

Article

Lightweight Multiscale CNN Model for Wheat Disease Detection

Xin Fang ^{1,2} , Tong Zhen ^{1,2} and Zhihui Li ^{1,2,*}¹ College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China² Key Laboratory of Grain Information Processing and Control, Henan University of Technology, Zhengzhou 450001, China

* Correspondence: lizhihui@haut.edu.cn

Abstract: Wheat disease detection is crucial for disease diagnosis, pesticide application optimization, disease control, and wheat yield and quality improvement. However, the detection of wheat diseases is difficult due to their various types. Detecting wheat diseases in complex fields is also challenging. Traditional models are difficult to apply to mobile devices because they have large parameters, and high computation and resource requirements. To address these issues, this paper combines the residual module and the inception module to construct a lightweight multiscale CNN model, which introduces the CBAM and ECA modules into the residual block, enhances the model's attention to diseases, and reduces the influence of complex backgrounds on disease recognition. The proposed method has an accuracy rate of 98.7% on the test dataset, which is higher than classic convolutional neural networks such as AlexNet, VGG16, and InceptionresnetV2 and lightweight models such as MobileNetV3 and EfficientNetb0. The proposed model has superior performance and can be applied to mobile terminals to quickly identify wheat diseases.

Keywords: disease recognition; attention mechanism; inception; residual networks



Citation: Fang, X.; Zhen, T.; Li, Z. Lightweight Multiscale CNN Model for Wheat Disease Detection. *Appl. Sci.* **2023**, *13*, 5801. <https://doi.org/10.3390/app13095801>

Academic Editors: Antonio Fernández-Caballero and Byung-Gyu Kim

Received: 15 March 2023

Revised: 28 April 2023

Accepted: 28 April 2023

Published: 8 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. The Significance of Wheat Disease Detection

According to statistics, China's wheat cultivation area in 2022 was about 22.962 million hectares, with a production of 135.76 million tons, accounting for about 18% of the world's total wheat production. Wheat has high nutritional value and contains abundant carbohydrates, fats, and proteins, and many other substances essential for human survival. Wheat yield and quality are largely affected by diseases. The decline in wheat yield not only causes economic losses but also jeopardizes human life. Nowadays, the world's population is still growing and human dietary demands are rising, so it is necessary to improve the quality and yield of wheat to meet human material needs [1–4].

Wheat powdery mildew, wheat rust, and wheat leaf blight are typical and severe wheat diseases [5]. Due to these diseases, the wheat yield has been reduced by nearly one-third, bringing huge damage to food security and the agricultural economy. Controlling crop diseases has become a serious challenge. Disease detection and identification have become a vital research field for improving the high yield and quality of the crop [6].

1.2. Disease Identification in Wheat Based on Machine Learning and Deep Learning

In the early years, the detection of wheat diseases was mainly performed by manual inspection and identification, but manual identification had problems such as subjectivity, low efficiency, and low accuracy. With the development of technology, spectral analysis, machine learning, and deep learning are now widely used for wheat disease detection. Zhang et al. [7] used hyperspectral remote sensing to detect and distinguish yellow rust from nutrient stress. They detected yellow rust and mapped its spatial distribution based on the physiological reflex index PhRI. The proposed smart agriculture has motivated the

use of various machine learning algorithms for the detection of wheat diseases. Using hyperspectral wheat images and classification regression trees to identify the severity of powdery mildew, Zhang et al. [8] achieved more than 87.8% identification of disease infection levels, but they had inaccurate identification of mildly infected wheat with a high probability of this being mistaken for healthy or moderately infected leaves. To enable early detection, prevention, and control of crop diseases, Khan et al. [9] proposed a least squares regression model to detect early wheat disease severity with an overall accuracy of more than 82.35%. However, the high cost of hyperspectral equipment makes it difficult for the average farmer to afford it. Wang et al. [10] used spectral data and established a combined model to detect and identify wheat stripe rust and wheat leaf rust, with an overall identification accuracy of 82% on a test set. However, the model's recognition accuracy is bound to decrease unless the influence of various factors such as weather, soil, and complex background on the spectral data is eliminated or attenuated. Bao et al. [11] proposed an algorithm for identifying leaf diseases and their severity. First, they segmented the wheat disease images to obtain disease spot features, and then they recognized the segmented diseases and their severity with a maximum recognition accuracy of 94.16%. This makes an important contribution to the intelligent recognition of wheat leaf diseases.

In recent years, computer vision and deep learning have been used to detect crop diseases. Aboneh et al. [12] collected and labeled wheat disease image data and used five deep learning models to identify wheat diseases, and they found that the VGG19 model had the highest classification accuracy after experimental comparison. Liu et al. [13] introduced a two-layer inception structure and cosine similarity convolution into a normal convolution block. The proposed model achieved 97.54% accuracy for buckwheat disease detection. However, the inclusion of the inception structure also increases the time consumption. Jin et al. [14] focused on the generalization capability of the model as the first consideration, shaped wheat head spectral data into two-dimensional data, and fed it into a hybrid neural network, which achieved an accuracy of 84.6% on the validation dataset. This pushed the development of large-scale crop disease detection. To address the low accuracy of traditional methods, Deng et al. [15] used the Segformer algorithm to segment the stripe rust disease images, and the performance of the model was greatly improved after the data were enhanced. Nevertheless, this method only applies to fall wheat diseases. Su et al. [16] proposed an integrated Mask-RCNN-based FHB severity assessment method for high-throughput wheat spike identification and the accurate segmentation of FHB infestation under complex field conditions, which can help in the selection of disease-resistant wheat varieties. To effectively prevent the damage of yellow rust, SHAFI et al. [17] conducted a classification study on the types of wheat yellow rust infection and deployed the ResNet-50 model on smart edge devices to detect the severity of yellow rust. Obtaining high-resolution, low-cost, and large-coverage remote sensing data through drones can improve the accuracy and efficiency of disease identification. Huang et al. [18], using UAV remote sensing technology to identify and detect wheat leaf spot, significantly improved the efficiency of disease monitoring. Considering the large amount of effort required for data annotation, Pan et al. [19] proposed a weakly supervised method for detecting yellow showers disease of wheat photographed by UAV with 98% accuracy. Some diseases are difficult to detect without prominent characteristics. To improve the recognition of disease features, Mi et al. [20] introduced the CBAM module based on DenseNet and achieved 97.99% test accuracy on the wheat stripe rust dataset. However, the above mentioned methods have complex models and large computational volumes that are difficult to port to mobile devices. To reduce the model parameters and computational effort, Bao et al. [21] proposed a lightweight SimpleNet model with an accuracy of 94.1%. Adding the CBAM attention mechanism to the inverted residual blocks of this model made the wheat ear disease information more significant. However, this method is not applicable to other crop images.

1.3. The Advantages of Lightweight Models in Wheat Disease Detection and the Work of This Article

With the development of technology, mobile devices are becoming more and more mature. Mobile devices can use computer vision technology to intelligently identify and diagnose crop diseases from the leaves, determine the type and severity of diseases, and provide farmers with timely suggestions for prevention and control. Therefore, lightweight networks have great potential and advantages in agricultural disease detection. Lightweight network models have the characteristics of high accuracy, low parameter numbers, and computational costs, and can serve scenarios with limited computing resources such as mobile devices and embedded systems. For example, without the need for professionals or laboratory equipment, smartphones or other portable devices can be used for testing, provide reasonable suggestions for prevention and control based on the test results, and interact with human experts or other data sources to improve control effectiveness and agricultural productivity. There is no doubt that the methods summarized above have achieved favorable results in wheat disease detection. However, these methods also have some limitations, such as hyperspectral remote sensing technology having high accuracy in disease detection but requiring very expensive equipment; large-scale network models being effective in disease detection but being difficult to run on mobile devices; environmental factors such as wind, temperature, and humidity that can affect the flight stability and safety of drones; the types of diseases studied being relatively limited, and the research on wheat disease detection under complex backgrounds being insufficient; and the coexistence of multiple different diseases and occluded diseases that are difficult to identify. To solve these problems, this paper develops a wheat disease identification method with a simple structure, a small amount of calculation, strong generalization ability, wide applicability, and that can be equipped with mobile devices. This method is of great significance to help farmers identify wheat diseases and improve wheat yield and quality. The contributions of this study are as follows: We design a lightweight Inception-ResNet-CE model for the automatic identification of wheat diseases on mobile and edge terminals. CE is composed of CBAM and ECA attention mechanisms.

- (1) We combine three Inception structures with residual structures, which can increase the depth and receptive field of the network, aggregate image information at different scales, and rapidly extract disease features.
- (2) We introduce the CBAM and ECA attention mechanisms into the residual blocks in the Inception-ResNet model to enhance the model's ability to capture disease characteristics and reduce the interference of complex backgrounds in images on model recognition performance.
- (3) The Inception-ResNet-CE model has only 4.24 M parameters and achieves a recognition accuracy of 98.78% on the validation dataset. It can be applied to the automatic recognition of wheat diseases on edge terminals or mobile devices.

The rest of the paper is organized as follows: Section 2 presents the experimental data and the proposed Inception-ResNet-CE model; Section 3 presents our five sets of experimental results; Section 4 discusses the optimal structure of our model; and Section 5 summarizes our work and looks at future research directions.

2. Materials and Methods

2.1. Image Dataset

The wheat disease dataset used in this paper has seven classes, including six disease classes and one healthy class. Some of these data are collected from the LWDCD2020 dataset, and the other part is captured by mobile phone photography. These disease images are taken from multiple perspectives and contain complex backgrounds, disease characteristics at different stages, and similar features among different wheat diseases. Figure 1 illustrates the distribution of each class.

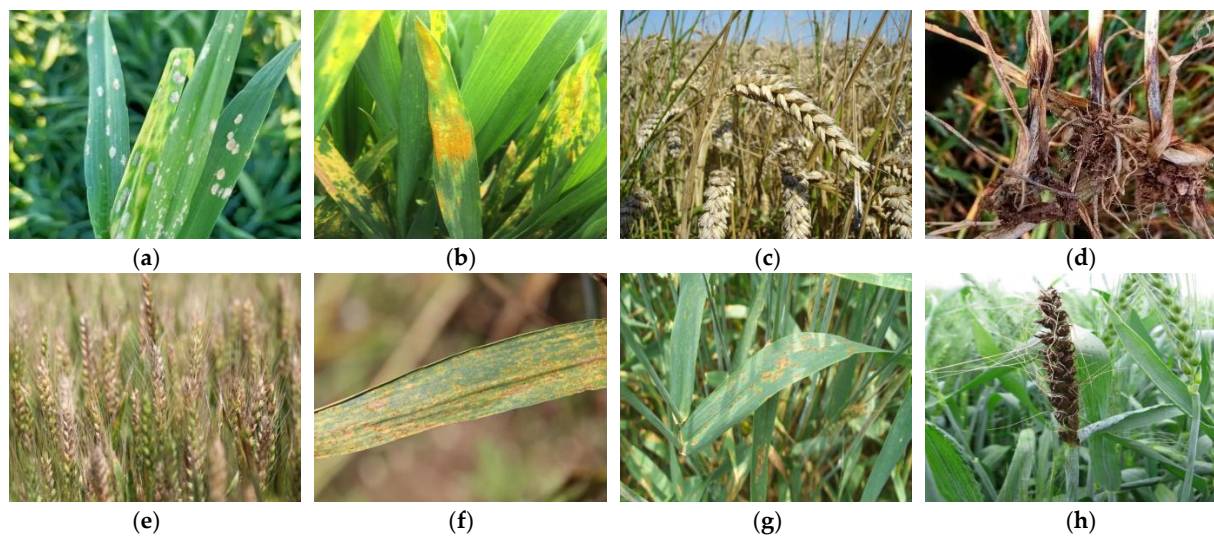


Figure 1. Wheat diseases. (a) Powdery mildew; (b) leaf rust; (c) healthy wheat; (d) root rot; (e) fusarium head blight; (f) leaf rust; (g) tan spot; (h) black chaff.

The LWDCD2020 [22] dataset has 12,000 images and contains 9 types of wheat disease images and 1 type of healthy image. However, the authors published data for only about 4500 images, which were divided into 3 groups of wheat diseases and 1 group of healthy categories. The details are as follows: leaf rust, crown root rot, healthy wheat, and black chaff of the wheat. Considering the small number of disease categories, we collected 829 images of wheat diseases (230 powdery mildew, 387 fusarium head blight, 212 tan spots). In total, 2174 images of wheat diseases (600 healthy, 560 rust, 504 rots, and 510 black chaff) were selected from the LWDCD2020 dataset. These two sets were combined into the experimental dataset of 3003 images used for the experiment.

2.2. Dataset Preprocessing

Since the color of wheat images can deviate from the true color due to different illumination, this can bring errors to the subsequent network model recognition. In this study, contrast enhancement is applied to the image in order to reduce the effect of light inhomogeneity on the image. Data augmentation is a technique that expands the training set to enhance the generalizability and robustness of deep learning. Considering the insufficient number of wheat disease images, the neural networks may overfit to the training set, leading to overfitting problems in the CNNs during testing. Therefore, the wheat disease data were augmented by rotating, symmetrically flipping, and increasing the contrast, among other operations. The original dataset was augmented to 8495 images (1156 leaf rust images, 1380 powdery mildew images, 1342 wheat smut images, 1096 root rot images, 1161 scab images, 1272 tar spot images, and 1086 healthy images). Several examples of enhanced images are shown in Figure 2. The augmented dataset was split into the training set and test set with an 8:2 ratio. The results are shown in Table 1.

Table 1. Wheat disease data after enhancement.

Wheat Types	Images	Training Images	Testing Images
healthy	1086	869	217
leaf rust	1156	925	231
powdery	1380	1104	276
wheat loose smut	1342	1073	269
root rot	1096	877	219
fusarium head blight	1161	929	232
tan spot	1272	1018	254

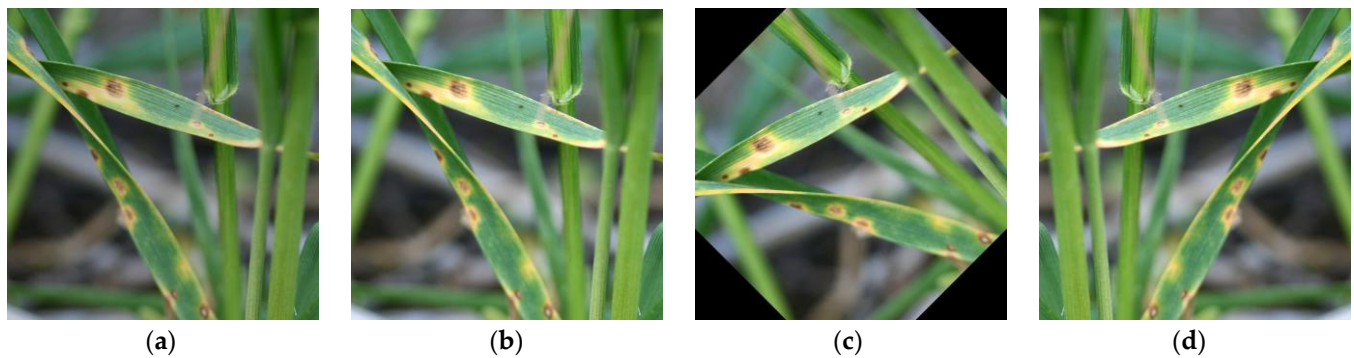


Figure 2. Wheat image enhancement. (a) Original image; (b) contrast enhancement; (c) rotation 45°; (d) horizontal flip.

2.3. Proposed Approach

CNNs [23] are widely popular neural networks. The earliest convolutional neural model for recognizing handwritten digits was LeNet. Nowadays, convolutional neural networks have achieved breakthroughs in numerous fields. The advancement of computer hardware and the continuous development of deep learning theory provides convolutional neural networks with unlimited potential for improvement. In this paper, we combine the Inception modules with the residual modules and introduce the attention mechanisms into the residual structures. We propose a lightweight multiscale CNN model to identify and classify six wheat diseases and evaluate their performance.

2.3.1. Inception Structure

The Inception [24] module was proposed by the Google team and is the core subnetwork structure in the classic GoogLeNet model. Subsequently, five versions (Inception-v1 to v4, Xception) were developed, and each version is an optimization and improvement of the previous version. The Inception module can automatically select the appropriate convolution kernels and pooling operations to improve the performance and efficiency of the network. The main advantage of the Inception module is that it can extract features at different scales, increase the width and depth of the network, speed up training, and prevent overfitting.

2.3.2. ResNet Model

ResNet [25] is a deep convolutional neural network architecture that can build very deep network structures, such as 18 layers, 34 layers, 50 layers, 101 layers, and 152 layers, by using residual units and skip connections. The characteristic of ResNet is that it can effectively address the degradation problem of deep networks; that is, as the depth of the network increases, the performance does not improve or even deteriorates. The advantage of the residual structure is that it can simplify the learning process and facilitate the propagation of gradients, making it easier for the network to learn to identity mappings or residual functions. The residual structure can also break the symmetry of the network, increase the rank of weight matrices, make the network more expressive, and prevent network degradation. As shown in Figure 3, in ResNet architecture, through skip connections, the output of a layer is no longer $F(x)$, but $H(x) = F(x) + x$. However, learning $H(x)$ directly is difficult. To facilitate learning, it is easier to reformulate the output as a residual function $F(x) = H(x) - x$.

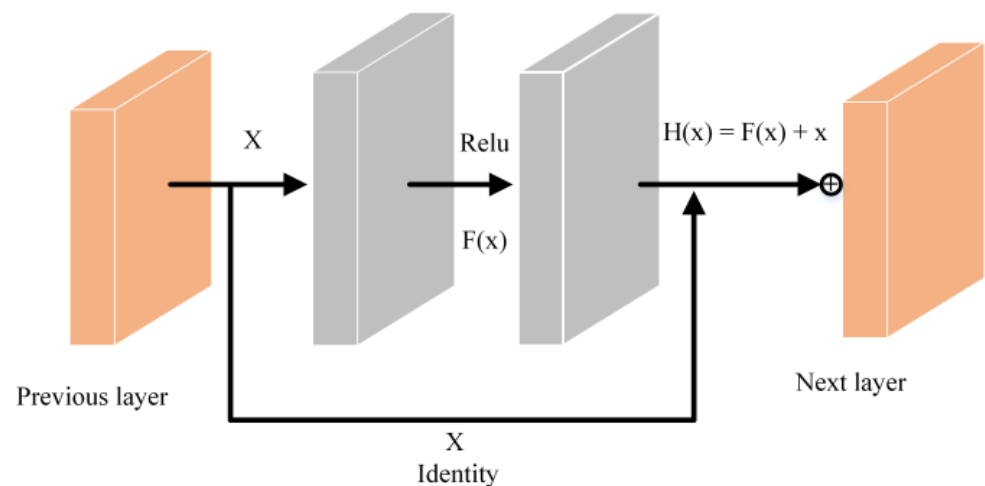


Figure 3. Residual module diagram.

2.3.3. Attentional Mechanisms

The attention mechanism is an essential core technology in deep learning and is widely applied in various fields. It helps the model focus on key regions, reduce the interference of unimportant information, and improve the efficiency and accuracy of the model. Thus, this paper introduces an attention mechanism to the residual structure, which can effectively enhance the model's ability to discriminate wheat diseases in a complex context. Attention mechanisms can be classified into channel attention mechanisms, spatial attention mechanisms, and mixed attention mechanisms. SE [26], ECA [27] are the frequently used channel attention mechanisms. SE is the beginning of the channel attention mechanism, and its core role is to automatically learn the reward and punishment feature weights through a fully connected network. ECA is an improvement of SE, using 1×1 convolution to exchange channel information. To acquire remote spatial interactions at precise locations, CA [28] divides the global level pooling into two steps, first for the height direction and then for the width direction. The key of RAM [29] is to focus the network on vital regions and reduce the amount of computation. The commonly used hybrid attention mechanisms are CBAM [30] and NAM [31]. CBAM is a lightweight general-purpose attention module. Due to its integration of channel attention and spatial attention, the attention mapping can be injected from two dimensions of channel and space in any intermediate feature map of the neural network, enabling adaptive feature refinement of the input feature map and improving the performance of detection. Therefore, CBAM has become a hot topic of interest, and many scholars have embedded CBAM modules in their models. Figure 4a shows the overall structure of the CBAM module. Figure 4b,c show the channel attention module and the spatial attention module, respectively. When the attention mechanism is added to the convolution block, the output of the convolution layer first goes to the channel attention module for weighting, then to the spatial attention module, and finally gets weighed once more.

Figure 4b shows the channel attention module diagram. First, both average pooling and max pooling operations need to be performed on the feature map, then their results are fed into a multilayer perceptron (MLP), the features outputted from MLP are summed up, and finally, a sigmoid activation function is applied to generate a channel attention feature map. The channel attention mechanism is concerned with which content on the picture is of importance. Figure 4c shows the spatial attention module diagram. The feature map output from the channel attention module in the previous stage is taken as the input feature map. First, conduct max pooling and average pooling on the feature map based on channel, and then conduct concat operation on the two results based on channel. Finally, descending and sigmoid operations are performed to generate spatial attention features. The spatial attention mechanism is concerned with where important information is located. In this study, CBAM was added to the remaining blocks and it was able to target

wheat disease regions in the images, reducing the effect of complex background on disease identification. NAM is integrated in the same way as CBAM, but NAM has reworked the channel attention and spatial attention submodules. NAM focuses on the weight factor to optimize the attention mechanism and uses batch-normalized scaling factors to strengthen the weights. NAM suppresses smaller significant weights and imposes a weight sparsity penalty on the attention modules so that they have higher computational efficiency while maintaining similar performance.

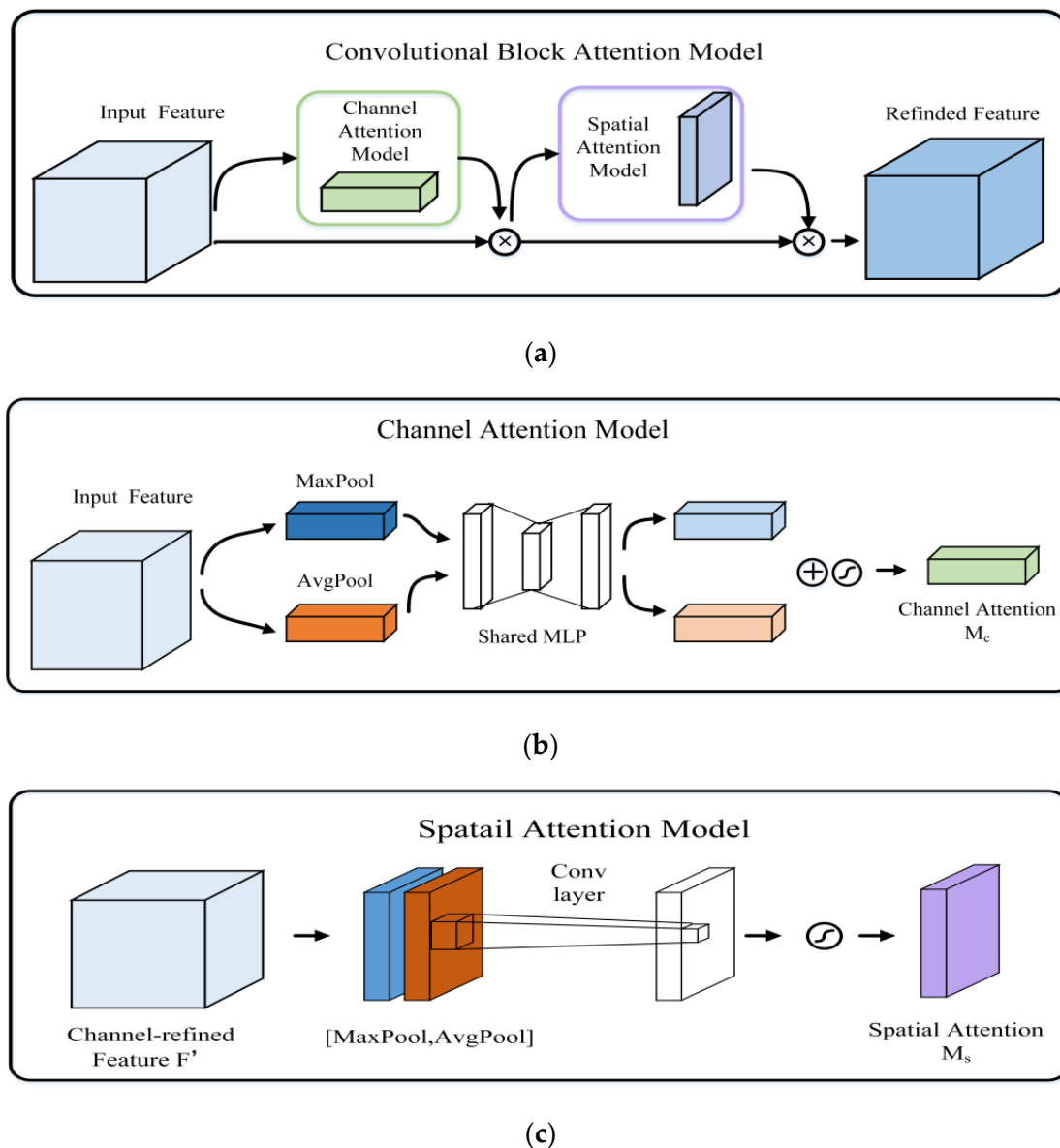


Figure 4. CBAM module diagram. (a) CBAM architecture; (b) channel attention; (c) spatial attention.

2.3.4. Proposed Model

The Inception module can expand the network width and speed up the training. The residual block can not only solve the degradation problem but also address the gradient problem. Combining these two modules reduces network complexity and redundancy while maintaining high accuracy. The attention mechanism can assess and weigh channel attention and spatial attention simultaneously, which is beneficial to improve the model's efficiency and accuracy.

Therefore, this paper proposes a multiscale CNN information fusion model with multiple attention mechanisms, named Inception-ResNet-CE (IRCE) model, as shown

in Figure 5. This model is an efficient and lightweight neural network model suitable for mobile devices, which can efficiently and quickly identify wheat diseases in complex backgrounds. The model framework includes six Residual-CE structures (as shown in Figure 6), three Inception structures (as shown in Figure 7), three pooling layers, and a fully connected layer.

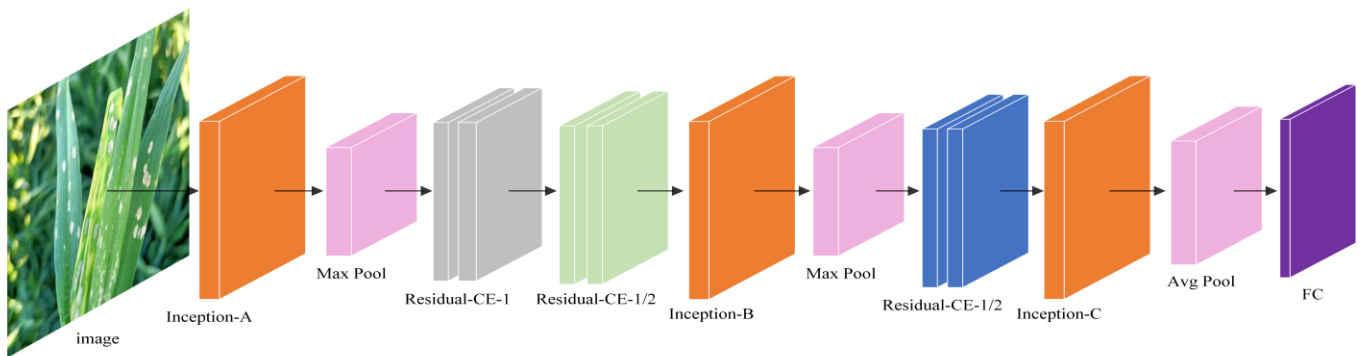
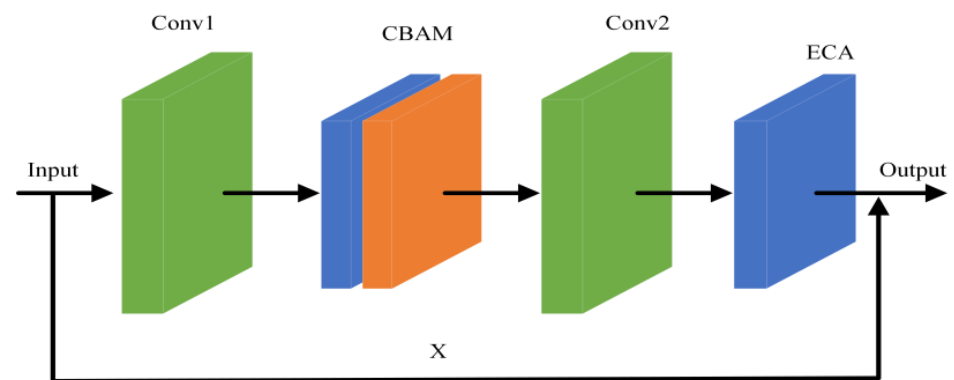
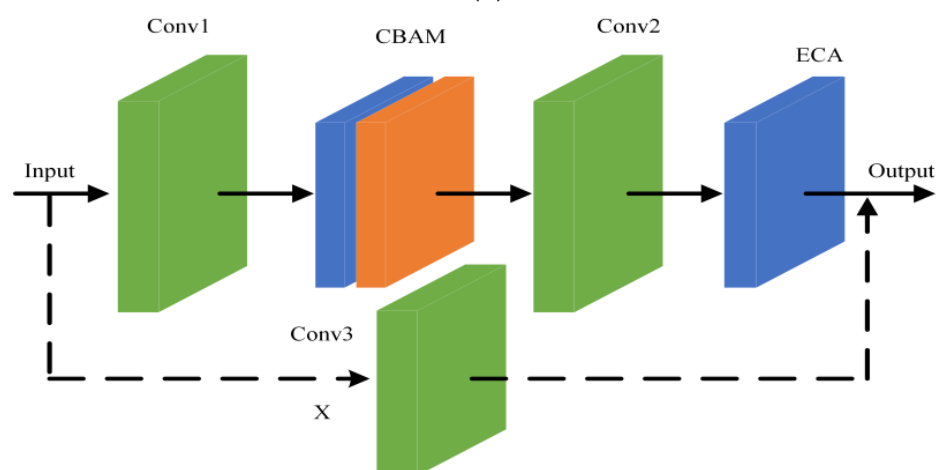


Figure 5. IRCE model structure.



(a)



(b)

Figure 6. Residual attention structure. (a) Residual-CE-1; (b) Residual-CE-2.

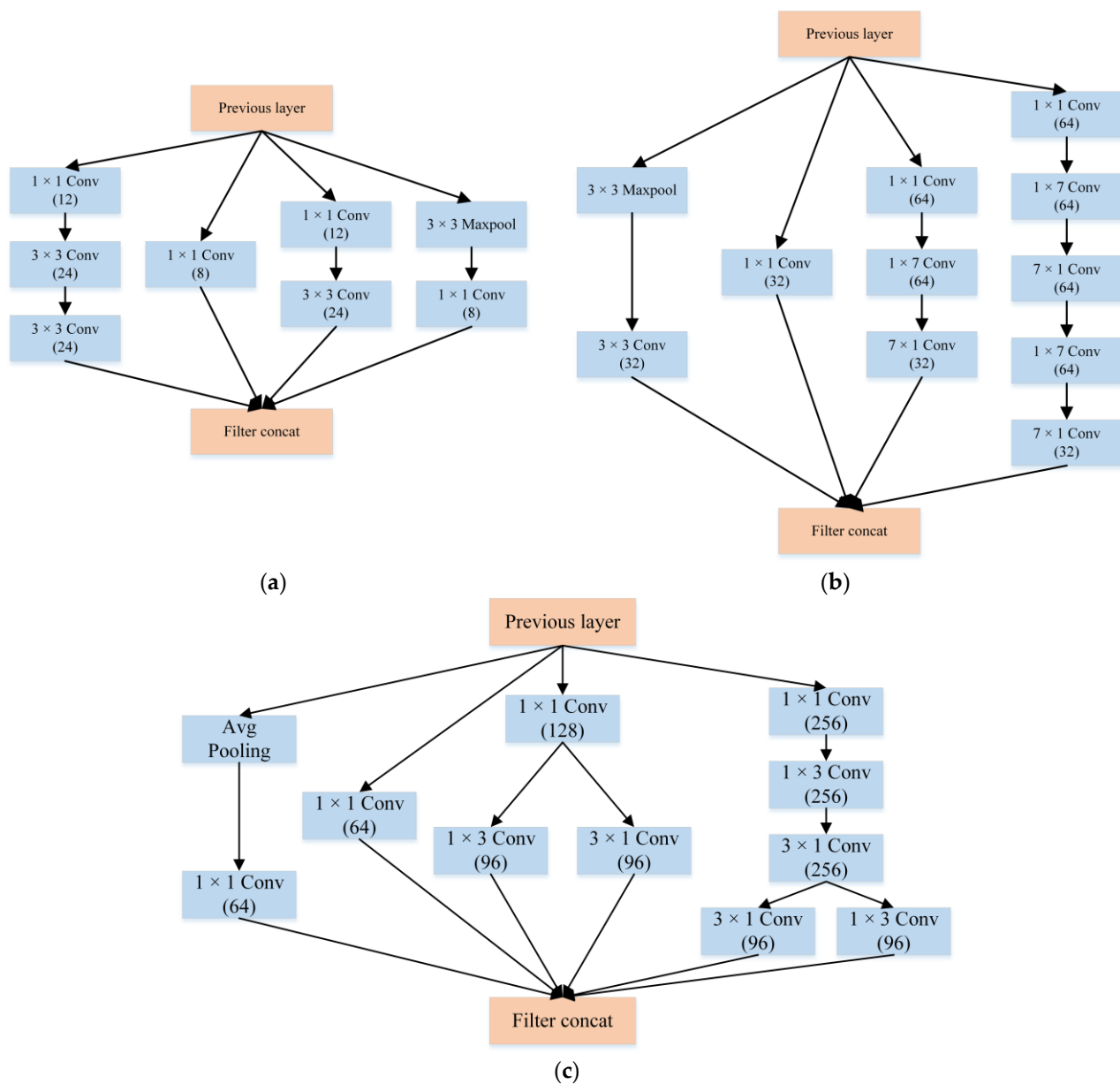


Figure 7. Inception structure. (a) Inception-1; (b) Inception-2; (c) Inception-3.

We list all parameters of the IRCE model in Table 2. To obtain comprehensive image feature information, the depth and width of the network were expanded by adding three different Inception modules to the model. Inception-1 module contains a 1×1 convolution, a 3×3 convolution, and a MaxPool. To reduce the computation, two 3×3 convolutions replaced the original 5×5 convolutions. Branch1~branch4 has seven convolution kernels (the number of kernels is 8, 12, 24, 8, 12, 24, 24) and a 3×3 MaxPool. The Inception-2 structure was combined using 1×1 convolution, asymmetric 1×7 convolution, and 7×1 convolution in series. To reduce complexity, two 1-dimensional 1×7 convolutions and 7×1 convolutions decomposed the original 7×7 convolution. Branch1~branch4 has nine convolution kernels (the number of kernels is 32, 32, 64, 64, 32, 64, 64, 64, 32) and a 3×3 MaxPool. The Inception-3 structure was used in a series-parallel combination of 1×1 convolution, asymmetric 1×3 convolution, and 3×1 convolution. Branch1~branch4 has ten convolution kernels (the number of kernels is 64, 64, 128, 96, 96, 256, 256, 256, 96, 96) and a 3×3 AvgPool.

Table 2. IRCE model structure parameters.

Network Layer		Inchannel, Outchannel, Kernel_Size, Stride, Padding	
Inception-1	Image input	$224 \times 224 \times 3$	
	branch1	$3, 8, 1 \times 1, 2, 0$	
	branch2	$3, 12, 1 \times 1, 2, 0$	
	branch3	$12, 24, 3 \times 3, 1, 1$	
		$—, —, 3 \times 3, 1, 2$	
		$3, 8, 3 \times 3, 1, 1$	
	branch4	$3, 12, 1 \times 1, 1, 0$	
	$12, 24, 3 \times 3, 1, 1$		
$24, 24, 3 \times 3, 1, 1$			
Filter concatenation		$112 \times 112 \times 64$	
MaxPool		$—, —, 3 \times 3, 2, 1$	
Residual-CE-1		$64, 64, 3 \times 3, 1, 1$	
Residual-CE-1		$64, 64, 3 \times 3, 1, 1$	
Residual-CE-2		$64, 128, 3 \times 3, 2, 1$	
Residual-CE-1		$128, 128, 3 \times 3, 1, 1$	
Inception-3	branch1	$128, 32, 1 \times 1, 2, 0$	
	branch2	$—, —, 3 \times 3, 2, 1$	
		$128, 32, 1 \times 1, 1, 0$	
		$128, 64, 1 \times 1, 2, 0$	
	branch3	$64, 64, 1 \times 7, 1, [0, 3]$	
		$64, 32, 7 \times 1, 1, [3, 0]$	
	branch4	$128, 64, 1 \times 1, 2, 0$	
		$64, 64, 1 \times 7, 1, [0, 3]$	
		$64, 64, 7 \times 1, 1, [3, 0]$	
		$64, 64, 1 \times 7, 1, [0, 3]$	
$64, 32, 7 \times 1, 1, [3, 0]$			
Filter concatenation		$14 \times 14 \times 128$	
MaxPool		$—, —, 3 \times 3, 2, 1$	
Residual-CE-2		$128, 256, 3 \times 3, 2, 1$	
Residual-CE-1		$256, 256, 3 \times 3, 1, 1$	
Inception-3	branch1	$256, 64, 1 \times 1, 2, 0$	
	branch2	$—, —, 3 \times 3, 2, 1$	
		$256, 64, 1 \times 1, 1, 0$	
	branch3	$256, 128, 1 \times 1, 2, 0$	
		$128, 96, 1 \times 3, 1, [0, 1]$	$128, 96, 3 \times 1, 1, [1, 0]$
	branch4	$256, 256, 1 \times 1, 2, 0$	
		$256, 256, 3 \times 1, 1, [1, 0]$	
		$256, 256, 1 \times 3, 1, [0, 1]$	
$256, 96, 1 \times 3, 1, [0, 1]$		$256, 96, 3 \times 1, 1, [1, 0]$	
Filter concatenation		$2 \times 2 \times 512$	
Avg_pool		$1 \times 1 \times 512$	
Fc		7	

“—” indicates that the pooling operation has no corresponding Inchannel and Outchannel.

The Residual-CE module can recombine the features of the original wheat disease image, which is beneficial to the learning of the model. Residual-CE can also map samples from high-dimensional feature space to low-dimensional feature space, and make the mapped samples still have excellent separability. The Residual-CE-1 structure includes two 3×3 convolutional layers, a CBAM module, an ECA module, and an identity map. The Residual-CE-2 adds a 1×1 convolution short link to Residual-CE-1.

2.4. Model Optimization

2.4.1. Optimizer

Optimizers often play a crucial role in machine learning and deep learning, and the performance of a model using different optimizers can potentially vary greatly. The commonly used optimizers are gradient descent optimizer, momentum optimizer, and adaptive learning rate optimizer. Stochastic gradient descent [32] is commonly used in

machine learning, which is characterized by fast learning through frequent updates, but frequent updating of the model causes huge computational effort, which is very bad for large-scale data training. The momentum optimizer [33] is used to solve the problem of oscillations leading to reduced learning speed. Adam [34] optimizer is a way to train deep learning models faster and better. It uses the average and the variation in the past gradients to adjust the learning rate for each parameter. Adam is a further improvement combining the advantages of AdaGrad and RMSProp algorithms with simple adjustment parameters. In this paper, we experimentally compare the effects of four optimizers—AdaGrad, RMSProp, SGD, and Adam—and select the most effective optimizer.

2.4.2. Learning Rate

The learning rate is an essential hyperparameter in deep learning, which controls the convergence of the objective function, and when it converges to a minimum, can even change the merit of the model. The optimal learning rate can boost both the accuracy and stability of the model. In this paper, we use StepLR to adjust the learning rate. StepLR is a learning rate adjustment method, which can multiply the learning rate by a decay factor (*gamma*) every certain number of epochs (*step_size*) to achieve better optimization results. The formula for StepLR is:

$$lr = lr \times gamma^{floor(\frac{epoch}{step_size})} \quad (1)$$

where (*lr*) is the learning rate, (*gamma*) is the decay factor, (*epoch*) is the current number of training epochs, (*step_size*) is the step size of the learning rate adjustment, and the (*floor*) is a function of rounding down.

2.4.3. Regularization

The purpose of regularization is to enhance the generalization ability of the model and prevent overfitting. There are generally two types of regularization: L1 and L2. L2 focuses on the absolute value of the weights, and the larger the absolute value, the more severely the weights are punished. Only when the weight is absolutely 0, are they not penalized. The function of L1 is to assign the values of the model parameters to 0 and sparsify the weight parameters to prevent the model from overfitting. Because L1 is suitable for models with sparse and few relevant features, and L2 can prevent overfitting and optimize the solution by using weight decay, L2 is used as a means to solve model overfitting in this paper.

2.5. Model Performance Evaluation Metrics

For image classification problems, the performance of a model is usually evaluated in five ways: accuracy, prediction rate, recall, F1 score, and confusion matrix. The accuracy is the proportion of the number of examples correctly classified by the model to the total number of examples, and it is a very intuitive evaluation metric. Precision refers to the proportion of samples predicted to be in the positive category that is predicted correctly, reflecting the model's ability to discriminate between negative samples. Recall indicates how many positive cases in the sample were predicted correctly. A higher recall indicates that the model is better at identifying positive cases and misses fewer positive cases. The F1-score reflects the balance of the model in terms of accuracy and recall. A higher F1-score indicates that the model performs better in both aspects. The confusion matrix is a table showing the prediction results of a classification model. It can count the number of samples that the model predicts correctly and incorrectly, as well as the actual and predicted categories of the samples. The confusion matrix can help us analyze the strengths and weaknesses of the model, as well as calculate other evaluation metrics.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$F1 - score = \frac{2TP}{2TP + FP + FN} \quad (5)$$

where TP is a positive example of a correct prediction, FN is a positive example of a wrong prediction, FP is a negative example of a wrong prediction, and TN is a negative example of a correct prediction.

3. Results

The experimental software and equipment configuration parameters are as follows: Ubuntu 18.04, 64-bit Linux operating system, Tesla T4 graphics card with 16 G memory, PyTorch 1.10, and CUDA 10.1.

There are five groups of experimental comparisons in this paper. The first group selects an appropriate optimizer. The second group explores the impact of the inception module on the model. The third group explores the effect of attentional mechanisms on the model. The fourth group compares the proposed model with the current mainstream algorithms in terms of performance and accuracy. The fifth group compares the proposed model with the classic lightweight CNN models. We set the basic parameters: epoch = 70, learning rate = 0.001, batch_size = 64, weight_decay = 0.001, step_size = 25, and gamma = 0.01. Figure 8 shows the accuracy curve of the IRCE model on the training dataset. It can be seen from the figure that the accuracy rate increases rapidly in the early training. When the model is between 10 and 30 epochs, the accuracy rate fluctuates, but the overall trend is rising. After 30 epochs, the accuracy rate starts to converge slowly, and when the model reaches 40 epochs, the training accuracy basically stops and approaches a stable value.

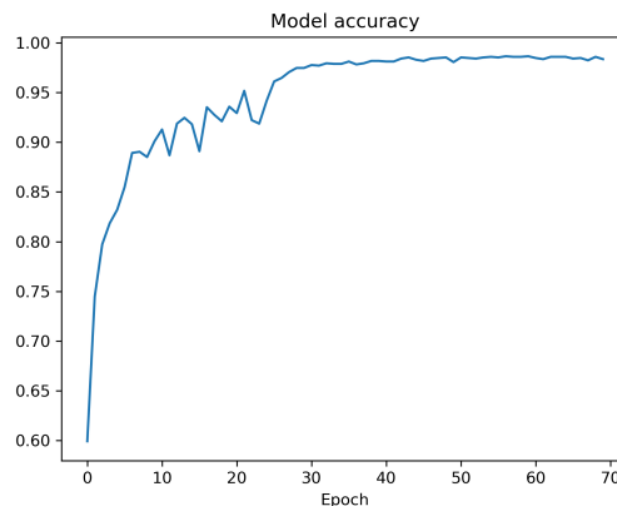


Figure 8. Accuracy in training.

3.1. Comparison of Effects of Different Optimizers

In order to select the optimizer with the best effect, we conducted four groups of comparative experiments using different optimizers. For each optimizer, we carried out 10 tests, and we obtained the average accuracy and variance for each optimizer. Table 3 shows the results. The model using the Adam optimizer achieved an average accuracy of 98.64%, which was 1.46% and 9.08% higher than those using RMSprop and AdaGrad, respectively. The model using the SGD optimizer only achieved an average accuracy

of 75.97%. SGD updates the model parameters with one wheat disease sample at a time to minimize the loss function. However, it converges slowly, tends to fall into local optima, and cannot adapt to the learning rate. RMSprop and Adagrad adjust the learning rate according to the historical gradients of each parameter to adapt to different parameters and features. However, they are prone to gradient vanishing or exploding. Adam combines the momentum and adaptive learning rate and uses the mean and variance of gradients to adjust the learning rate, thus accelerating convergence and improving model stability. Therefore, Adam was chosen as the optimizer for the subsequent experiments in this paper.

Table 3. Test results for different optimizers.

Optimizer	Average Accuracy (%)
SGD	75.97 $\pm$ 0.21
RMSprop	97.18 $\pm$ 0.15
AdaGrad	89.56 $\pm$ 0.13
Adam	98.64 $\pm$ 0.12

3.2. Exploring the Impact of the Inception Module on the Model

In order to explore the influence of the Inception module on the model, we conducted four sets of experiments using the wheat disease dataset and tested different models with Inception modules. The results are shown in Table 4. It can be seen from the table that only the Inception-1 module was added to the model, and the accuracy, precision, recall, and F1-score of the model increased by 0.76%, 0.75%, 0.83%, and 0.78%, respectively, while Param slightly increased, and FLOPs increased by 0.9 G. The model with Inception-1 and Inception-2 had a lower performance improvement than that of adding only the Inception-1 module. The model with the three Inception modules achieved the greatest improvement in performance but also increased Param by 1.18 M, and FLOPs by double. Based on the comparison, we conclude that adding more Inception modules improves the detection accuracy of the model for wheat diseases but also increases the number of parameters and computations.

Table 4. The impact of the Inception module on the model.

Inception-1	Inception-2	Inception-3	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Param (M)	FLOPs (G)
×	×	×	95.40	95.53	95.52	95.53	3.06	0.42
✓	×	×	96.16	96.28	96.35	96.31	3.07	0.51
✓	✓	×	96.35	96.32	96.36	96.33	3.23	0.62
✓	✓	✓	98.76	98.77	98.81	98.79	4.24	0.84

Figure 9 shows the feature maps of layer 1, layer 2, and layer 3 of the model. The first row of the feature maps does not have any Inception module, the second row contains the Inception-1 module, the third row contains both the Inception-1 and Inception-2 modules, and the fourth row contains all three Inception modules. The shallow layers of the model mainly extract features such as image texture, edge, and color. The deep feature maps of the model reduce the visual information of the image and increase the abstract information. The model without any Inception module loses a lot of disease features. Adding more Inception modules enhances the model's ability to integrate the features in the shallow feature maps with the deep semantic information, highlighting the characteristics of the diseased area. The feature map of layer 3 for the model without any Inception module in Figure 9c still preserves clear visual information of the image, which may be caused by the insufficient depth of the model.

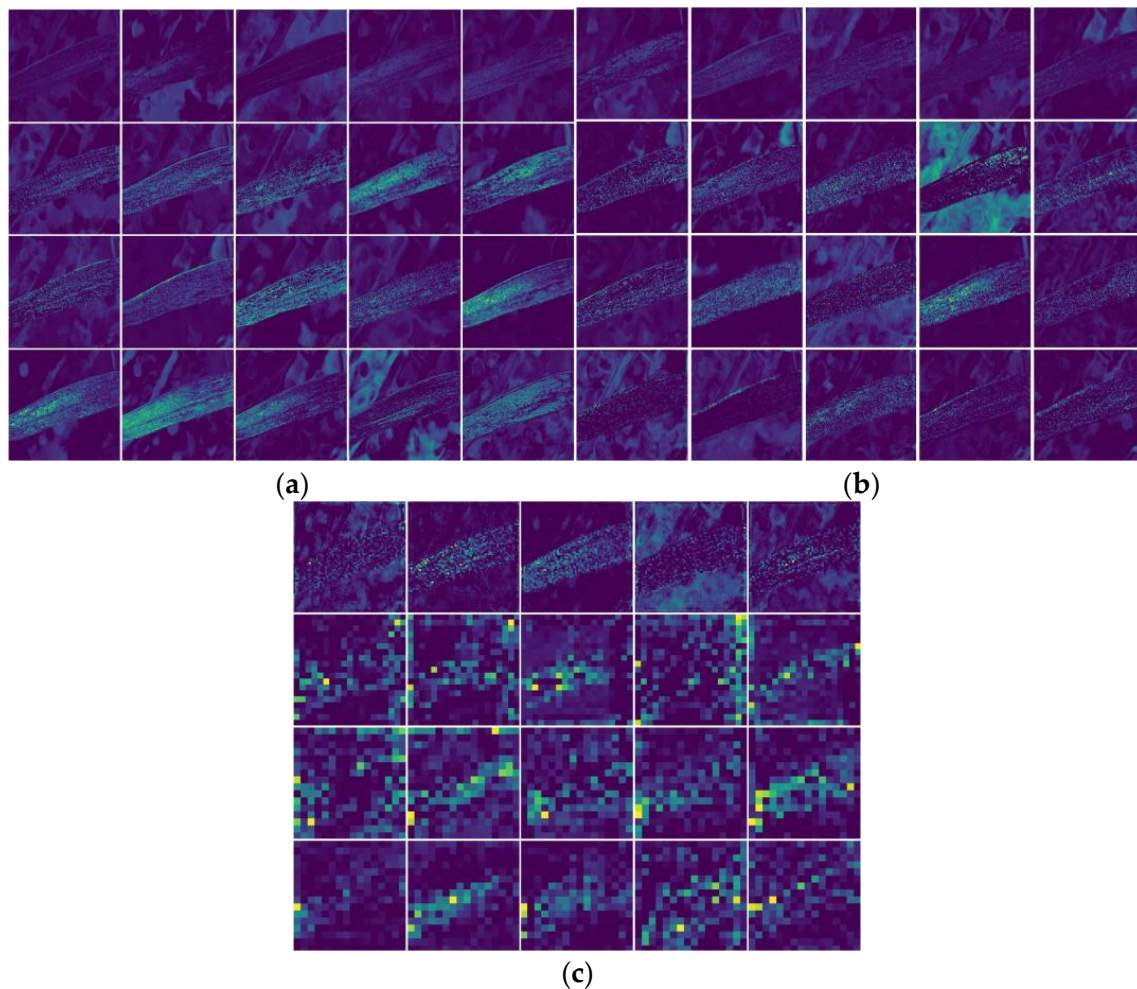


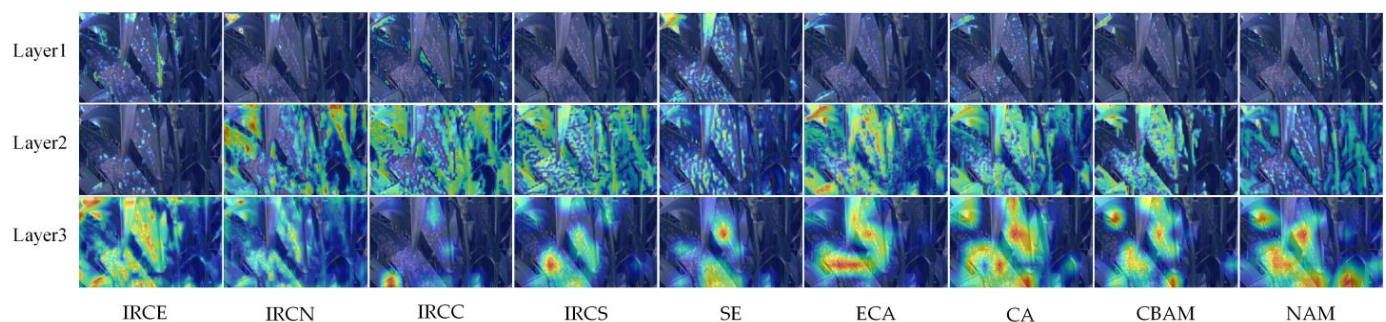
Figure 9. Visualization result of feature map output by layer. The first row of the feature maps does not have any Inception module, the second row contains the Inception-1 module, the third row contains both the Inception-1 and Inception-2 modules, and the fourth row contains all three Inception modules. (a) Feature map of layer 1; (b) feature map of layer 2; (c) feature map of layer 3.

3.3. Effect of Attentional Mechanisms on the Model

In the experiment, we investigated how CBAM, NAM, SE, CA, and ECA affected the performance of a residual structure model, and designed nine sets of experiments to compare their effects. Table 5 shows the results of adding different attention mechanisms to the model. The table indicates that adding an attention mechanism has little impact on the number of model parameters and computations. Adding only one attention mechanism improves the accuracy, precision, recall, and F1-score of the model for all mechanisms except CA; ECA is the most helpful to the model, as evidenced by the increase in its accuracy, precision, recall, and F1-score by 0.59%, 0.51%, 0.62%, and 0.57%, respectively. Adding two attention mechanisms increases the detection accuracy of the model by 1.18–2.71%, with CBAM + ECA working best. To further explore the role of the attention mechanism, we used Grad-CAM to draw a heat map of the model with the attention module and visualize the network's focus area on wheat diseases. Figure 10 shows that each row corresponds to a residual layer and each column corresponds to the added attention mechanism. The model with an added attention mechanism can focus on the diseased area of wheat leaves, effectively obtaining the relevant information in the pictures, which is very helpful for disease identification.

Table 5. Effect of attentional mechanisms on the model.

Methods	CBAM	ECA	SE	CA	NAM	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Param (M)	FLOPs (G)
IR	×	×	×	×	×	96.05	96.13	96.17	96.15	4.22	0.83
IRCBAM	✓	×	×	×	×	96.46	96.54	96.55	96.55	4.24	0.84
IRNAM	×	×	×	×	✓	96.11	96.02	96.31	96.25	4.22	0.83
IRSE	×	×	✓	×	×	96.46	96.54	96.56	96.55	4.22	0.83
IRCA	×	×	×	✓	×	95.81	95.84	95.94	95.89	4.22	0.83
IRECA	×	✓	×	×	×	96.64	96.64	96.79	96.72	4.22	0.83
IRCS	✓	×	✓	×	×	97.23	97.32	97.25	97.27	4.24	0.84
IRCC	✓	×	×	✓	×	97.66	97.66	97.67	97.66	4.24	0.84
IRCE	✓	✓	×	×	×	98.76	98.77	98.81	98.79	4.24	0.84

**Figure 10.** Heat map of nine experiments.

The heat map shows that in the first residual layer, SE performs the best in detection, but it also misclassifies some healthy areas as diseased ones; CBAM + CA detects edge wheat leaves and basically avoids diseased areas; and CBAM + SE pays little attention to disease characteristics. For the second residual layer, except for CBAM + ECA and SE, all the other attention mechanisms greatly improve the model's focus on the disease, but they over-attend to the image features, resulting in some background features also being considered as characteristics of the disease. For the third residual layer, all models except CBAM + CA focus on the diseased areas, but only CBAM + ECA identifies the disease features most clearly and detects almost all of them. ECA, CA, CBAM, and NAM perform similarly in detection, but they only focus on the main regions of the disease and ignore the features that exist at the margins. After comparing nine sets of experiments, we chose CBAM + ECA as the best-performing method.

3.4. Comparison of the Proposed Model with the Classical CNN Model

To test the performance and effect of the proposed model, we compared it with six classic CNN models: AlexNet, VGG16, ResNet34, ResNet50, ResNet101, and Inception-resnetV2. After training, we drew the loss and accuracy curves of the seven models, as shown in Figure 11, to clearly and quickly compare the accuracy of the algorithms. The figure shows that the loss curves of InceptionresnetV2, ResNet34, ResNet50, and ResNet101 converge at the same speed as the ACC curves, slightly faster than those of the VGG16 and AlexNet models. Our IRCE model not only has the lowest training loss and the highest training accuracy but also the fastest convergence speed.

In order to further demonstrate the advantages of our model, we compared seven indicators of various models. Table 6 shows that AlexNet has the least training time and the lowest FLOPs, but its F1-score is only 87.33%; VGG16 has the largest number of parameters, reaching 138.37 M, but its accuracy and F1-score are only 87.12% and 87%, respectively. The ResNet series models show little difference in the accuracy of disease detection. Among them, ResNet34 has an accuracy rate of 95.05%, ResNet50 has an accuracy rate of 96.52%, and ResNet101 has an accuracy rate of 95.68%. It is possible that ResNet101 has overlearned

other features as well as diseases, resulting in a lower accuracy rate than ResNet50. The accuracy rate of the InceptionresnetV2 model is 96.70%, but its parameters reach 55.80 M, its FLOPs reach 14.98 G, and its training time is as high as 5.8 h. The IRCE model has the minimum parameter of only 4.24 M, its FLOPs is second only to AlexNet, and its accuracy rate, precision rate, recall rate, and F1 average for wheat disease classification are 98.76%, 98.81%, 98.76%, and 98.78% respectively. The accuracy of the IRCE model is 2.06~3.71% higher than the ResNet series and InceptionresnetV2 models, and 11.61% higher than AlexNet. The time required to train the model is only 1.34 h, so the IRCE model proposed in this paper has excellent performance.

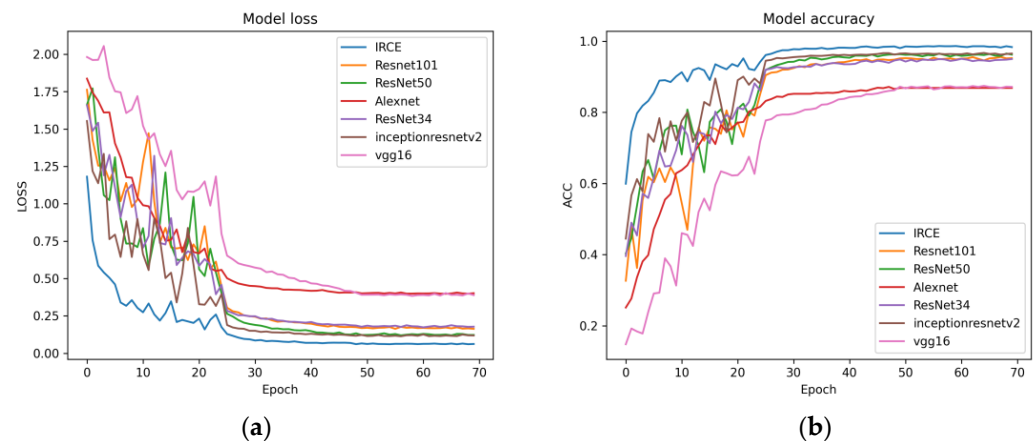


Figure 11. Seven model training results. (a) Loss curve; (b) accuracy curve.

Table 6. Training results of seven models.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Param (M)	FLOPs (G)	Training Time (h)
AlexNet	87.15	87.33	87.32	87.33	16.63	0.72	1.12
VGG16	87.44	87.45	87.76	87.61	138.37	15.52	3.08
ResNet34	95.05	95.09	95.07	95.08	11.69	3.61	1.20
ResNet50	96.52	96.52	96.57	96.55	25.56	4.11	2.64
ResNet101	95.68	95.68	95.65	95.66	44.55	7.82	2.62
InceptionresnetV2	96.70	96.72	96.70	96.71	55.80	14.98	5.80
IRCE	98.76	98.77	98.81	98.79	4.24	0.84	1.34

To evaluate the recognition performance of the seven models for wheat diseases, we drew the confusion matrix of the seven models based on the test set, as shown in Figure 12. The models include AlexNet, VGG16, ResNet34, ResNet50, InceptionresnetV2, ResNet101, and IRCE. The actual class (abscissa) is compared with the predicted class (ordinate) in Figure 12 to describe the individual classification performance of each class. In the figure, 0 means healthy wheat, 1 means leaf rust, 2 means powdery mildew, 3 means wheat loose smut, 4 means root rot, 5 means fusarium head blight, and 6 means tan spot. The value of the diagonal of the confusion matrix represents the number of samples of true positive (TP) and true negative (TN), and the larger the value, the better the recognition effect. It can be seen from Figure 12 that all models are the most sensitive to the identification of powdery mildew and smut because the characteristics of powdery mildew and smut are more obvious than those of other diseases, and the models are easier to identify; ResNet50, ResNet101, InceptionresnetV2, and IRCE have the lowest recognition rate of healthy wheat because the image background of healthy wheat is complex, which interferes with the model's discrimination; for the six diseases, the recognition rate of leaf rust and root rot is low. The recognition rate of leaf rust is low because some leaf rust images have no obvious disease characteristics and are considered healthy wheat; the recognition rate of root rot is low because of the influence of image

background, and it is considered to be another disease. The IRCE model has the same recognition accuracy as the classic CNN model for the identification of fusarium head blight. Except for the fusarium head blight, the recognition accuracy of the IRCE model for the other six types of wheat is higher than that of other models.

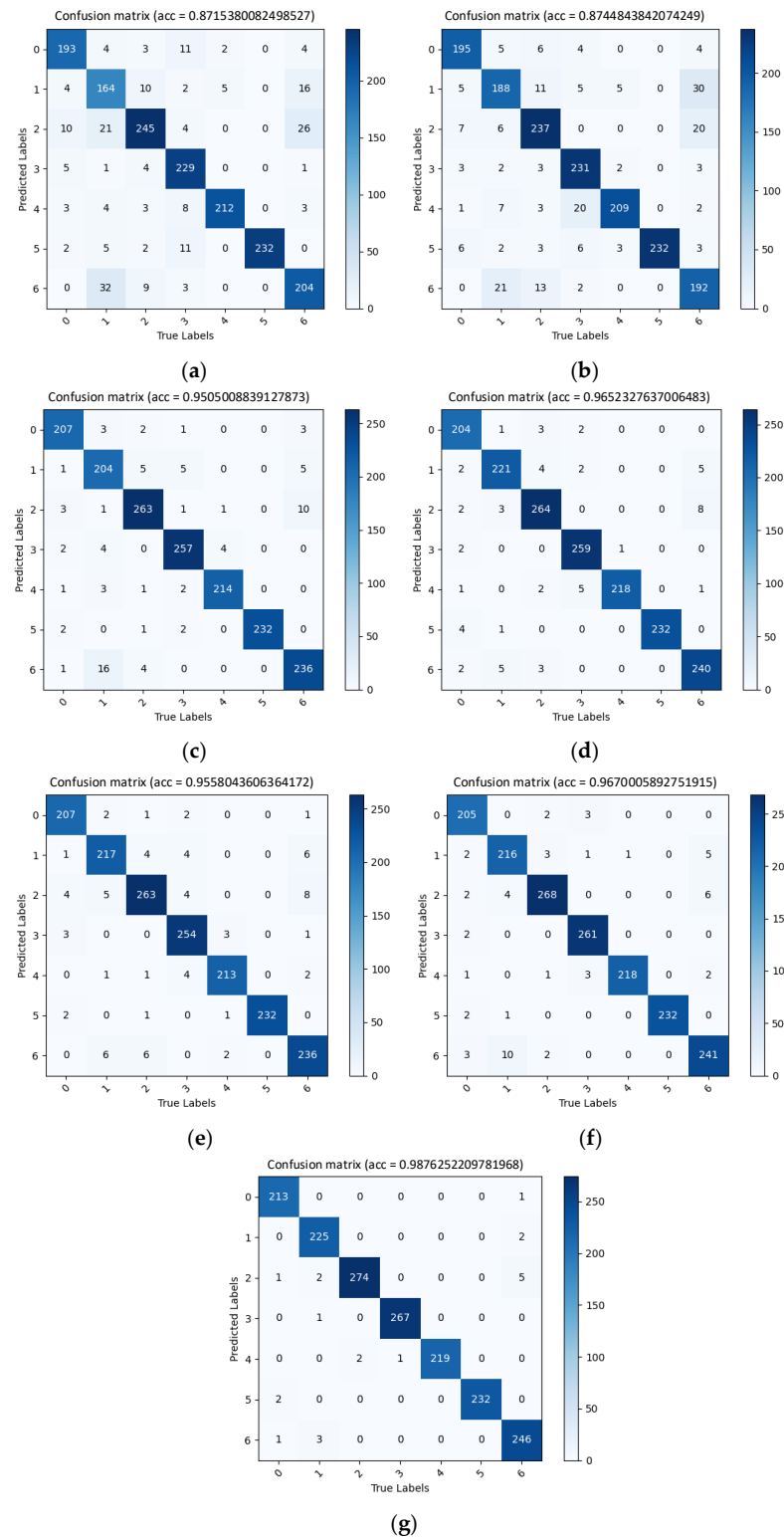


Figure 12. Seven model confusion matrices. (a) AlexNet; (b) VGG16; (c) ResNet34; (d) ResNet50; (e) ResNet101; (f) InceptionresnetV2; (g) IRCE.

3.5. Comparison of the Proposed Model with the Classical Lightweight Model

To test the effectiveness of the proposed lightweight model, we compared the IRCE model with five models: MobileNetV1, MobileNetV2, MobileNetV3-Small, MobileNetV3-Large, and EfficientNetb0. The recognition results are shown in Table 7. The four metrics of accuracy, precision, recall, and F1-score of the IRCE model are optimal. In terms of model recognition accuracy, the IRCE model has the highest accuracy, 2.42–4.35% higher than the MobileNet series, and 1.95% higher than EfficientNetb0. In terms of model parameters, the parameter of the IRCE model is 4.24 M, which is smaller than those of MobileNetV3-Large and EfficientNetb0. However, the IRCE model has a drawback. Among the tested models, the IRCE model has the largest FLOPs, which means that it has a high demand for computing resources. Overall, our model performs well.

Table 7. Lightweight model performance comparison.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Param (M)	FLOPs (G)	Training Time (h)
MobileNetV1	94.41	94.42	94.47	94.49	2.59	0.33	1.52
MobileNetV2	95.23	95.27	95.24	95.26	3.25	0.31	1.44
MobileNetV3-Small	95.34	95.44	95.42	95.43	2.54	0.06	1.36
MobileNetV3-Large	96.75	96.89	96.76	96.82	5.48	0.23	1.45
EfficientNetb0	96.81	96.92	96.87	96.87	5.29	0.40	1.65
IRCE	98.76	98.77	98.81	98.79	4.24	0.84	1.34

3.6. Generalization Ability Test of the Proposed Model

To verify the generalization performance of the model, we evaluated our model on three public datasets: Plant-Village, CGIAR, and the Wheat Leaf Dataset; and we compared it with several state-of-the-art models. Plant-Village: a dataset of 54,306 plant leaf images was used to identify 20 diseases of 14 crops. CGIAR: a dataset of 1486 wheat leaf images was used to identify three wheat diseases. The Wheat Leaf Dataset is a dataset released on the Kaggle website, and it contains a dataset of 412 wheat leaf images. This dataset has three categories: healthy wheat leaves, stripe rust, and septoria. Table 8 shows that our model has a test accuracy of 99.74%, 96.70%, and 96.70% on the three public datasets of Plant-Village, CGIAR, and Wheat Leaf Dataset, respectively. This is enough to prove that our model has a strong generalization performance and achieves the best or comparable results.

Table 8. Experimental comparison of three datasets.

Dataset	Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Plant-Village [35]	LF-CNN [36]	98.93	95.61	97.16	96.65
	ResNet50	92.51	90.34	92.16	91.78
	VGG16	94.13	92.10	93.18	92.35
	MobileNetV2	97.81	97.42	97.93	97.65
	IRCE	99.74	99.71	99.66	99.68
CGIAR [37]	ResNet34	92.11	92.52	92.74	92.65
	VGG19	94.40	94.52	95.31	94.82
	EfficientNetb0	93.90	93.12	93.55	93.38
	InceptionV3	95.72	95.32	95.71	95.42
	IRCE	96.70	96.70	96.70	96.70
Wheat Leaf Dataset	MobileNetV3_Large	92.17	91.72	91.98	91.77
	ResNet34	89.92	89.94	91.01	90.91
	EfficientNetb0	94.61	94.32	95.21	94.62
	InceptionresnetV2	96.32	96.21	96.45	96.32
	IRCE	96.70	96.80	97.10	96.95

4. Discussion

For the three Inception modules introduced by the model, we decided to embed them in the network in a certain order. We considered that Inception-1 was the most basic Inception module, which could be used to extract features of different scales, so we placed it at the front as the basis of the network. Inception-2 was a reduction module that could be used to change the size of the feature map, thereby increasing the depth and receptive field of the network, so we placed it in the middle as a transition of the network. Inception-3 was a more complex Inception module, which could be used to extract larger-scale and finer-grained features, so we placed it at the end as the upper layer of the network. The experiments in Section 3.2 verified the correctness of our embedding methods for the three inception modules.

In deep learning, the residual structure is a method to solve the degradation problem when the number of network layers increases through the identity mapping. The number of residual structures needs to be determined according to the task and dataset to balance the expressiveness and computational efficiency of the network. In this paragraph, we discuss how we tried three ways to introduce the residual structure and compared their results. The first way was to introduce four layers, each layer having two BasicBlocks. This way, although the network became relatively deep, also increased the complexity of the model and the amount of calculation. The second way was to introduce two layers, each layer having two BasicBlocks. For this way, the model complexity was low and the number of parameters was small. The third way was to introduce three layers, each layer having two BasicBlocks. In Table 9, Param, FLOPs, and the training time of method 1 are the highest. Although method 2 consumes fewer resources, the integrity of the model is too low, and method 3 has the highest overall model performance and the least resource consumption. Therefore, we concluded that the third way was the best one for our task.

Table 9. Effect of Different Residual Layers.

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Param (M)	FLOPs (G)	Training Time (h)
Method 1	95.28	95.52	95.38	95.45	13.39	2.04	1.52
Method 2	95.23	95.27	95.24	95.26	4.02	0.73	1.03
Method 3	98.76	98.77	98.81	98.79	4.24	0.84	1.34

In order to improve the model's attention to the disease characteristics of wheat leaves, we tried adding the five attention mechanisms of CBAM, NAM, SE, CA, and ECA to the residual structure. First, taking into account CBAM and NAM with mixed attention mechanisms, we chose CBAM as it had a better performance after comparison. Since CBAM passes features through the channel attention mechanism and spatial attention mechanism in a linear manner, channel features are lost to a certain extent. Therefore, we added another layer of the channel attention mechanism. After comparison, ECA was better than CA and SE. We found that the combination of CBAM and ECA had the best effect on focusing on diseased regions and improved the detection performance of the model.

Finally, we compared our proposed model with classical CNN models and lightweight models in Section 3. The recognition accuracy of the classical CNN model and the lightweight model of wheat diseases is low. This makes it difficult to apply pesticides accurately, which can reduce the yield and quality of wheat, and reduce the efficiency and safety of agricultural production. Our model has the advantages of high precision, a fast speed, and small parameters, and can handle the identification problems of various diseases in complex backgrounds. Our proposed model facilitates the intelligent identification and diagnosis of crop diseases by mobile terminals, discovers the types and proportions of diseases, and provides farmers with timely prevention and control suggestions. However, our method also has limitations. Our model is more accurate than classical lightweight models, but our model is computationally intensive and requires more

computing resources. The training and deployment of models require more hardware resources and power consumption, which increases the cost and environmental burden of agricultural production.

5. Conclusions

This paper proposes a new lightweight wheat disease identification model that can quickly and accurately identify wheat diseases on a mobile terminal in complex farmland backgrounds, and make an important contribution to wheat disease control. We combine the residual module and the inception module to build a new network with a small number of parameters and a low amount of computation and then introduce the CBAM and ECA modules into the residual block, which can perform adaptive feature refinement on the input feature map. In our experiments, the precision, recall, and F1-score of the IRCE model we designed reached 98.76%, 98.77%, 98.81%, and 98.76%, respectively. Compared with classic CNN models, the IRCE model is 11.61% more accurate than AlexNet, 11.32% more accurate than VGG16, and 2.24%~3.71% more accurate than the ResNet series, and the parameter number of the IRCE model is the smallest, only 4.24 M. In addition, we compared the IRCE model with classic lightweight models. The IRCE model is 1.95% more accurate than the highest accuracy EfficientNetb0, and the model training time is the least, only 1.34 h. Through experimental comparison, this paper proves the feasibility of the proposed model. Our method can provide reliable and accurate technical means for wheat disease identification and detection. In the actual detection process, farmers can use mobile devices such as mobile phones to detect diseases and obtain recommendations for spraying pesticides to achieve appropriate levels and increase wheat yields.

However, our method also has limitations. Our model has a higher computational load than classic lightweight models and requires more computing resources. This is exactly what we need to improve in the future work. During the experiment, we found that the image's quality can affect the model's performance. Current crop disease image databases are not perfect, and their image quality is poor. Therefore, it is still necessary to improve crop disease image databases and image quality in the future. In future work, we will work hard on the following aspects. (1) Since there is no fully unified wheat disease database, it is very important to establish a high-quality wheat disease database. (2) The presence of insects can also affect wheat diseases, so insect detection is important. (3) Disease severity detection is also very important, which can guide us to spray the appropriate amount of pesticides to reduce pollution.

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

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The LWDCD2020 dataset is available at <https://medium.com/analytcs-vidhya/wheat-disease-detection-using-keras-48ae78990502>, accessed on 10 January 2022.

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

References

1. Sabenca, C.; Ribeiro, M.; Sousa, T.; Poeta, P.; Bagulho, A.S.; Igrejas, G. Wheat/Gluten-Related Disorders and Gluten-Free Diet Misconceptions: A Review. *Foods* **2021**, *10*, 1765. [CrossRef]
2. Chai, Y.; Senay, S.; Horvath, D.; Pardey, P. Multi-peril pathogen risks to global wheat production: A probabilistic loss and investment assessment. *Front. Plant Sci.* **2022**, *13*, 1034600. [CrossRef]

3. Biel, W.; Jaroszewska, A.; Stankowski, S.; Sobolewska, M.; Kępińska-Pacelik, J. Comparison of yield, chemical composition and farinograph properties of common and ancient wheat grains. *Eur. Food Res. Technol.* **2021**, *247*, 1525–1538. [\[CrossRef\]](#)
4. Yao, F.; Li, Q.; Zeng, R.; Shi, S. Effects of different agricultural treatments on narrowing winter wheat yield gap and nitrogen use efficiency in China. *J. Integr. Agric.* **2021**, *20*, 383–394. [\[CrossRef\]](#)
5. Kloppe, T.; Boshoff, W.; Pretorius, Z.; Lesch, D.; Akin, B.; Morgounov, A.; Shamanin, V.; Kuhnem, P.; Murphy, P.; Cowger, C. Virulence of *Blumeria graminis* f. sp. *tritici* in Brazil, South Africa, Turkey, Russia, and Australia. *Adv. Breed. Wheat Dis. Resist.* **2022**, *13*, 954958. [\[CrossRef\]](#)
6. Mahum, R.; Munir, H.; Mughal, Z.-U.-N.; Awais, M.; Sher Khan, F.; Saqlain, M.; Mahamad, S.; Tlili, I. A novel framework for potato leaf disease detection using an efficient deep learning model. *Hum. Ecol. Risk Assess. Int. J.* **2023**, *29*, 303–326. [\[CrossRef\]](#)
7. Zhang, J.; Pu, R.; Huang, W.; Yuan, L.; Luo, J.; Wang, J. Using in-situ hyperspectral data for detecting and discriminating yellow rust disease from nutrient stresses. *Field Crops Res.* **2012**, *134*, 165–174. [\[CrossRef\]](#)
8. Zhang, D.; Lin, F.; Huang, Y.; Wang, X.; Zhang, L. Detection of Wheat Powdery Mildew by Differentiating Background Factors using Hyperspectral Imaging. *Int. J. Agric. Biol.* **2016**, *18*, 747–756. [\[CrossRef\]](#)
9. Khan, I.H.; Liu, H.; Li, W.; Cao, A.; Wang, X.; Liu, H.; Cheng, T.; Tian, Y.; Zhu, Y.; Cao, W.; et al. Early Detection of Powdery Mildew Disease and Accurate Quantification of Its Severity Using Hyperspectral Images in Wheat. *Remote Sens.* **2021**, *13*, 3612. [\[CrossRef\]](#)
10. Wang, H.; Qin, F.; Liu, Q.; Ruan, L.; Wang, R.; Ma, Z.; Li, X.; Cheng, P.; Wang, H. Identification and Disease Index Inversion of Wheat Stripe Rust and Wheat Leaf Rust Based on Hyperspectral Data at Canopy Level. *J. Spectrosc.* **2015**, *2015*, 651810. [\[CrossRef\]](#)
11. Bao, W.; Zhao, J.; Hu, G.; Zhang, D.; Huang, L.; Liang, D. Identification of wheat leaf diseases and their severity based on elliptical-maximum margin criterion metric learning. *Sustain. Comput. Inform. Syst.* **2021**, *30*, 100526. [\[CrossRef\]](#)
12. Aboneh, T.; Rorissa, A.; Srinivasagan, R.; Gemechu, A. Computer Vision Framework for Wheat Disease Identification and Classification Using Jetson GPU Infrastructure. *Technologies* **2021**, *9*, 47. [\[CrossRef\]](#)
13. Liu, X.; Zhou, S.; Chen, S.; Yi, Z.; Pan, H.; Yao, R. Buckwheat Disease Recognition Based on Convolution Neural Network. *Appl. Sci.* **2022**, *12*, 4795. [\[CrossRef\]](#)
14. Jin, X.; Jie, L.; Wang, S.; Qi, H.; Li, S. Classifying Wheat Hyperspectral Pixels of Healthy Heads and Fusarium Head Blight Disease Using a Deep Neural Network in the Wild Field. *Remote Sens.* **2018**, *10*, 395. [\[CrossRef\]](#)
15. Deng, J.; Lv, X.; Yang, L.; Zhao, B.; Zhou, C.; Yang, Z.; Jiang, J.; Ning, N.; Zhang, J.; Shi, J.; et al. Assessing Macro Disease Index of Wheat Stripe Rust Based on Segformer with Complex Background in the Field. *Sensors* **2022**, *22*, 5676. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Su, W.-H.; Zhang, J.; Yang, C.; Page, R.; Szinyei, T.; Hirsch, C.D.; Steffenson, B.J. Automatic Evaluation of Wheat Resistance to Fusarium Head Blight Using Dual Mask-RCNN Deep Learning Frameworks in Computer Vision. *Remote Sens.* **2020**, *13*, 26. [\[CrossRef\]](#)
17. Shafi, U.; Mumtaz, R.; Qureshi, M.D.M.; Mahmood, Z.; Tanveer, S.K.; Haq, I.U.; Zaidi, S.M.H. Embedded AI for Wheat Yellow Rust Infection Type Classification. *IEEE Access* **2023**, *11*, 23726–23738. [\[CrossRef\]](#)
18. Huang, H.; Deng, J.; Lan, Y.; Yang, A.; Zhang, L.; Wen, S.; Zhang, H.; Zhang, Y.; Deng, Y. Detection of Helminthosporium Leaf Blotch Disease Based on UAV Imagery. *Appl. Sci.* **2019**, *9*, 558. [\[CrossRef\]](#)
19. Pan, Q.; Gao, M.; Wu, P.; Yan, J.; Li, S. A Deep-Learning-Based Approach for Wheat Yellow Rust Disease Recognition from Unmanned Aerial Vehicle Images. *Sensors* **2021**, *21*, 6540. [\[CrossRef\]](#)
20. Mi, Z.; Zhang, X.; Su, J.; Han, D.; Su, B. Wheat stripe rust grading by deep learning with attention mechanism and images from mobile devices. *Front. Plant Sci.* **2020**, *11*, 558126. [\[CrossRef\]](#)
21. Bao, W.; Yang, X.; Liang, D.; Hu, G.; Yang, X. Lightweight convolutional neural network model for field wheat ear disease identification. *Comput. Electron. Agric.* **2021**, *189*, 106367. [\[CrossRef\]](#)
22. Goyal, L.; Sharma, C.M.; Singh, A.; Singh, P.K. Leaf and spike wheat disease detection & classification using an improved deep convolutional architecture. *Inform. Med. Unlocked* **2021**, *25*, 100642.
23. Zeng, W.; Li, M. Crop leaf disease recognition based on Self-Attention convolutional neural network. *Comput. Electron. Agric.* **2020**, *172*, 105341. [\[CrossRef\]](#)
24. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
25. 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.
26. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
27. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.
28. Gu, R.; Wang, G.; Song, T.; Huang, R.; Aertsen, M.; Deprest, J.; Ourselin, S.; Vercauteren, T.; Zhang, S. CA-Net: Comprehensive attention convolutional neural networks for explainable medical image segmentation. *IEEE Trans. Med. Imaging* **2020**, *40*, 699–711. [\[CrossRef\]](#)

29. Rab Ratul, M.A.; Tavakol Elahi, M.; Yuan, K.; Lee, W. RAM-Net: A Residual Attention MobileNet to Detect COVID-19 Cases from Chest X-ray Images. In Proceedings of the 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 14–17 December 2020; pp. 195–200.
30. 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; pp. 3–19.
31. Liu, Y.; Shao, Z.; Teng, Y.; Hoffmann, N. NAM: Normalization-based Attention Module. *arXiv* **2021**, arXiv:2111.12419.
32. Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–436.
33. Heo, B.; Chun, S.; Oh, S.J.; Han, D.; Yun, S.; Kim, G.; Uh, Y.; Ha, J.-W. Adamp: Slowing down the slowdown for momentum optimizers on scale-invariant weights. *arXiv* **2020**, arXiv:2006.08217.
34. Mehta, S.; Paunwala, C.; Vaidya, B. CNN based traffic sign classification using adam optimizer. In Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 15–17 May 2019; pp. 1293–1298.
35. 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.
36. Gokulnath, B. Identifying and classifying plant disease using resilient LF-CNN. *Ecol. Inform.* **2021**, *63*, 101283.
37. Kukreja, V.; Kumar, D. Automatic classification of wheat rust diseases using deep convolutional neural networks. In Proceedings of the 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 3–4 September 2021; pp. 1–6.

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.