image is flipped to 0.5. Second, we will save the five models with the lowest validation losses during training, and then select the best model as the test model. Figure 1 shows some of the data-enhanced images.

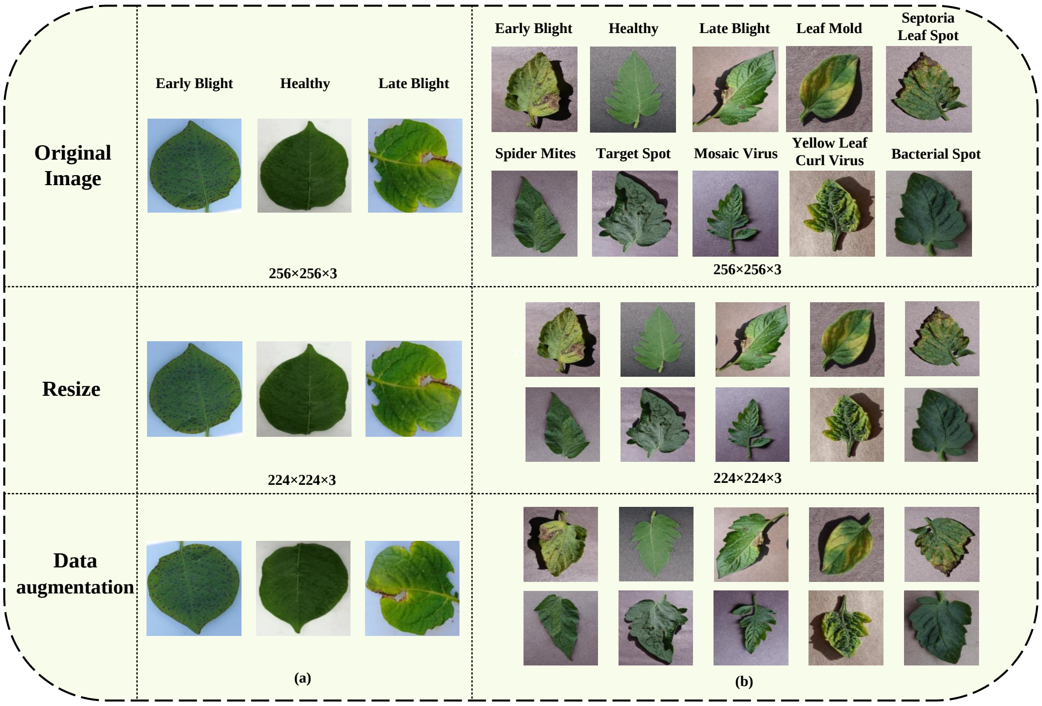


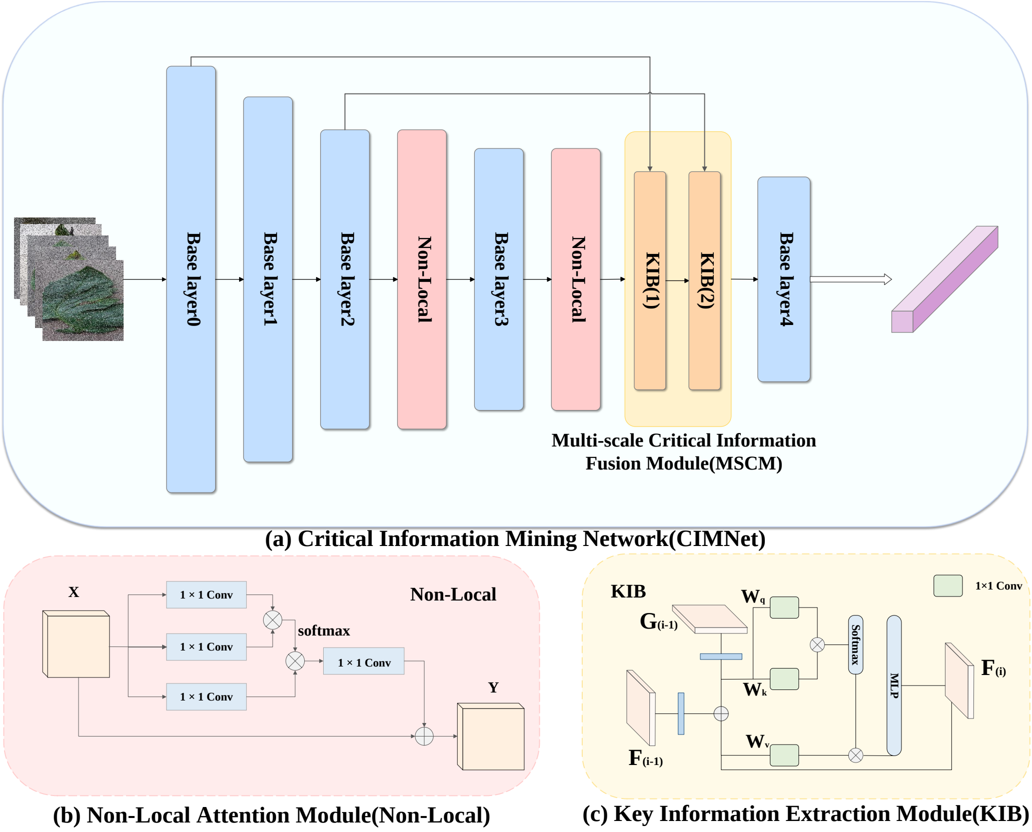
Figure 1. Graphical representation of the datasets and data preprocessing. (**a**) Potato diseased leaves dataset. (**b**) Tomato subset of Plant Village.

3.3. Method

In this section, we will elaborate on the design principles and implementation details of the Critical Information Mining Network (CIMNet), as shown in Figure 2. CIMNet's core innovation is its successful integration of the Non-Local Attention Module (Non-Local) and the Multi-scale Critical Information Fusion Module (MSCM), which significantly improves the feature extraction capability of the ResNet network.

Specifically, to effectively capture global features, we embed Non-Local behind the second and third layers of the ResNet network. This design enables CIMNet to more fully understand the contextual information in the image, thereby increasing the richness and accuracy of the feature representation. Next, we introduced MSCM after the third layer of the model. This module, through a well-designed algorithm, deeply digs the effective features in the shallow features, and integrates with the deep features, effectively reducing the information loss in the process of feature extraction. Unlikw the existing multi-scale feature fusion method, which fuses all shallow features, our multi-scale feature fusion architecture only fuses shallow features of layers 0 and 2. This fusion method can well solve the problem of declining recognition accuracy caused by increasing complexity in small sample datasets. In addition, to alleviate the degradation problem caused by the excessive depth of the model, we used ResNet's residual connection design. This design can not only maintain the depth of the network but also improve the training stability and performance of the model. Table 2 shows the configuration of CIMNet network parameters.

In this chapter, Section 3.3.1 mainly introduces the Non-Local Attention Module (Non-Local), Section 3.3.2 introduces the Multi-scale Critical Information Fusion Module (MSCM), and Section 3.3.3 mainly introduces the loss function used in model training.



(a) Critical Information Mining Network(CIMNet)

(b) Non-Local Attention Module(Non-Local)

(c) Key Information Extraction Module(KIB)

Figure 2. The framework of our proposed method: (**a**) The Critical Information Mining Network consists of the base ResNet, the Non-Local Attention Module (Non-Local), and the Multi-scale Critical Information Fusion Module (MSCM). (**b**) Description of the designed Non-Local. (**c**) The basic components of Key Information Extraction Network (KIB) in MSCM.

Table 2. Parameter setting of CIMNet.

Network Layer	Kernel Size	Number of Steps	Output Dimension
Input	-	-	(b, 3, 224, 224)
Conv2d	7 × 7	1	(b, 64, 112, 112)
Max pool	3 × 3	1	(b, 64, 56, 56)
Conv2d	3 × 3	2	(b, 64, 56, 56)
Conv2d	3 × 3	2	(b, 128, 28, 28)
Non-Local	-	1	(b, 128, 28, 28)
Conv2d	3 × 3	2	(b, 256, 14, 14)
Non-Local	-	1	(b, 256, 14, 14)
KIB	-	1	(b, 256, 14, 14)
KIB	-	1	(b, 256, 14, 14)
Conv2d	3 × 3	2	(b, 512, 7, 7)
Average Pool	7 × 7	1	(b, 512, 1, 1)
FC	-	-	-

3.3.1. Non-Local Attention Module (Non-Local)

Considering the limitations of the original ResNet network in extracting global features, to compensate for this shortcoming, and inspired by the Non-Local mean method [39] and non-local neural network [40], we cleverly integrate Non-Local into the overall network architecture. This improvement enables the network to more fully understand the context information in the image, which significantly improves the capability of feature representation, as shown in Figure 2b. The mathematical expression of the non-local attention feature $\mathbf{y}_i$ is as follows:

$$\mathbf{y}_i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j) \tag{4}$$

where $\mathbf{x}$ represents the preprocessed feature map, and $\mathbf{y}$ is the output map with the same scale as $\mathbf{x}$. $\boldsymbol{i}$ represents the index of the output location where the response needs to be computed, and $\boldsymbol{j}$ traverses all possible location indexes. The $\mathcal{C}(\mathbf{x})$ function is a normalized term, which is realized by dividing the attention weight by element in this study. The calculated $\mathbf{y}_i$ can capture picture context information, that is, global features. Here is the formula for calculating $f\left(\mathbf{x}_i, \mathbf{x}_j\right)$:

$$f(\mathbf{x}_i, \mathbf{x}_j) = \theta(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \tag{5}$$

The function $f\left(\mathbf{x}_i, \mathbf{x}_j\right)$ is used to calculate the similarity between position $\boldsymbol{i}$ and position $\boldsymbol{j}$ in the feature graph $\mathbf{x}$. $\theta(\mathbf{x}_i)^T$ means the 1 × 1 convolution of $\mathbf{x}_i$ and transpose, $\phi(\mathbf{x}_j)$ means the 1 × 1 convolution of $\mathbf{x}_j$ and matrix multiplication of the two gives $f\left(\mathbf{x}_i, \mathbf{x}_j\right)$.

Function $g(\mathbf{x}_j)$ computes the representation of the feature graph at position $\boldsymbol{j}$. In this study, we obtain it by applying 1 × 1 convolution to $\mathbf{x}_j$, and the specific expression is:

$$g(\mathbf{x}_j) = W_g \mathbf{x}_j \tag{6}$$

where W_g denotes the weight of the 1 × 1 convolution.

A complete Non-Local module is described as follows:

$$\mathbf{z}_i = W_z \mathbf{y}_i + \mathbf{x}_i \tag{7}$$

We carry out a 1 × 1 convolution operation on the Non-Local attention $\mathbf{y}_i$ features calculated by Equation (4) to achieve the dimensional alignment of $\mathbf{y}_i$ and $\mathbf{x}_i$ and then introduce the residual join "$+\mathbf{x}_i$" operation. Residual joins allow us to insert Non-Local into any model without destroying the original information. W_z represents the weight of the 1 × 1 convolution.

By introducing Non-Local and defining the corresponding functions f and g, our model can capture the non-local dependencies in the image more effectively, thus improving the robustness and discriminability of the feature representation.

3.3.2. Multi-Scale Critical Information Fusion Module (MSCM)

Multi-scale feature fusion enriches the network's deep feature information by combining shallow features and deep features and improves the classification ability and recognition accuracy of the model. However, we inevitably face the problem that in the fusion process, shallow features may contain both invalid features and effective features, and the effective features are what we need. To solve this problem, we designed the innovative MSCM. MSCM has the following functions:

- Through the Key Information Extraction Module (KIB), the effective parts of shallow features are mined to improve the image recognition ability of the model in complex scenes.
- The shallow information and deep information are integrated to enrich the deep network feature information.

We combined two KIB modules to form a complete MSCM by designing an innovative multi-scale architecture [41]. As shown in Figure 2c, each KIB module has two input features and one output feature. The input feature consists of deep feature $\mathcal{F}_i$ and shallow feature G_i, and the output feature is the key feature information that integrates shallow effective feature and deep feature. We set the output feature of the third layer of the backbone network to be $\mathcal{F}_0$. To make better use of this feature information, we carry out convolution and regularization operations on $\mathcal{F}$ and G and map them to the same feature space. We define ψ as follows:

$$\psi_i^Q = F_{convN}(G_i) \tag{8}$$

$$\psi_i^K = F_{convN}(G_i) \tag{9}$$

$$\psi_i^V = \alpha \cdot F_{convN}(G_i) \oplus (1-\alpha) \cdot F_{convN}(F_i) \tag{10}$$

F_{convN} indicates the dimension reduction operation, and α is set to 0.1. ψ_i^Q, ψ_i^K, and ψ_i^V computations obtain the queries, keys, and inputs needed by the attention mechanism.

KIB explores the effective parts of shallow features and combines them with deep features. The specific expression is as follows:

$$\mathcal{F}_{i+1} = F_{conv}(\mathcal{A}(\psi_i^Q, \psi_i^K, \psi_i^V) + \mathcal{F}_i) \tag{11}$$

$\mathcal{A}$ denotes the attention mechanism, which consists of a combination of a multi-head attention mechanism [42] and a single-layer MLP. Specifically, we use ψ_i^Q, ψ_i^K, and ψ_i^V as the query, key, and input of the attention mechanism, and then determine the shallow effective features after passing through the attention mechanism. "$+\mathcal{F}_i$" stands for residual connectivity. Finally, we use a convolutional layer F_{conv} for the above key features that fuses the shallow valid features and deep features to increase the dimension of the features after the attention mechanism, which makes the subsequent KIB modules use these features smoothly.

3.3.3. Loss Function

In this study, the cross-entropy loss function CrossEntropyLoss is used as a loss function to improve the network accuracy when solving the plant disease classification problem. The cross-entropy loss function is commonly used to assess the gap between the probability distribution predicted by the model and the true probability distribution. Suppose there are two probability distributions p and q. The cross entropy of p through q is:

$$H(p,q) = -\sum_m p(m)logq(m) \tag{12}$$

Assuming that m denotes an event, then $p(m)$ denotes the true probability of event m occurring and $q(m)$ denotes the predicted probability of event m occurring.

In the plant disease classification problem, the output of the model is usually a probability distribution, and we want this probability distribution to be as close as possible to the true labeling distribution. By minimizing the cross-entropy loss function, we can adjust the parameters of the model so that the predicted probability distribution of the model gradually approaches the true label distribution.

4. Experimental Results and Discussion

This section focuses on discussing and analyzing the results of the experiments. Section 4.1 describes our experimental environment, Section 4.2 describes the experimental setup, Section 4.3 describes the comparative experiments on the two datasets, Section 4.4 describes the ablation experiments that we conducted, and in Section 4.5 we discuss and analyze the predicted probability of the output of CIMNet and the other two excellent models in the early blight pictures of potatoes.

4.1. Experimental Environment

In this study, we performed single-card training using an NVIDIA A40. The NVIDIA SMI version is 460.106.00, the driver version is 460.10.6.00, the CUDA version of the graphics card is 11.2, the Torch version is 2.0.1, and the Torchvision version is 0.15.2. We implemented all models in the PyTorch deep learning framework where all models were implemented.

4.2. Experimental Settings

The whole experiment is divided into three parts: the first part is image preprocessing, the second part is model training, and the third part is model performance evaluation. Section 3.2 focuses on the preprocessing part. Section 3.3.3 introduces the loss function in model training. Therefore, in this section, we focus on the model training hyperparameters and performance evaluation part. Our hyperparameters during model training are shown in Table 3.

Table 3. Model training hyperparameters.

Parameter	Value	Description
Epoch	100	Number of complete training with all data from the training set
Batch Size	32	Number of images in a batch
Optimizer	SGD	Minimizing the loss function
Scheduler	MultiStepLR	Adjusting the learning rate of the optimizer
Learning rate	0.0001	The parameters of the MultiStepLR
Loss function	CrossEntropyLoss	Measuring the difference between the value of predicted and true

Model Performance Evaluation

In this study, we utilized three performance metrics, *Top1 Accuracy*, *precision*, and *F1 score*, to assess the effectiveness of the model. True positives (*TPs*) represent the number of leaves correctly classified as infected. True negatives (*TNs*) represent the number of leaves correctly classified as healthy. False positives (*FPs*) represent the number of healthy leaves misclassified as infected. False negatives (*FNs*) represent the number of infected leaves misclassified as healthy.

Top1 Accuracy indicates the accuracy with which the category with the highest probability of prediction among all predictions matches the actual result. It measures the proportion of predictions that the model is most likely to predict correctly given a sample. In this study, we calculate the *Top1 Accuracy* of the proposed model on two datasets and it is calculated as follows:

$$Top1\ Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{13}$$

Precision, also known as checking the accuracy, indicates the proportion of samples with positive predictions that are positive. The formula for calculating precision is as follows:

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

The *F1 score* is a reconciled average of *precision* and *Recall*. It reflects the overall performance of the model, especially when dealing with unbalanced data, and provides a more comprehensive and fair evaluation metric. In this study, we calculated the *F1 score* of the proposed model on two datasets, and the formula for the *F1 score* is as follows:

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{15}$$

The formula for the recall rate to be used in the above equation is as follows:

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

4.3. Comparative Experiments

We compare CIMNet with previous models including AlexNet, SqueezeNet [43], CF-ViT [44], MobileNetV2 [45], and MobileNetV3 [46], which are based on convolutional neural networks. Pooling-based vision Transformer (PiT) [47], Transformer-iN-Transformer (TNT) [48], LeViT [49], CoAtNet [50], Vision Transformer (ViT), and Swin Transformer [51] are also used. We also compared CvT [52] and Mobile-Former [53] models based on CNN+Transformer architecture.

In this section, the experimental results are analyzed in three parts. The first part analyzes the disease recognition accuracy of the CIMNet model in a noisy environment, the second part analyzes the model performance in a stable environment, and the third part conducts a case study. To simulate the signal interference and noise problems that may be encountered during image acquisition, we injected 10%, 20%, and 30% salt noise into the test images, as shown in Figure 3. In this study, the model performance is measured mainly by Top1 Accuracy in the noise test and by Top1 Accuracy and F1 score in the stable environment.

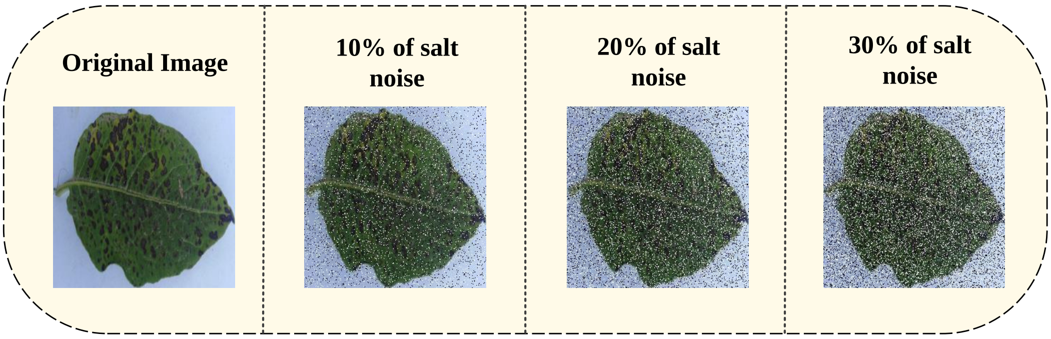


Figure 3. Adding three proportions of salt noise to a picture of potato early blight separately.

4.3.1. Noise Environment Experiment

To verify the higher accuracy of our model compared to other models for recognition in noisy environments, we tested CIMNet and other models in 10%, 20%, and 30% noise environments, and the experimental results are shown in Table 4.

Our CIMNet demonstrates superior performance on the potato dataset compared to traditional convolutional neural network (CNN) models. Specifically, CIMNet's Top1 Accuracy improves by 0.061, 0.088, and 0.091 at 10%, 20%, and 30% noise levels, respectively. On the tomato dataset, CIMNet also performs well, with accuracies improving by 0.059, 0.109, and 0.056, respectively. Of particular interest is that, in the environment containing only 10% noise, CIMNet achieves recognition accuracies of 0.965 and 0.754 on the two datasets, respectively, and these scores even exceed the test accuracies of some models in stable environments. In addition, when the noise percentage increases to 30%, CIMNet achieves an accuracy improvement of 0.091 compared to the best-performing AlexNet model. This significant improvement demonstrates the importance of global feature extraction and utilization for improving recognition accuracy in noisy environments.

Comparison experiments are conducted with the model of Transformer architecture. On the potato dataset, CIMNet improves its Top1 Accuracy by 0.071, 0.066, and 0.054 in 10%, 20%, and 30% noise environments, respectively. On the tomato dataset, the accuracy also improves by 0.009, 0.084, and 0.042, respectively. It is worth noting that, although the Transformer model can, through the self-attention mechanism, extract global features, CIMNet still improves by 0.042 over the best-performing CoAtNet in a 30% noisy environment. This result further demonstrates the robustness and effectiveness of our model in noisy environments.

Comparing CIMNet and two models based on CNN+Transformer architecture, we can see that the recognition accuracy of CvT is significantly different from CIMNet in three different proportions of noisy environments, and even reaches a precision difference of 0.347 in the potato dataset with a 30% noise environment. We believe this may be due to the CvT model dividing images into nonoverlapping image blocks and then performing feature extraction, which is suitable for high-precision images. The two datasets used in this study have lower resolutions. The recognition rate of the Mobile-Former model on two datasets decreased less compared to CIMNet, with a maximum of only 0.092. This fully demonstrates the importance of the self-attention mechanism. The reason why CIMNet has higher accuracy than Mobile-Former may be due to our proposed MSCM architecture, which can fully utilize shallow key features to enrich deep features, thereby achieving the goal of optimizing feature extraction. From the comparison of these two models, it can be found that the MSCM architecture in CIMNet has unique feature extraction capabilities.

As shown in Figure 4, we can see that under different proportions of noisy environments, our model has a significant improvement compared to AlexNet, which has the highest accuracy among CNN methods, and CoAtNet, which has the highest recognition accuracy among Transformer architectures.

We can see from the ROC curve in Figure 5 that in a 20% noise environment, CIMNet performs well in the overall classification of the two types of diseases in the potato dataset. The classification of leaf mold and mosaic virus diseases in the tomato dataset is good, while the classification of the other categories is poor. Overall, the classification effect is good.

From the confusion matrix (Figures 6 and 7), it can be seen that the recognition accuracy of CIMNet varies for different diseases on the two datasets. In the potato dataset, the recognition rates of early blight and late blight in a stable environment can reach 0.979 and 0.987. However, as noise increases, the recognition rates of both diseases in a 30% environment drop to 0.889 and 0.765, with a significant decrease in the recognition rate of late blight. This indicates that our model has poor recognition performance for late blight in high-noise environments. In the tomato dataset, our model has poor recognition performance for spider mites and leaf mold diseases in a 10% noise environment, while other diseases have good recognition performance. However, as noise increases, the recognition performance of other diseases decreases. We believe this may be due to the similarity of image features of tomato diseases. Overall, CIMNet has high recognition accuracy for different diseases.

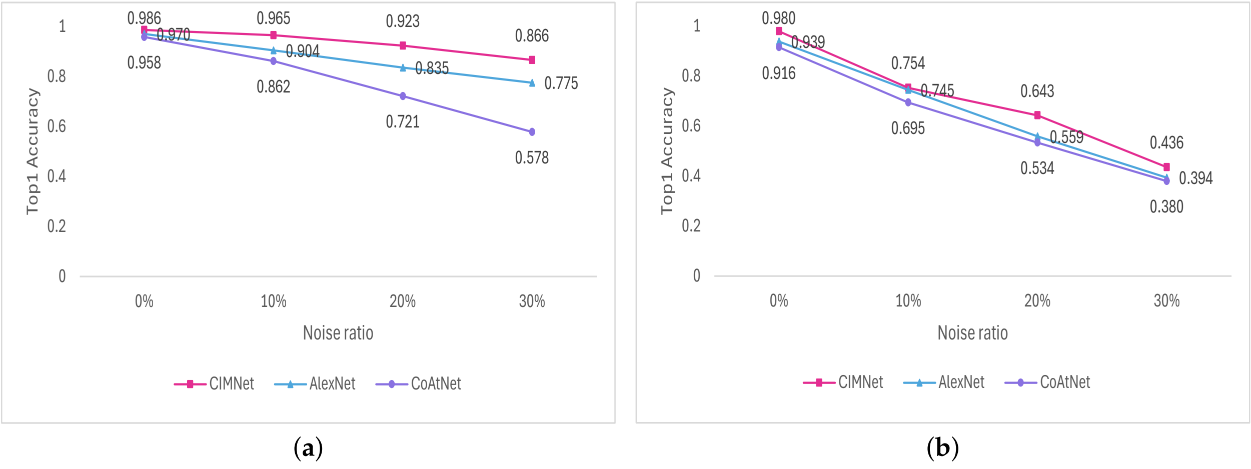
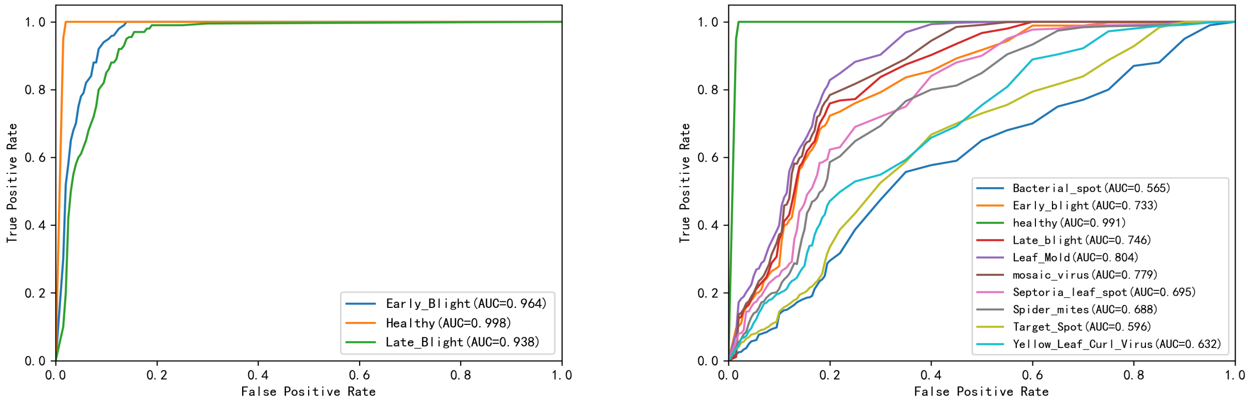


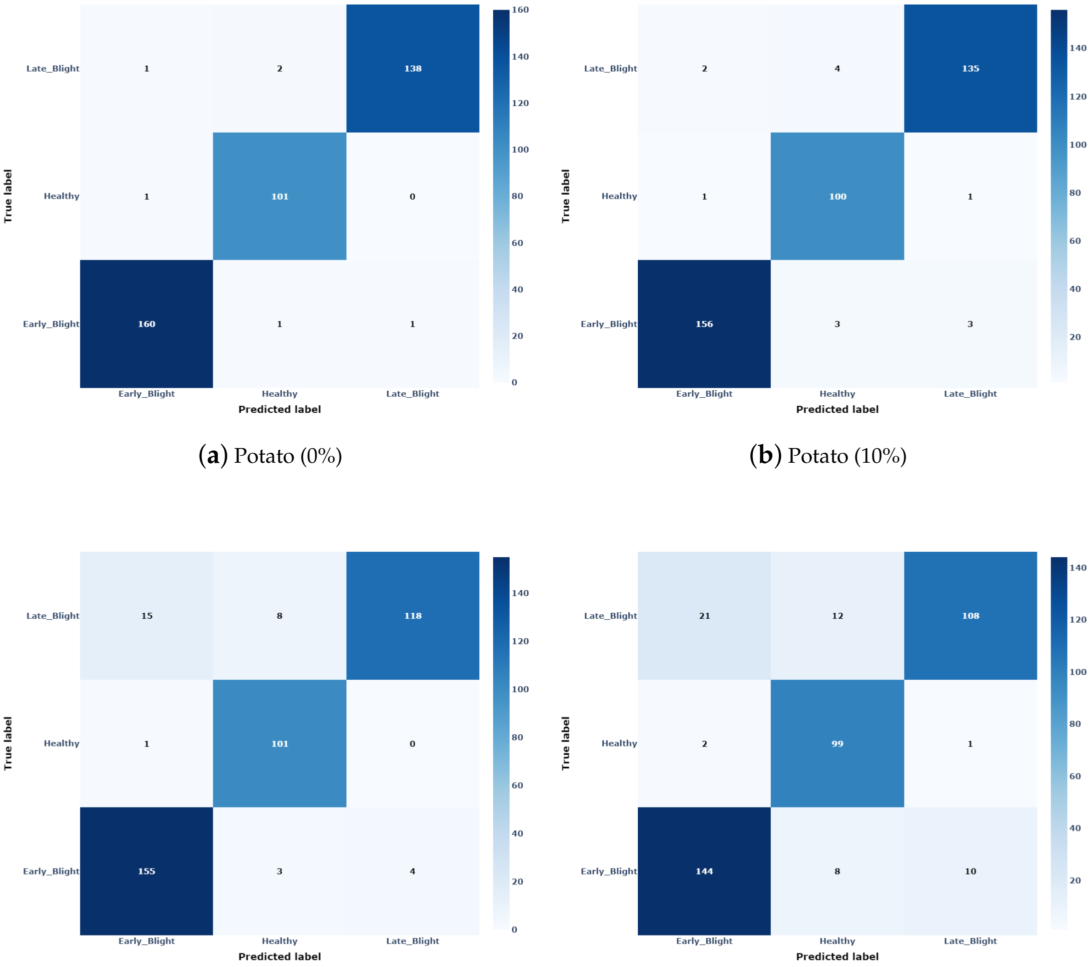
Figure 4. Experimental comparison: CIMNet is our proposed model, AlexNet is the best-performing model in CNN, and CoAtNet is the best-performing model in the Transformer architecture. (**a**) Experimental results on the potato dataset. (**b**) Experimental results on the tomato dataset.



(**a**) ROC: Potato

(**b**) ROC: Tomato

Figure 5. ROC curves of potato dataset and tomato dataset under 20% noise environment. (**a**) ROC curve of the potato dataset. (**b**) ROC curve of tomato dataset.



(**a**) Potato (0%)

(**b**) Potato (10%)

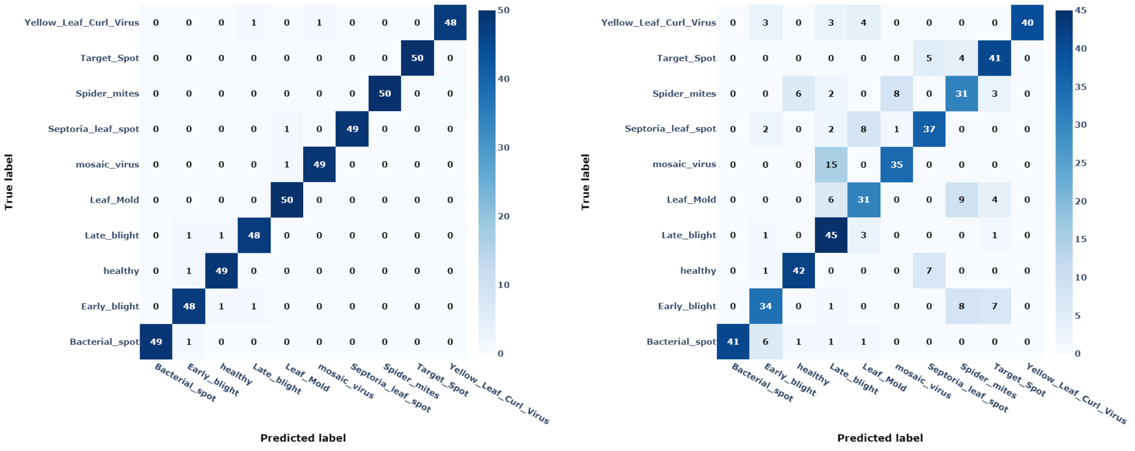
(**c**) Potato (20%)

(**d**) Potato (30%)

Figure 6. The confusion matrix of the potato dataset. (**a**) Stable environment. (**b**) 10% noise environment. (**c**) 20% noise environment. (**d**) 30% noise environment.

Table 4. Experimental results for the potato diseased leaf dataset and the tomato diseased leaf dataset in a noisy environment. Results for the potato diseased leaf dataset are from [54] (except for CF-ViT, CvT, and Mobile-Former models). Marked in blue are sub-optimal. The optimal experimental results are marked in bold.

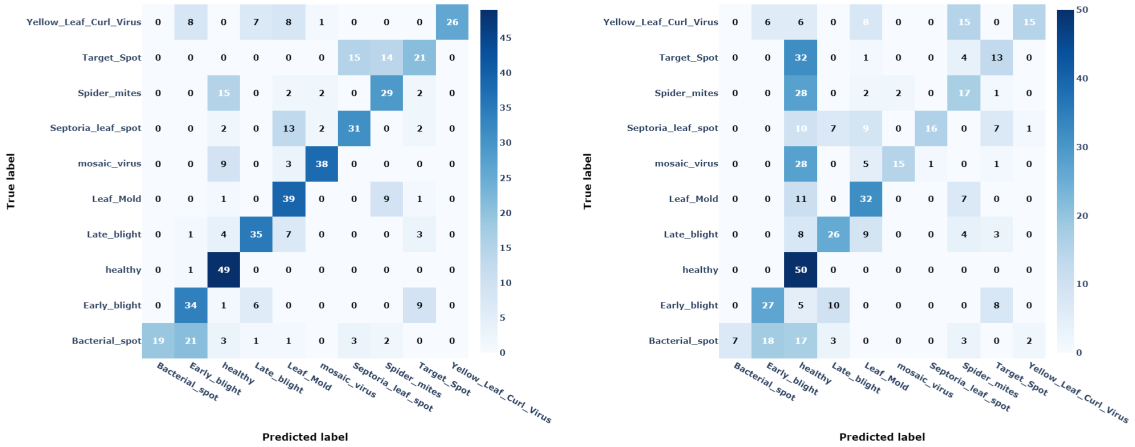
Models	Potato Top1-Acc			Tomato Top1-Acc		
	10%	20%	30%	10%	20%	30%
CNNS						
AlexNet	0.904	0.835	0.775	0.695	0.534	0.380
SqueezeNet1_0	0.667	0.491	0.430	0.411	0.231	0.138
SqueezeNet1_1	0.506	0.494	0.491	0.298	0.211	0.167
MobileNetV2	0.383	0.257	0.252	0.254	0.157	0.118
MobileNetV3_small	0.412	0.400	0.395	0.331	0.229	0.175
Transformer based models						
PiT-s	0.889	0.844	0.793	0.743	0.545	0.372
PiT-b	0.886	0.857	0.812	0.610	0.398	0.312
LeViT-128	0.640	0.447	0.435	0.223	0.160	0.145
LeViT-256	0.432	0.407	0.400	0.374	0.244	0.131
ViT	0.768	0.647	0.509	0.635	0.441	0.325
CF-ViT (η = 1.0)	0.874	0.811	0.761	0.687	0.496	0.330
TNT	0.894	0.830	0.778	0.677	0.511	0.365
CoAtNet	0.862	0.721	0.578	0.745	0.559	0.394
Swin Transformer-small	0.810	0.588	0.477	0.534	0.417	0.355
Swin Transformer-base	0.869	0.768	0.654	0.622	0.405	0.253
CNN+Transformer						
CvT	0.796	0.631	0.519	0.633	0.431	0.249
Mobile-Former	0.895	0.831	0.798	0.712	0.511	0.381
Our model						
CIMNet	**0.965**	**0.923**	**0.866**	**0.754**	**0.643**	**0.436**



(**a**) Tomato (0%)

(**b**) Tomato (10%)

Figure 7. *Cont.*



(**c**) Tomato (20%)

(**d**) Tomato (30%)

Figure 7. The confusion matrix of the tomato dataset. (**a**) Stable environment. (**b**) 10% noise environment. (**c**) 20% noise environment. (**d**) 30% noise environment.

4.3.2. Stable Environment Experiment

To verify that our model has excellent recognition accuracy even in stable environments, we compared CIMNet with previous methods, as shown in Table 5.

Table 5. Experimental results for the potato diseased leaf dataset and the tomato diseased leaf dataset in a stabilized environment. Results for the potato-diseased leaf dataset were obtained from [37]. Marked in blue are sub-optimal. The optimal experimental results are marked in bold.

Models	**Complexity (Gmac)**	**Parameters (M)**	**Top1-Acc Potato**	**F1 Score Potato**	**Top1-Acc Tomato**	**F1 Score Tomato**
			CNNS			
AlexNet	0.71	57.16	0.958	0.957	0.939	0.938
SqueezeNet1_0	1.47	1.25	0.965	0.966	0.930	0.930
SqueezeNet1_1	0.27	1.24	0.973	0.972	0.939	0.939
MobileNetV2	0.32	3.4	0.978	0.977	0.971	0.971
MobileNetV3_small	0.16	1.77	0.973	0.972	0.977	0.977
			Transformer based models			
PiT-s	2.42	23.46	0.916	0.915	0.891	0.891
PiT-b	10.54	73.76	0.894	0.894	0.754	0.753
LeViT-128	0.37	8.46	0.911	0.910	0.978	0.978
LeViT-256	1.05	17.89	0.926	0.925	0.970	0.970
ViT	1.36	1.36	0.852	0.851	0.766	0.766
CF-ViT (η = 1.0)	4.00	29.8	0.971	0.971	0.914	0.914
TNT	13.4	65.24	0.926	0.925	0.758	0.758
CoAtNet	11.51	10.5	0.970	0.969	0.916	0.916
Swin Transformer-small	8.51	48.75	0.924	0.921	0.795	0.795
Swin Transformer-base	15.13	86.62	0.867	0.869	0.850	0.850
			CNN+Transformer			
CvT	4.53	19.98	0.932	0.931	0.855	0.855
Mobile-Former	0.34	11.42	0.974	0.974	0.969	0.969
			Our model			
CIMNet	2.18	12.64	**0.986**	**0.986**	**0.980**	**0.980**

Comparing our model to mainstream methods, on the potato dataset, our model improves by 0.008 on Top1 Accuracy and F1 score. On the tomato dataset, our model also achieves some performance improvement of 0.002. It is worth noting that compared to the huge improvement in recognition accuracy in noisy environments, the performance of our model improves less in stable environments, and we believe that this is mainly because the CIMNet model's advantage over other models mainly lies in feature extraction in noisy environments, i.e., the Non-Local Attention Module and the Multi-scale Critical Information Fusion Module focus more on the contextual dependency of the images. It is undeniable that CIMNet still achieves the best model performance in stable environments.

Our model has increased in complexity compared to the CNN architecture. Compared to MobileNetV2, which has the best recognition performance, it is 1.86 higher, but the recognition accuracy is 0.8% higher, which is a huge improvement in a stable environment. Comparing the models of CIMNet and Transformer architectures results in lower complexity. Even though models like LeViT have slightly lower complexity than CIMNet, their recognition accuracy does decrease. Compared with the CNN+Transformer architecture model, CIMNet has lower complexity and higher recognition accuracy. From the above analysis, it can be concluded that CIMNet is the optimal model in balancing complexity and recognition accuracy.

4.4. Ablation Experiment

To verify the impact of our Non-Local and MSCM on the performance, we performed ablation experiments on the potato and tomato datasets using the same random number of seeds and the same loss function CrossEntropyloss, and the results are shown in Table 6. We used the Top1 Accuracy as the test criterion and the noise environment using a 20% scale. Here, "w/o Non-Local" means remove Non-Local module; at this time this is our model for adding a multi-scale key information fusion module to the original ResNet for feature extraction. "w/o MSCM" means removing the MSCM; at this time it becomes adding a Non-Local module to ResNet for feature extraction. "w/o Non-Local + MSCM" means to remove the Non-Local and MSCM, which means using the original ResNet for feature extraction.

Table 6. Model performance when different components are turned off.

Model	**Noisy Environment**		**Stabilized Environment**	
	Potato	**Tomato**	**Potato**	**Tomato**
CIMNet	0.923	0.643	0.986	0.980
w/o Non-Local	0.893	0.481	0.977	0.973
w/o MSCM	0.835	0.358	0.981	0.977
w/o Non-Local + MSCM	0.793	0.275	0.975	0.972

From Table 6, we can observe that in noisy environments, when the Non-Local module is removed, there is a significant decrease of 0.03 and 0.162 in the Top1 Accuracy. Under stable environmental conditions, there are also decreases of 0.009 and 0.007 in Top1 Accuracy. This data change indicates that the Non-Local module, through its unique self-attention mechanism, can effectively extract global attention, which in turn helps the network to better utilize context dependence for object recognition in noisy environments. Even in stable environments, the global feature extraction of the Non-Local module gives a large performance improvement in image recognition.

Through the experimental results in Table 6, we can find that in the noisy environment, the decrease of Top1 Accuracy after removing the MSCM is on the large side, specifically 0.088 and 0.285, which indicates that the innovative method of fusing the shallow effective features and the deeper features is of great significance for image recognition in noisy environments. Meanwhile, there is also a reduction of 0.005 and 0.003 in Top1 Accuracy in a stable environment. This indicates that the innovative Multi-scale Critical Information Fusion Module contributes prominently to image recognition in noisy environments, and the module is also effective in stable environments.

After removing Non-Local + MSCM, the CIMNet model becomes raw ResNet at this point, and there is a significant decrease in accuracy in both noisy and stable environments. In the noisy environment especially, it shows steep decreases of 0.13 and 0.368. This indicates that our Non-Local and MSCM are useful for crop disease recognition in both noisy and stable environments, especially in noisy environments.

4.5. Model Performance Analysis

We selected an early blight picture from the potato disease dataset (shown in Figure 3). The image was fed into three networks, namely CIMNet, AlexNet, and CoAtNet, and then the early blight prediction probability values output by Softmax were analyzed to verify the performance of our model. Among them, AlexNet is the model with the highest recognition accuracy in the CNN method and CoAtNet is the model with the highest recognition accuracy in Transformer architecture.

We can see from Figure 8 that in the stable environment, the predicted probability value of CIMNet output is 0.96, and the predicted probability values of AlexNet and CoAtNet are 0.92 and 0.87. All three classify the image as early blight, and we can see that the classification is correct, and the predicted probability values of the output are very high. This shows that all three models have high recognition performance in a stable environment. In a 10% noise environment, the predicted probability value of CIMNet output is 0.88, which classifies the image as early blight. The predicted probability values of AlexNet and CoAtNet models are 0.66 and 0.63, and both of them classify the image as early blight. At this point, all three models have the same classification results and are correct, but the predicted probability value of early blight for CIMNet is much higher than that of the other two models, by 0.22 and 0.25, respectively. In a 20% noise environment, CIMNet outputs a predicted probability value of 0.91, which classifies the image as early blight. The AlexNet and CoAtNet models output a predicted probability value of 0.35 and 0.74. AlexNet considers the leaf healthy and outputs a probability value of 0.51, thus classifying it incorrectly. CoAtNet classifies it correctly. This shows that AlexNet does not recognize noise-contaminated images well without making full use of global features, and incorrectly classified early blight leaves as healthy in this test due to the interference of salt noise. CIMNet and CoAtNet utilize global features to give excellent recognition results in the case of moderate noise contamination. In the test results in a 30% noise environment we can see that CIMNet classifies the leaf correctly with an output predictive probability value of 0.69. CoAtNet and AlexNet classify the leaf incorrectly. CoAtNet and AlexNet incorrectly recognize the leaf as a healthy leaf with an output predictive probability value of 0.55 and 0.63. The results show that our Multi-scale Critical Information Fusion Module performs well in a heavily noise-polluted environment.

Through the above analysis, we know that CIMNet has obvious advantages over the CNN and Transformer architecture models in noisy environments, and improves the recognition of crop diseases in complex environments significantly.

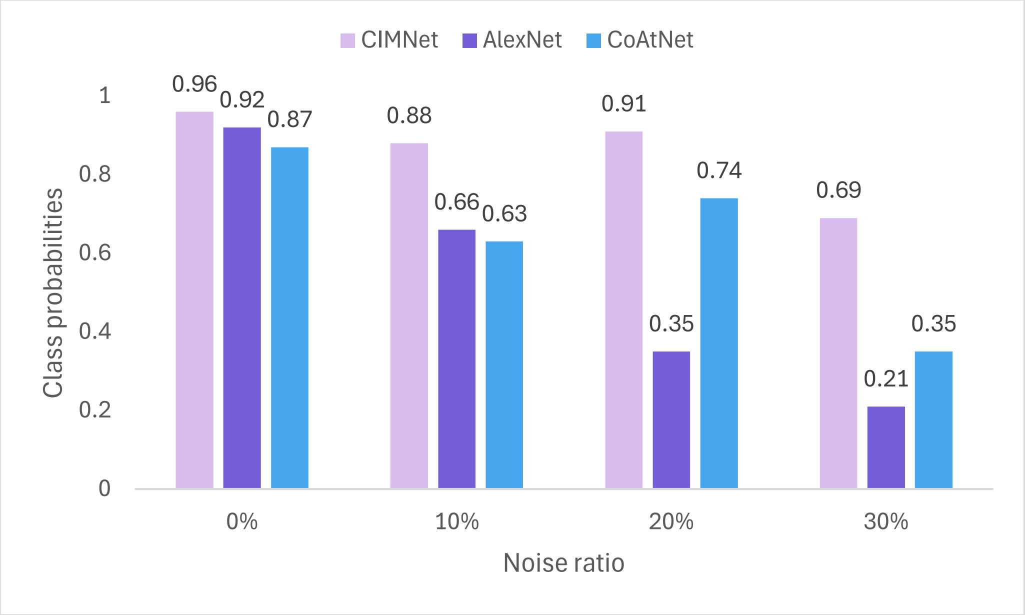


Figure 8. Predicted probability values of early blight for CIMNet, AlexNet, and CoAtNet.

5. Conclusions

In this paper, we address the problem that agricultural experts focus on crop disease recognition in stable environments and neglect research in noisy environments. A Critical Information Mining Model (CIMNet) is proposed. CIMNet introduces a Non-Local Attention Module (Non-Local) to better capture the image context information, which makes up for the shortcoming of under-utilizing the global features of traditional convolutional neural networks. An innovative Multi-scale Critical Information Fusion Module (MSCM) is used to fuse shallow critical features with deeper features, allowing the model to capture both the detail and texture information contained in the low-level features as well as the semantic and contextual information in the high-level features, which helps the network to better recognize and detect in noisy environments. The experimental results show that the CIMNet network can achieve a maximum recognition accuracy of 0.965 in a 10% noise environment, which is significantly better than other comparative models. In a stable environment, the recognition rate reached 0.986. CIMNet can provide technical support for crop disease identification. There are still many areas worth further research in the identification of crop diseases such as combining multi-source data (such as hyperspectral images, infrared images, etc.) to provide more comprehensive disease information. On can also utilize multimodal learning techniques to combine image data with other types of data (such as meteorological data, soil data, etc.) for model training. Finally, we can reduce model complexity while maintaining robustness, making it easier to integrate into daily life. These issues are still unresolved in crop disease identification work and will be the focus of further research.

Author Contributions: Conceptualization, Y.S.; methodology, Y.S.; validation, Y.S.; writing—original draft preparation, Y.S.; writing—review and editing, Y.S., W.Y. and D.C.; supervision, D.C., Z.L. and H.G.; funding acquisition, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was a research achievement supported by the National Key R&D Program of China (Grant No. 2022ZD0115802), the Key Research and Development Program of the Autonomous Region (Grant No. 2022B01008), the National Natural Science Foundation of China (Grant No. 62262065), and the Tianshan Science and Technology Innovation Leading talent Project of the Autonomous Region (Grant No. 2022TSYCLJ0037).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

References

1. Yuan, Y.; Chen, L.; Wu, H.; Li, L. Advanced agricultural disease image recognition technologies: A review. *Inf. Process. Agric.* **2022**, *9*, 48–59. [CrossRef]
2. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 215232. [CrossRef]
3. Arnal Barbedo, J.G. Digital image processing techniques for detecting, quantifying and classifying plant diseases. *SpringerPlus* **2013**, *2*, 660. [CrossRef]
4. Alsharif, M.H.; Kelechi, A.H.; Yahya, K.; Chaudhry, S.A. Machine learning algorithms for smart data analysis in internet of things environment: Taxonomies and research trends. *Symmetry* **2020**, *12*, 88. [CrossRef]
5. Li, L.; Zhang, S.; Wang, B. Plant disease detection and classification by deep learning—A review. *IEEE Access* **2021**, *9*, 56683–56698. [CrossRef]
6. Ebrahimi, M.; Khoshtaghaza, M.H.; Minaei, S.; Jamshidi, B. Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* **2017**, *137*, 52–58. [CrossRef]
7. Kannadasan, R.; Alsharif, M.H.; Jahid, A.; Khan, M.A. Categorizing diseases from leaf images using a hybrid learning model. *Symmetry* **2021**, *13*, 2073.
8. Nandhini, S.; Ashokkumar, K. Improved crossover based monarch butterfly optimization for tomato leaf disease classification using convolutional neural network. *Multimed. Tools Appl.* **2021**, *80*, 18583–18610. [CrossRef]
9. Zeng, T.; Li, C.; Fu, W. Rubber leaf disease recognition based on improved deep convolutional neural networks with a cross-scale attention mechanism. *Front. Plant Sci.* **2022**, *13*, 829479. [CrossRef]
10. Aggarwal, M.; Khullar, V.; Goyal, N. Exploring classification of rice leaf diseases using machine learning and deep learning. In Proceedings of the 2023 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM), Uttar Pradesh, India, 22–24 February 2023; IEEE: New York, NY, USA, 2023; pp. 1–6.
11. Peng, J.; Wang, Y.; Jiang, P.; Zhang, R.; Chen, H. RiceDRA-net: Precise identification of rice leaf diseases with complex backgrounds using a res-attention mechanism. *Appl. Sci.* **2023**, *13*, 4928. [CrossRef]
12. Aggarwal, M.; Khullar, V.; Goyal, N.; Gautam, R.; Alblehai, F.; Elghatwary, M.; Singh, A. Federated transfer learning for rice-leaf disease classification across multiclient cross-silo datasets. *Agronomy* **2023**, *13*, 2483. [CrossRef]
13. Hamid, O.H. From model-centric to data-centric AI: A paradigm shift or rather a complementary approach? In Proceedings of the 2022 8th International Conference on Information Technology Trends (ITT), Dubai, United Arab Emirates, 25–26 May 2022; IEEE: New York, NY, USA, 2022; pp. 196–199.
14. Hamid, O.H. Data-centric and model-centric AI: Twin drivers of compact and robust industry 4.0 solutions. *Appl. Sci.* **2023**, *13*, 2753. [CrossRef]
15. Ng, A. MLOps: From Model-Centric to Data-Centric AI. AI. 2021. Available online: https://www.deeplearning.ai/wp-content/uploads/2021/06/MLOps-From-Model-centric-to-Data-centric-AI.pdf (accessed on 5 March 2024).
16. Jarrahi, M.H.; Memariani, A.; Guha, S. The principles of data-centric AI (DCAI). *arXiv* **2022**, arXiv:2211.14611.
17. Gautam, V.; Trivedi, N.K.; Singh, A.; Mohamed, H.G.; Noya, I.D.; Kaur, P.; Goyal, N. A transfer learning-based artificial intelligence model for leaf disease assessment. *Sustainability* **2022**, *14*, 13610. [CrossRef]
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
20. Tang, Z.; Yang, J.; Li, Z.; Qi, F. Grape disease image classification based on lightweight convolution neural networks and channelwise attention. *Comput. Electron. Agric.* **2020**, *178*, 105735. [CrossRef]
21. De Ocampo, A.L.P.; Dadios, E.P. Mobile platform implementation of lightweight neural network model for plant disease detection and recognition. In Proceedings of the 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Baguio City, Philippines, 29 November–2 December 2018; IEEE: New York, NY, USA, 2018; pp. 1–4.
22. Rahman, C.R.; Arko, P.S.; Ali, M.E.; Khan, M.A.I.; Apon, S.H.; Nowrin, F.; Wasif, A. Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosyst. Eng.* **2020**, *194*, 112–120. [CrossRef]
23. Alfarisy, A.A.; Chen, Q.; Guo, M. Deep learning based classification for paddy pests & diseases recognition. In Proceedings of the 2018 International Conference on Mathematics and Artificial Intelligence, Chengdu, China, 20–22 April 2018; pp. 21–25.
24. Aggarwal, M.; Khullar, V.; Goyal, N. Contemporary and futuristic intelligent technologies for rice leaf disease detection. In Proceedings of the 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 13–14 October 2022; IEEE: New York, NY, USA, 2022; pp. 1–6.
25. Mishra, A.M.; Harnal, S.; Mohiuddin, K.; Gautam, V.; Nasr, O.A.; Goyal, N.; Alwetaishi, M.; Singh, A. A Deep Learning-Based Novel Approach for Weed Growth Estimation. *Intell. Autom. Soft Comput.* **2022**, *31*, 1157–1173. [CrossRef]

26. Zhang, S.; Huang, W.; Zhang, C. Three-channel convolutional neural networks for vegetable leaf disease recognition. *Cogn. Syst. Res.* **2019**, *53*, 31–41. [CrossRef]
27. Aggarwal, M.; Khullar, V.; Goyal, N.; Singh, A.; Tolba, A.; Thompson, E.B.; Kumar, S. Pre-trained deep neural network-based features selection supported machine learning for rice leaf disease classification. *Agriculture* **2023**, *13*, 936. [CrossRef]
28. Kanna, G.P.; Kumar, S.J.; Kumar, Y.; Changela, A.; Woźniak, M.; Shafi, J.; Ijaz, M.F. Advanced deep learning techniques for early disease prediction in cauliflower plants. *Sci. Rep.* **2023**, *13*, 18475. [CrossRef] [PubMed]
29. Dhaka, V.S.; Meena, S.V.; Rani, G.; Sinwar, D.; Ijaz, M.F.; Woźniak, M. A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. *Sensors* **2021**, *21*, 4749. [CrossRef] [PubMed]
30. Kundu, N.; Rani, G.; Dhaka, V.S.; Gupta, K.; Nayak, S.C.; Verma, S.; Ijaz, M.F.; Woźniak, M. IoT and interpretable machine learning based framework for disease prediction in pearl millet. *Sensors* **2021**, *21*, 5386. [CrossRef] [PubMed]
31. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16×16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
32. Zeng, W.; Li, M. Crop leaf disease recognition based on Self-Attention convolutional neural network. *Comput. Electron. Agric.* **2020**, *172*, 105341. [CrossRef]
33. Wang, Y.; Wang, H.; Peng, Z. Rice diseases detection and classification using attention based neural network and bayesian optimization. *Expert Syst. Appl.* **2021**, *178*, 114770. [CrossRef]
34. Lee, S.H.; Goëau, H.; Bonnet, P.; Joly, A. Attention-based recurrent neural network for plant disease classification. *Front. Plant Sci.* **2020**, *11*, 601250. [CrossRef] [PubMed]
35. Wang, P.; Niu, T.; Mao, Y.; Zhang, Z.; Liu, B.; He, D. Identification of apple leaf diseases by improved deep convolutional neural networks with an attention mechanism. *Front. Plant Sci.* **2021**, *12*, 723294. [CrossRef] [PubMed]
36. Rashid, J.; Khan, I.; Ali, G.; Almotiri, S.H.; AlGhamdi, M.A.; Masood, K. Multi-level deep learning model for potato leaf disease recognition. *Electronics* **2021**, *10*, 2064. [CrossRef]
37. Hughes, D.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.
38. Bovik, A.C. Basic gray level image processing. In *The Essential Guide to Image Processing*; Elsevier: Amsterdam, The Netherlands, 2009; pp. 43–68.
39. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–26 June 2005; IEEE: New York, NY, USA, 2005; Volume 2, pp. 60–65.
40. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
41. Gao, S.H.; Cheng, M.M.; Zhao, K.; Zhang, X.Y.; Yang, M.H.; Torr, P. Res2net: A new multi-scale backbone architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 652–662. [CrossRef] [PubMed]
42. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
43. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
44. Chen, M.; Lin, M.; Li, K.; Shen, Y.; Wu, Y.; Chao, F.; Ji, R. Cf-vit: A general coarse-to-fine method for vision transformer. In Proceedings of the AAAI Conference on Artificial Intelligence, Singapore, 17–19 July 2023; Volume 37; pp. 7042–7052.
45. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.
46. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, South Korea, 27 October–2 November 2019; pp. 1314–1324.
47. Heo, B.; Yun, S.; Han, D.; Chun, S.; Choe, J.; Oh, S.J. Rethinking spatial dimensions of vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 11936–11945.
48. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15908–15919.
49. Graham, B.; El-Nouby, A.; Touvron, H.; Stock, P.; Joulin, A.; Jégou, H.; Douze, M. Levit: A vision transformer in convnet's clothing for faster inference. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 12259–12269.
50. Dai, Z.; Liu, H.; Le, Q.V.; Tan, M. Coatnet: Marrying convolution and attention for all data sizes. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 3965–3977.
51. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
52. Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; Zhang, L. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 22–31.

53. Chen, Y.; Dai, X.; Chen, D.; Liu, M.; Dong, X.; Yuan, L.; Liu, Z. Mobile-former: Bridging mobilenet and transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5270–5279.
54. Guo, Y.; Lan, Y.; Chen, X. CST: Convolutional Swin Transformer for detecting the degree and types of plant diseases. *Comput. Electron. Agric.* **2022**, *202*, 107407. [CrossRef]

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