Image recognition for crop diseases using a novel multi-attention module

Lei Chen[1], Yuan Yuan[1*], Haiyan He[2]

(1. *Institute of Intelligent Machines, Hefei Institutes of Physical Science (HFIPS), Chinese Academy of Sciences, Hefei 230031, China*; 2. *Department of Science Island Branch, University of Science and Technology of China, Hefei 230026, China*)

Abstract: Deep convolution neural networks constitute a breakthrough in computer vision. Based on this, the Convolutional Neural Network (CNN) models offer enormous potential for crop disease classification. However, significant training data are required to realize their potential. In the case of crop disease image recognition, especially with complex backgrounds, it is sometimes difficult to acquire adequately labeled large datasets. This research proposed a solution to this problem that integrates multi-attention modules, i.e., channel and position block (CPB) module. Given an intermediate feature map, the CPB module can infer attention maps in parallel with the channel and position. The attention maps can then be multiplied to form input feature maps for adaptive feature refinement. This provides a simple yet effective intermediate attention structure for CNNs. The module is also lightweight and produces little overhead. Some experiments on cucumber and rice image datasets with complex backgrounds were conducted to validate the effectiveness of the CPB module. The experiments included different module locations and class activation map display characteristics. The classification accuracy reached 96.67% on the cucumber disease image dataset and 95.29% on the rice disease image dataset. The results show that the CPB module can effectively classify crop disease images with complex backgrounds, even on small-scale datasets, which providing a reference for crop disease image recognition method under complex background conditions in the field.

Keywords: image recognition, crop disease, multi-attention module, deep learning, small sample

DOI: 10.25165/j.ijabe.20251801.8136

Citation: Chen L, Yuan Y, He H Y. Image recognition for crop diseases using a novel multi-attention module. Int J Agric & Biol Eng, 2025; 18(1): 238–244.

1 Introduction

Traditionally, it has been necessary to turn to human experts to diagnose plant anomalies caused by diseases, pests, nutritional deficiencies, or extreme weather. However, this is expensive, time-consuming, and frequently impractical[1]. At the same time, identifying crop diseases can be a challenge for ordinary farmers because there are too many different kinds of diseases with different causes[2], which often exceed their limited experience. This can result in crops not being treated in time and environmental harm through the misuse of drugs[3]. Therefore, the timely and correct identification of crop diseases plays a vital role in ensuring enhanced and sustainable agricultural productivity.

Computer vision technology based on machine learning, deep learning, transfer learning, and attention mechanisms provides the prospect for low-cost and contact-free automatic identification of crop diseases[4]. Over the past few years, various vision-based machine-learning techniques have come to be widely used in agricultural disease classification. Most are based on hand-engineered features and intelligent algorithms. However, while these methods have achieved good classification results, using hand-crafted features generates an excessive waste of human and material resources[5]. Due to the complexity of agricultural diseases, some features such as color, texture, shape, etc. have inter-class similarity and intra-class differences, making it difficult to represent information about diseases solely based on specific features, which can easily have adverse effects on subsequent modeling.

Out of the options mentioned above, deep learning might seem to offer one of the most promising ways forward. Deep learning was first developed by Hinton et al.[6] in 2006. It involves using an algorithm based on artificial neural networks, which provide the architecture for characterizing and learning data. One kind of deep learning approach, deep Convolutional Neural Networks (CNNs), has proven tremendously successful for crop classification tasks[7]. Here, a convolution-based method is used to extract image features, resulting in a high-level fusion of semantics and deep feature extraction[8]. Mohanty et al.[9] have effectively used deep-learning techniques to diagnose plant diseases. They used a public dataset (PlantVillage) that contains 54 306 images of diseased and healthy plant leaves. Their work is based on two popular architectures, AlexNet[10] and GoogLeNet[11]. These were originally designed for the Scale Visual Recognition Challenge (ILSVRC) for the ImageNet[12] dataset. Their approach delivered good classification results for 14 crop species and 26 diseases in the Plant Village dataset. However, deep learning is a supervised learning method[13]. As a result, especially in the studies of crop disease image recognition, the modeling quality relies heavily on large batches of labeled training samples[14]. The scale of agricultural disease image data is often insufficient to directly support deep learning models for training and often requires some data augmentation methods[15]. In addition, in agricultural contexts, the parameters for deep learning models can become particularly complex because of the diversity of crops and their diseases[16].

Unlike traditional machine learning and deep learning, transfer learning[17] provides a way of transferring learned knowledge to a target domain to help train new models. Thus, even if the target

Received date: 2023-01-05 **Accepted date:** 2024-05-08

Biographies: Lei Chen, PhD, Associate Professor, research interest: machine learning, new computational model under big data, and their applications in computer vision and natural language processing, Email: chenlei@iim.ac.cn; **Haiyan He**, Master candidate, research interest: computer vision, Email: 40901287@qq.com.

***Corresponding author: Yuan Yuan**, PhD, Associate Professor, research interest: smart agriculture, machine learning and their applications in computer vision and natural language processing, The Institute of Intelligent Machines, Mailbox 1130, Shushanghu Road, Hefei 230031, Anhui, China. Tel: +86-551-65591145, Email: yuanyuan@iim.ac.cn.

domain lacks large-scale labeled datasets, it can still be effectively modeled. This approach is especially effective when training machine learning models with very limited training data[18]. Fang et al.[19] optimized a transfer learning method called Tradabost and developed an example-based migration learning system to deal with the lack of labeled training samples for agricultural disease image recognition. Shi et al.[20] used transfer learning to train a VGG net model that can detect tomato diseases and pests. This achieved an average classification accuracy of 89%. Jiang et al.[21] improved the VGG16 model based on the idea of multi-task learning and then used the pre-training model on ImageNET for transfer learning and alternating learning, achieving an accuracy of 97.22% for rice leaf diseases and 98.75% for wheat leaf diseases. Liu et al.[22] proposed an improved network based on MobileNet V3 to identify cucumber diseases from leaf images in natural scenes, and selected PlantVillage and Apple Disease datasets for transfer learning, achieving accuracy rates of 99.0% and 98.1%, respectively. Unfortunately, while transfer learning can solve problems with limited datasets, its modeling quality can be affected by several factors, such as the quality of the dataset, the choice of the model prototype, and negative transfer. Thus, the final results do not always meet expectations, and there are still many problems in transfer learning that need further research.

An alternative option is to use attention mechanisms. Visual attention is a signal processing mechanism associated with human vision, where people quickly scan a global image to acquire a target area that needs to be focused. The basic idea of attention mechanisms in computer vision is to adopt a similar method, where irrelevant information is ignored and important information is focused instead. This can be implemented in ways that can be roughly divided into soft attention and hard attention[23]. Typical examples of soft attention are spatial transformer networks[24], residual attention networks[25], and two-level attention[26]. The specific attention mechanism can be based on calculus and can be trained by backpropagation. Hard attention involves predicting the areas of concern and usually uses reinforcement learning for the training process. These different attention mechanisms have only been tested on large public datasets, and their research and validation on small-sample agricultural datasets are relatively limited. Some recent works, such as Feng et al.[27]and Zheng et al.[28], have conducted preliminary studies, indicating that this method has good potential research value.

Given the above issues, this study adopted a different approach. It began by developing a dataset of crop disease images that includes four cucumber diseases and four rice diseases. These were captured in real-life agricultural conditions containing a lot of noise, e.g., cluttered field backdrops and uneven illumination intensities. With the limited manpower available, only about one thousand pictures of each kind of disease were able to be acquired. If large-scale deep CNNs were applied to the dataset rather than the popular large datasets they usually work with (i.e., 10 000+ images), they would quickly start overfitting the training data. To deal with this, a novel multi-attention net structure with a channel and position block (CPB) module has been developed, which combines channel and spatial information that can express the features of crop diseases. Different experiments were designed to verify the CPB model, the impact of different positions on the accuracy of the CPB model, and the robustness of the CPB model under small-sample conditions. Three main contributions of this paper are as follows:

1) A mechanism removes dependence on the scale of the annotated data during the deep learning modeling process without needing the help of auxiliary datasets like in transfer learning;

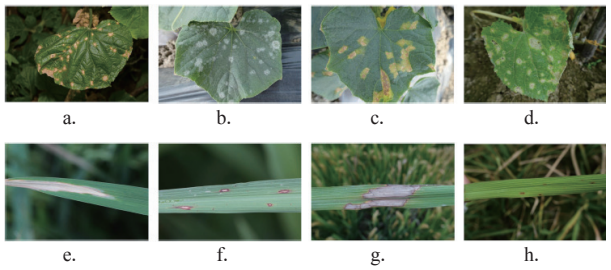
2) An experimental approach to collecting crop disease images in the field that does not need complex operations such as background removal or spot segmentation;

3) A CPB module enables neural networks to achieve better classification accuracy than some popular deep-learning models.

2 Materials and methods

2.1 Image acquisition

The dataset used here is part of the Image Database for Agricultural Diseases and Pests (IDADP)[29], which is collected by the researchers of this study. The images were captured in a real field environment using a digital single-lens reflex camera (Canon EOS6D, Japan). Some requirements need to be met during image acquisition. For example, the light needs to be uniform, the plane of the crop organ needs to be perpendicular to the shooting angle, etc. To experimentally test the approach of this study, 4210 cucumber disease images and 3081 rice disease images were used, including four cucumber diseases and four rice diseases. For the convenience of subsequent data processing, each cucumber disease and rice disease was put in one folder with the same class label. The resolution of all images is 3000×2000 pixels. Figure 1 shows some examples of crop diseases in the original target dataset. It can be seen that most of the crop disease images in this study have complex backgrounds.



Note: a. Cucumber corynespora target leaf spot, b. Cucumber powdery mildew, c. Cucumber downy mildew, d. Cucumber anthracnose, e. Bacterial rice blight, f. Rice blast, g. Rice sheath blight, h. Rice brown spot

Figure 1 Images of eight kinds of crop diseases

Table 1 lists the number of samples for the cucumber diseases, and Table 2 lists the number for the rice diseases. It can be seen that, compared to other deep learning datasets, the number of samples in the dataset was small and the data distribution was unbalanced. In each of the experiments described in this study, 90% of the disease images were used for training and the remaining 10% were used for testing. The task was to classify crop disease images in the test set.

Table 1 Number of cucumber disease samples

Class	Number
Cucumber corynespora target leaf spot	828
Cucumber powdery mildew	1011
Cucumber downy mildew	1089
Cucumber anthracnose	1282

Table 2 Number of rice disease samples

Class	Number
Bacterial blight of rice	684
Rice blast	958
Rice sheath blight	611
Rice brown spot	828

2.2 Image preprocessing

The flow of image preprocessing is shown in Figure 2. First, the image was resized to 448×448 pixels and randomly flipped horizontally to increase the diversity of the features. Random horizontal flipping means a 50% chance of flipping and a 50% chance of not flipping. The dataset itself is not expanded when samples are selected randomly for flipping in the input phase. The purpose of center clipping is to cut the picture with the specified length and width starting from the center of the picture, to obtain the original picture's central part. However, the disease not only occurs in the central part of the leaves but also some lesions are distributed on the edge of the leaves. As a result, rather than using central clipping, random clipping was employed. This ensured that some leaf edge information was incorporated into the input features. The cropping ratio was also very important for data processing. The cropping ratio needs to remove a lot of background noise and retain most of the features for the network to recognize. By decreasing the size of the ratio by 5 percentage points at a time, it was found that the effect was best when the image was randomly cut into 400× 400 pixels.

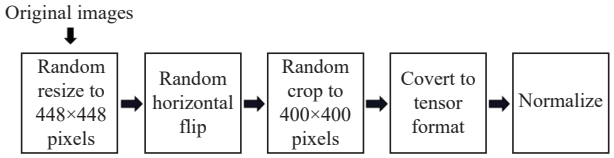


Figure 2 Flowchart of image preprocessing

In machine learning or deep learning, the loss calculation of most models needs to assume that all data features are zero-mean and have the same order of variance. In this way, when calculating loss, all feature attributes can be processed uniformly. The image was normalized, that is, the feature attributes of the data were subtracted from the mean, divided by the variance, and converted to a standard normal distribution with a mean of 0 and a variance of 1 so that the data has a reasonably regular distribution and greater generality. At the same time, data normalization can also improve the convergence speed of the network.

2.3 Multi-attention module

As shown in Figure 3, the proposed CPB module has two parallel sub-modules: one for max-pooling channel attention, and the other for position attention. The CPB module adaptively refines the intermediate feature map in the residual convolution block of the deep network.

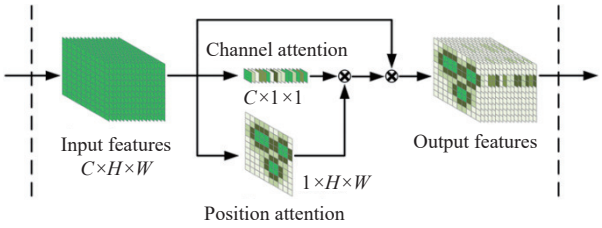


Figure 3 Overview of the CPB module

In the CPB module, given an intermediate feature map $F \in R^{C\times H\times W}$ as input, where R is the feature matrix; C, H, and W are the number of channels, the height of the feature map, and the width of the feature map, respectively; and the CPB parallel structure infers a one-dimensional max-pooling channel attention map $M_c \in R^{C\times 1\times 1}$, and a two-dimensional position attention map $M_p \in R^{1\times H\times W}$. The complete attention mechanism process can be summarized as follows:

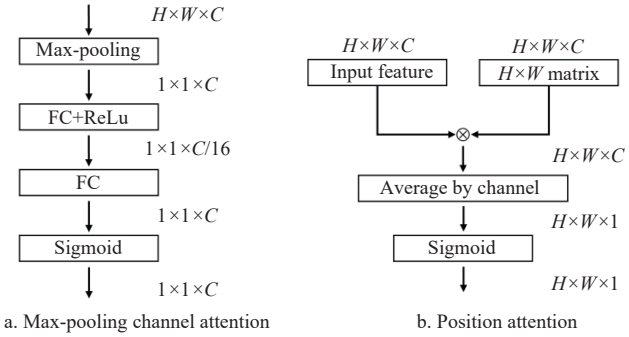
$$F_1 = M_c(F) \tag{1}$$

$$F_2 = M_p(F) \tag{2}$$

$$F' = F_1 \otimes F_2 \tag{3}$$

$$F'' = F' \otimes F \tag{4}$$

where, ⊗ denotes element-wise multiplication, F_1 is the result of the input features after the channel attention mechanism operation, F_2 is the result of the input features after the spatial attention mechanism operation, and F' is the result of the bitwise multiplication of F_1 and F_2. A broadcast mechanism is used during multiplication to ensure the attention values are functional. The channel attention values are copied from the spatial dimension, while the position attention values are copied from the channel dimension. F'' is the final extracted feature output. Figure 4 shows how each attention map is calculated. The details regarding each attention module are given below.



a. Max-pooling channel attention b. Position attention

Figure 4 Calculation processes of the max-pooling channel attention and the position attention

As shown in Figure 4, the channel sub-module produces max-pooling outputs. These outputs pass through two fully connected layers. The purpose of the pooling is to maintain some invariance (rotation, translation, scaling, etc.). It results in a reduced number of features and parameters, thus limiting the risk of overfitting. There are two commonly used pooling methods: mean pooling and max pooling. Mean pooling involves averaging the feature points in a neighborhood. Max-pooling involves taking the largest feature point in a neighborhood. Errors in feature extraction mainly arise in two ways:

1) The estimation variance may increase due to the limited neighborhood size;

2) Errors in the convolution layer parameters may result in a mean shift in the estimation.

Generally speaking, mean-pooling addresses the first error and retains more background information, while max-pooling addresses the second error and retains more texture information. The cropped dataset contained a lot of background noise, but the texture information was necessary for disease detection. This study therefore used max-pooling to optimize the network structure.

In the channel attention block, the inter-channel relationship between features was exploited to generate a channel attention map. Each channel of the initial feature map was treated as a feature detector[30]. The importance of each feature channel was automatically obtained by learning the loss. Then, using this as a weight, useful features can be enhanced and irrelevant features (for the current task) suppressed. In this way, it was possible to achieve adaptive feature channel calibration. The channel's attention focused on what is important in an image.

Max-pooling was used to aggregate the spatial information. This enabled more important clues about the distinctive features of an object to be collected. The detailed operations involved in doing this can described as follows: First, the spatial information in the feature map was aggregated by using max-pooling operations. This generated a single spatial context descriptor: $F^c_{\max}$. The descriptor was then forwarded to two fully connected layers to produce the channel attention map, $M_c \in R^{C\times 1\times 1}$. This encoded what needs to be emphasized or suppressed. The channel attention can be computed using the following formula:

$$m_{\text{channel}}(F') = \sigma\left(W_1\left(W_0\left(F^c_{\max}\right)\right)\right) \tag{5}$$

where, σ denotes the sigmoid function, W_0 is the weight parameter for the first fully connected layer, and W_1 is the weight parameter for the second fully connected layer.

A position attention map was produced by integrating the inter-spatial relationships between features. Unlike channel attention, position attention focuses on "where" the most useful information is in a given input image. This can then be combined with the channel attention. In order to compute the position attention, the input feature matrix was multiplied by a given matrix of the same size, bit by bit. To highlight informationally useful areas, it was best to apply pooling operations along the channel axis[31]. An average pooling operation was therefore applied in this way to generate the effective feature descriptors, $M_p \in R^{H\times W\times C}$. When the input matrix was multiplied by a given matrix of the same size, it generated a new feature: $F_p \in R^{H\times W\times C}$. The channel information of the new feature map was then converged by using average pooling operations to produce a 2D spatial attention map. In short, the position attention can be computed as follows:

$$m_{\text{position}}(F) = \sigma(F^p) \tag{6}$$

where, σ denotes the sigmoid function; F^p represents the result of multiplying the input matrix by a given matrix of the same size.

When considering the basic network framework, ResNet has five different depth structures in the official code of PyTorch, and the depths are 18, 34, 50, 101, and 152, respectively. The depth of the network refers to the number of layers that need to update parameters through training, such as convolutional layers, fully connected layers, etc. According to the type of Block, these five types of ResNet can be divided into two categories: 1) based on BasicBlock, the shallow network ResNet18, 34 are composed of BasicBlock, and 2) based on Bottleneck, the deep network ResNet50, 101, 152 are composed of Bottleneck built. The BasicBlock contains a residual branch and a short-cut branch, which are used to transmit low-level information so that the network can be trained very deeply. Bottleneck uses a 1×1 convolutional layer to process inputs with a larger number of channels with a smaller amount of parameters in a deeper network. The ResNet18 and ResNet50 structures are the classic structures in the ResNet series, so these two network models were chosen for experimental research.

Blocks are equivalent to building blocks; each layer is constructed from several blocks, and the entire network is composed of layers. Each ResNet network has four layers. Each layer of a convolution neural network is an abstract representation of an image. The higher the level, the more abstract the feature is. A low-level convolutional layer aims to identify low-level features such as curves and edges. A high-level convolutional layer then extracts higher-level features, such as semicircles (a combination of curves and edges) or rectangles (a combination of four edges). This study needed to determine in which layer it would make the most sense to add the CPB module. Figure 5 shows a schematic diagram of its positional placement. The results regarding the block location's influence on model performance are in Section 3.3.

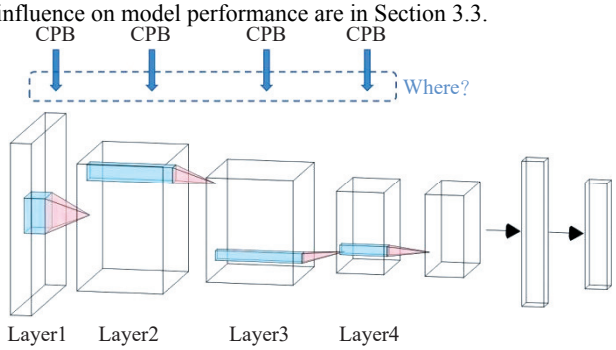


Figure 5 Exploration of where the CPB module is added to the ResNet network

The model structure resulting from using ResNet18 as a baseline is listed in Table 3, and the model structure resulting from using ResNet50 as a baseline is listed in Table 4, where the value of "FC+softmax" indicates the number of classifications. The output size refers to the changed image size. The content in CPB-18 and CPB-50 indicates the size of the convolution kernels and the number of output channel convolution layers. FC is the abbreviation of a fully connected layer. The output of the convolutional layer represents the high-level features of the data. When the output can be flattened and connected to the output layer, adding a fully connected layer can often learn these nonlinear combined features

Table 3 CPB-18 network architecture

Layer	Output Size	CPB-18
Conv1	200×200	7×7 stride 2
Max pooling	100×100	3×3 stride 2
Residual unit +CPB	100×100	$\begin{bmatrix} 1\times1,\ 64 \\ 3\times3,\ 64 \end{bmatrix}\times2$
Residual unit + CPB	50×50	$\begin{bmatrix} 1\times1,\ 128 \\ 3\times3,\ 128 \end{bmatrix}\times2$
Residual unit	25×25	$\begin{bmatrix} 1\times1,\ 256 \\ 3\times3,\ 256 \end{bmatrix}\times2$
Residual unit	13×13	$\begin{bmatrix} 1\times1,\ 512 \\ 3\times3,\ 512 \end{bmatrix}\times2$
Adaptive average pooling	1×1	7×7 stride 1
FC+ softmax	4	

Table 4 CPB-50 network architecture

Layer	Output Size	CPB-50
Conv1	200×200	7×7 stride 2
Max pooling	100×100	3×3 stride 2
Residual unit +CPB	100×100	$\begin{bmatrix} 1\times1,\ 64 \\ 3\times3,\ 64 \\ 1\times1,\ 256 \end{bmatrix}\times3$
Residual unit + CPB	50×50	$\begin{bmatrix} 1\times1,\ 128 \\ 3\times3,\ 128 \\ 1\times1,\ 512 \end{bmatrix}\times4$
Residual unit	25×25	$\begin{bmatrix} 1\times1,\ 256 \\ 3\times3,\ 256 \\ 1\times1,\ 1024 \end{bmatrix}\times6$
Residual unit	13×13	$\begin{bmatrix} 1\times1,\ 512 \\ 3\times3,\ 512 \\ 1\times1,\ 2048 \end{bmatrix}\times3$
Adaptive average pooling	1×1	7×7 stride 1
FC + softmax	4	

easily. ReLU is the most commonly used activation function in deep learning model training. ReLU will make the output value of some neurons 0, making the network sparse, reducing the interdependence of parameters, and alleviating the overfitting problem.

3 Results and discussion

3.1 Experimental settings

The CPB was evaluated by using it to classify images in the cucumber and rice datasets. All the evaluated networks were reproduced in the PyTorch framework to facilitate comparison. The results for all of the experiments are reported below.

To evaluate the effectiveness of the final module, extensive ablation experiments were first performed. Then it was verified that the CPB could outperform all the baselines, thus demonstrating its general applicability to the dataset across different architectures. Tables 3 and 4 provide the details of having the CPB integrated with ResNet18 and ResNet50 as examples. The CPB was put in the first two residual modules of ResNet18 and ResNet50, and then the final classification was carried out.

To compare the effects of the CPB, the hyper-parameters were standardized in all of the experiments. This study fully refers to similar model designs in related literature and a series of experiments based on this agricultural disease dataset and finally unifies the hyper-parameters. The hyper-parameters are described in Table 5. The model adopted the stochastic gradient descent (SGD) optimization algorithm. The learning rate determined the update speed of the weights. A high learning rate will cause the model to skip the optimal solution, resulting in large shocks, increasing the loss value, and reducing the accuracy. Combined with the data parameter settings previously saved in the laboratory, it was set to 0.001 to ensure accurate results within an acceptable time. Dropout rate can be used to prevent overfitting. According to the empirical value, it was set to 0.5. When dropout was set to 0.5, the randomly generated network structure was the most and the generalization ability was the best. The cross-entropy loss function was often used in the PyTorch deep learning framework for classification training. Cross-entropy was mainly used to determine how close the actual output was to the expected output. It was very useful when faced with a training set with unbalanced samples. Accuracy was defined as the ratio of all correctly classified samples to the total number of samples, and this was used to evaluate the model's performance.

Table 5 Hyper-parameters of the experiments

Hyper-parameter	Value
Optimization algorithm	SGD
Learning rate	0.001
Momentum	0.9
Weight decay	1e−4
Learning rate adjustment	Multi Step LR ([10,20,30])
Epochs	40
Batch size	10
Dropout	0.5
Loss function	Cross entropy loss function
Evaluation indicators	Accuracy

3.2 CAM-based network visualization

Class activation mapping (CAM)[32] is a tool that can help with visualizing CNNs by revealing their focal points. Therefore, CAM visualization was used for the final convolution outputs. In Figure 6 and Figure 7, the busier regions are visible. The various classes' discriminative regions of the images were also highlighted. Observing which part of the image the neural network thought was effective for classification prediction made it easier to assess whether the module had played a useful role. The visualization results were compared for a CBAM-integrated network in ResNet50 (ResNet50+CBAM)[33] with those for a SENET-integrated network (ResNet50+SENET)[34] and a CPB-integrated network (ResNet50+CPB).

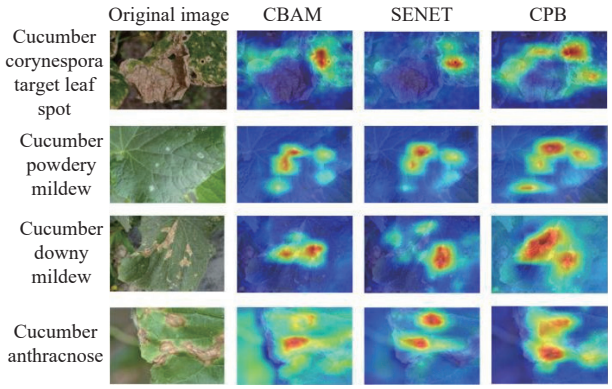


Figure 6 CAM visualization results of cucumber diseases under different attention mechanisms

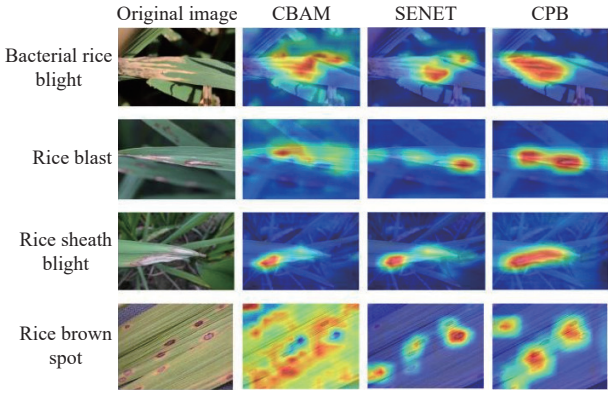


Figure 7 CAM visualization results of rice diseases under different attention mechanisms

Four images were selected that had been correctly classified by the three methods mentioned above (i.e., CBAM net, SENET, and CPB) from the disease validation dataset. Each category contained two pictures. The red area in each image is the part containing characteristic information that was used by the network. The blue area is the part that was relatively unimportant for output prediction. Compared with the other two attention networks, it can be seen from the features that the CPB was better able to ignore the background while extracting more essential information. This means that the CPB module makes more effective use of the disease characteristics in the data.

3.3 Location of the CPB module

Different CPB module positions produced different effects during the experiments. CNN layers are not black boxes, and each layer has its specific function. Shallow CNN filters detect the initial features such as edge and color. The filters in the middle layer can identify various texture patterns. When reaching the deeper layers, the filters can detect patterns composed of basic features. Taking the ResNet18 network as an example, the most suitable position of the CPB module was explored in a neural network. The CPB module was experimented with placing in the first two layers, the last two layers, and all layers of ResNet18 to discover the most appropriate position for prediction. This experiment was conducted on the

cucumber dataset. The accuracy of the test set is shown in Figure 8. The formula for calculating the accuracy is as follows:

$$\text{Accuracy} = \frac{\text{Number of images classified accurately}}{\text{Total number of images in the test set}} \quad (7)$$

The accuracies of adding CPB module in the first two layers, the last two layers, and all layers of ResNet18 are 96.15%, 94.36%, and 91.54%, respectively. The experimental results show that the module can extract enough feature information when placed in the first two layers. Still, it leads to overfitting and poor generalizability when placed in other positions. This means that the CPB module uses a feature's edge and color information, but that more abstract information has an inhibitory effect.

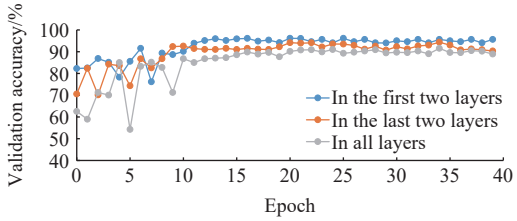


Figure 8 Accuracy comparison of adding CPB modules in different positions in ResNet18 on the cucumber dataset

3.4 Influence of different attention structures

To refine the design of the module, the impact of having different numbers of attention channels was also compared. This experiment was conducted on the cucumber dataset with ResNet18 as the basic model. The structure of the model is shown in Figure 9. The input features were processed through a 3×3 convolution layer, then an extra attention channel was extracted. Figure 10 shows the effect of increasing the number of channels. It can be seen from Figure 10 that increasing the number of attention channels does not, by itself, make the module more effective. Thus, when using an attention mechanism to enhance feature extraction, the appropriate number of channels is important, and adding too many channels does not bring better results.

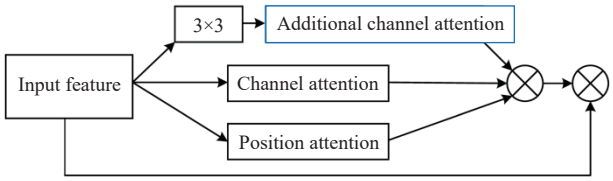


Figure 9 Adding channel in the original CPB module

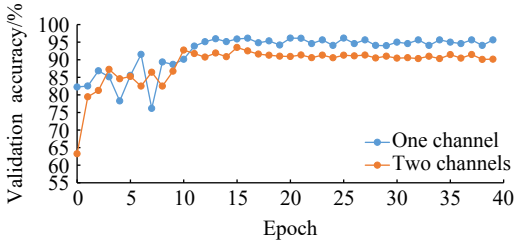


Figure 10 Experimental results of ResNet18 adding additional channels on the cucumber dataset

3.5 Results of the disease recognition experiments

To further assess the impact of the CPB module, a series of ablation experiments was conducted. ResNet50 and ResNet18 were used as the basic models, and the CPB module, a SENET attention mechanism, or a CBAM attention mechanism was added respectively. Tables 6 and 7 list the outcome of these experiments in terms of validation accuracy. Table 6 lists the recognition accuracy of all models on the cucumber test dataset, and Table 7 lists the recognition accuracy of all models on the rice test dataset. The CPB model of this study produced the best results overall.

Table 6 Validation accuracy of the cucumber test dataset

Model	Accuracy/%
ResNet50+CPB	96.67
ResNet18+CPB	96.15
ResNet18+SENET	95.90
ResNet50+SENET	95.90
ResNet50+CBAM	95.13
ResNet18+CBAM	94.87
ResNet50	94.87
ResNet18	87.95

Table 7 Validation accuracy of the rice test dataset

Model	Accuracy/%
ResNet50+CPB	95.29
ResNet50+CBAM	94.41
ResNet18+CPB	93.82
ResNet18+CBAM	93.24
ResNet50+SENET	92.94
ResNet18+SENET	92.82
ResNet50	92.35
ResNet18	91.18

Besides, to better evaluate the performance of CPB under the condition of small samples, six datasets were constructed by randomly selecting from the rice disease datasets, namely, Dataset600, Dataset500, Dataset400, Dataset300, Dataset200, and Dataset100, where the number of rice disease images in each dataset is shown in Table 8. Similarly, in each dataset, 90% of the images were used for training and the remaining 10% were used for testing.

Table 8 Datasets of different sizes

Dataset	Rice blast	Rice sheath blight	Rice brown spot	Rice bacterial blight
Dataset600	600	600	600	600
Dataset500	500	500	500	500
Dataset400	400	400	400	400
Dataset300	300	300	300	300
Dataset200	200	200	200	200
Dataset100	100	100	100	100

The model ResNet50+CPB with the best performance in the above experiments was used. The results are shown in Table 9. It can be seen that the proposed method can still maintain good classification performance even as the dataset size gradually decreases.

Table 9 Experimental results on datasets of different sizes

Dataset	Accuracy/%
Dataset600	95.00
Dataset500	95.00
Dataset400	93.75
Dataset300	93.33
Dataset200	93.75
Dataset100	95.00

4 Conclusions

This study proposed a novel multi-attention CPB module that can improve the representational power of attention-based networks. When applied to crop disease images collected in the field, it can obviate the need for complex operations such as background removal and spot segmentation.

The CPB-based method was adopted for feature refinement and achieved significant performance improvements without additional

overhead. It is recommended to use max-pooled features for the module's channel attention and position attention components. Its performance can be further improved by connecting the two attention components in parallel. Due to the small number of image samples available, the CPB module was put in the first two residual blocks of the network. This enabled it to extract the required features while avoiding overfitting. The final CPB learned where and what to focus on in a disease image to extract intermediate features effectively.

To verify the capability of the CPB module, extensive experiments were conducted with various models using cucumber and rice disease datasets with significant differences in disease characteristics. The results confirmed that the model could outperform other attention-based approaches.

This work will continue to be pursued along the following lines: 1) the testing of more agricultural datasets; 2) ongoing optimization and improvement of the CPB module, so that it can be loaded into other basic classification models for testing; and 3) integrating with multimodal technology to further enhance the performance of attention mechanisms.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (Grants No. 32271981, No. 32071901) and the database in the National Basic Science Data Center (No. NBSDC-DB-20).

[References]

[1] Ngugi L, Abelwahab M, Abo-Zahhad M. Recent advances in image processing techniques for automated leaf pest and disease recognition-A review. Information Processing in Agriculture, 2021; 8(1): 27–51.

[2] Kasinathan T, Singaraju D, Uyyala S R. Insect classification and detection in field crops using modern machine learning techniques. Information Processing in Agriculture, 2021; 8(3): 446–457.

[3] Chen J D, Chen J X, Zhang D F, Nanehkaran Y A, Sun Y D. A cognitive vision method for the detection of plant disease images. Machine Vision and Applications, 2021; 32(1): 31.

[4] Wells III W M. Medical image analysis - past, present, and future. Medical Image Analysis, 2016; 33: 4–6.

[5] Wan J, Wang D Y, Hoi S C H, Wu P C, Zhu J K, Zhang Y D, et al. Deep learning for content-based image retrieval: A comprehensive study. In: Proceedings of the 22nd ACM International Conference on Multimedia, 2014; pp.157–166. doi: 10.1145/2647868.2654948.

[6] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. Science, 2006; 313(5786): 504–507.

[7] Picon A, Alvarez-Gila A, Seitz M, Ortiz-Barredo A, Echazarra J, Johannes A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture, 2019; 161: 280–290.

[8] Shi Z F, Li H, Cao Q J, Ren H Z, Fan B Y. An image mosaic method based on convolutional neural network semantic features extraction. Journal of Signal Processing Systems, 2020; 92(4): 435–444.

[9] Mohanty S P, Hughes D P, Salathé M. Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 2016; 7: 1419.

[10] Arya S, Singh R. A comparative study of CNN and AlexNet for detection of disease in potato and mango leaf. In: Proceedings of 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad: IEEE, 2019; pp.1–6. doi: 10.1109/ICICT46931.2019.8977648.

[11] Szegedy C, Liu W, Jia Y Q, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015; pp.1–9. doi: 10.1109/CVPR.2015.7298594.

[12] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 2015; 115(3): 211–252.

[13] Yuan Y, Chen L, Wu H R, Li L. Advanced agricultural disease image recognition technologies: A review. Information Processing in Agriculture, 2022; 9(1): 48–59.

[14] Yuan Y, Chen L, Ren Y C, Wang S M, Li Y. Impact of dataset on the study for crop disease image recognition. Int J Agric & Biol Eng, 2022; 15(5): 181–186.

[15] Yang L, Yu X Y, Zhang S P, Long H B, Zhang H H, Xu S, et al. GoogLeNet based on residual network and attention mechanism identification of rice leaf diseases. Computers and Electronics in Agriculture, 2023; 204: 107543.

[16] Loey M, ElSawy A, Afify M. Deep learning in plant diseases detection for agricultural crops: A survey. International Journal of Service Science, Management, Engineering, and Technology, 2020; 11(2): 41–58.

[17] Dai W, Yang Q, Xue G, Yu Y. Boosting for transfer learning. In: Proceedings of the 24th International Conference on Machine Learning, 2007; pp.193–200.

[18] Pan S J, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2009; 22(10): 1345–1359.

[19] Fang S S, Yuan Y, Chen L, Zhang J, Li M, Song S D. Crop disease image recognition based on transfer learning. In: Proceedings of International Conference on Image and Graphics, 2017; pp.545–554. doi: 10.1007/978-3-319-71607-7_48.

[20] Jia S J, Jia P J, Hu S P, Liu H B. Automatic detection of tomato diseases and pests based on leaf images. In: Proceedings of 2017 Chinese Automation Congress (CAC), Jinan: IEEE, 2017; pp.2537–2564. doi: 10.1109/CAC.2017.8243388.

[21] Jiang Z C, Dong Z X, Jiang W P, Yang Y Z. Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning. Computers and Electronics in Agriculture, 2021; 186: 106184.

[22] Liu Y M, Wang Z L, Wang R J, Chen J S, Gao H J. Flooding-based MobileNet to identify cucumber diseases from leaf images in natural scenes. Computers and Electronics in Agriculture, 2023; 213: 108166.

[23] Zhao B, Wu X, Feng J S, Peng Q, Yan S C. Diversified visual attention networks for fine-grained object classification. IEEE Transactions on Multimedia, 2017; 19(6): 1245–1256.

[24] Jaderberg M, Simonyan K, Zisserman A. Spatial transformer networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, Cambridge, MA, USA: MIT Press, 2015; pp.2017–2025.

[25] Wang F, Jiang M Q, Qian C, Yang S, Li C, Zhang H, et al. Residual attention network for image classification. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017; pp.6450–6458. doi: 10.1109/CVPR.2017.683.

[26] Xiao T J, Xu Y C, Yang K Y, Zhang J X, Peng Y X, Zhang Z. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015; pp.842–850. doi: 10.1109/CVPR.2015.7298685.

[27] Feng S, Zhao D X, Guan Q, Li J P, Liu Z Y, Jin Z Y, et al. A deep convolutional neural network-based wavelength selection method for spectral characteristics of rice blast disease. Computers and Electronics in Agriculture, 2022; 199: 107199.

[28] Zheng J Y, Li K Y, Wu W B, Ruan H J. RepDI: A light-weight CPU network for apple leaf disease identification. Computers and Electronics in Agriculture, 2023; 212: 108122.

[29] Chen L, Yuan Y. Agricultural disease image dataset for disease identification based on machine learning. In: Proceedings of International Conference on Big Scientific Data Management, 2018; pp.263–274.

[30] Zeiler M, Fergus R. Visualizing and understanding convolutional networks. In: Proceedings of the European Conference on Computer Vision, 2014; pp.818–833.

[31] Komodakis N, Zagoruyko S. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In: Proceedings of International Conference on Learning Representations, 2017; pp.1–13.

[32] Zhou B L, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016; pp.2921–2930. doi: 10.1109/CVPR.2016.319.

[33] Woo S, Park J, Lee J-Y, Kweon I S. CBAM: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision, 2018; 11211: 3–19.

[34] Hu J, Shen L, Albanie S, Sun G, Wu E H. Squeeze-and-excitation networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018; 42(8): 2011–2023.