Enhancing Tea Disease Identification with Lightweight MobileNetV2

Zhilin Li
College of Big Data and Information Engineering, Guizhou University
Yuxin Li
College of Big Data and Information Engineering, Guizhou University
Chunyu Yan
College of Big Data and Information Engineering, Guizhou University
Peng Yan
College of Big Data and Information Engineering, Guizhou University
Xiutong Li
College of Big Data and Information Engineering, Guizhou University
Mei Yu
College of Big Data and Information Engineering, Guizhou University
Tingchi Wen
The Mushroom Research Centre, Guizhou University
Benliang Xie (blxie@gzu.edu.cn)
College of Big Data and Information Engineering, Guizhou University

Research Article

Keywords: Disease identification, Coordinate attention mechanism, Multi-scale feature extraction, Dilated convolution, Model pruning

Posted Date: November 9th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3565337/v1

Additional Declarations: No competing interests reported.

Abstract

Plant diseases in tea trees can result in significant losses in both the quality and quantity of tea production. Regular monitoring can prevent the occurrence of large-scale diseases in tea plantations. However, existing methods face challenges such as a high number of parameters and low recognition accuracy, which hinder their application for monitoring tea gardens on edge devices. This paper presents a lightweight I-MobileNetV2 model for identifying diseases in tea leaves, with the goal of addressing these challenges. The proposed method embeds a coordinate attention mechanism module into the original MobileNetV2 network, enabling the model to accurately locate disease regions. Furthermore, a multi-branch parallel convolution module is employed to extract disease features across multiple scales, which improves the adaptability of the model to different disease scales. Then an automated pruning strategy is employed to compress the model and reduce computational complexity. The results indicate that algorithm proposed surpass the original MobileNetV2 by 1.91 percentage points with an average accuracy of 96.12% based on self-built tea disease dataset, the model parameters have been reduced by 40%, making it more suitable for practical application in tea garden environments.

1. Introduction

China, as the largest producer and consumer of tea globally, holds a significant position in the national economy due to its thriving tea industry. The growth of this industry is beneficial for improving the income level of tea farmers and is an effective way to help them escape poverty. Tea diseases can significantly reduce both the yield and quality of tea during its planting and growth stages, leading to economic losses for farmers in the industry. Among them, the reduction in production caused by diseases such as tea algal spot disease, tea bud blight, and tea leaf blight account for approximately 20% [1]. Therefore, the rapid and accurate identification of tea diseases, and guidance for tea farmers on the correct use of pesticides, are of great significance for improving tea yield and reducing tea production losses.

China is home to approximately 130 known tea tree diseases, with more than 40 common diseases, among which leaf diseases account for a large proportion. Currently, tea diseases are primarily identified through manual recognition methods, which is often suboptimal [2]. Firstly, there are many types of tea diseases, and some diseases have similar symptoms, which can easily lead to misjudgments due to human subjective factors. Secondly, manual identification of tea diseases is both time-consuming and labor-intensive, requiring prior knowledge about tea diseases and a long time to identify tea diseases. Therefore, for the actual tea plantation environment, manual identification of tea diseases is impractical.

Machine learning and image processing techniques, driven by the rapid advancement of computer technology, have been widely applied in recognizing crop diseases [3–7]. When tea tree leaves are diseased, the color of the leaves changes from green to other colors, and there is a noticeable color contrast between healthy and diseased leaves. Machine learning and image processing techniques can extract texture information of different diseases, such as color, shape, or other features, and identify diseases based on this texture information. This process eliminates the need for human intervention, ensures the objectivity of the identification results, and significantly reduces the time required for accurate detection. Therefore, the feasibility and superiority of using machine learning and image processing techniques for the automatic identification of tea diseases are evident.

During the early stages, many scholars conducted numerous studies on tea diseases recognition using machine learning. Sun et al. [8] proposed a method that combines Simple Linear Iterative Cluster (SLIC) with Support Vector Machine (SVM) to accurately extract the saliency map of tea disease from complex backgrounds. Zou et al. [9] proposed a tea disease recognition method that utilizes spectral reflectance and machine learning techniques. The approach consists of a feature selector, employing a decision tree, and a tea disease recognizer, utilizing a random forest. By using the decision tree, relevant features are selected and redundant features are eliminated. The proposed method enables more effective feature selection and the ability to learn from high-dimensional data, leading to non-destructive and efficient identification of various tea diseases. Hossain et al. [10] developed an image processing system utilizing SVM as a classifier, enabling swift and precise identification of the two prevalent tea diseases in Bangladesh.

With the advancements in deep learning and computer technology, Convolutional Neural Networks (CNNs) have emerged as one of the most classical algorithms in this field due to its outstanding performance. It has have been widely applied and achieved remarkable results in tea disease recognition. Pandian et al. [11] proposed a Deep Convolutional Neural Network (DCNN) based on inverted residual and linear bottleneck layers for the diagnosis of tea grey blight disease, which can effectively identify the disease. Prabu et al. [12] developed a real-time disease prediction system for tea diseases using CNN on the Platform-as-a-Service (PaaS) cloud. The system can be accessed via a hyperlink deployed by the model using a smartphone. It allows users to capture images of tea diseases using their smartphone's camera and upload them to the cloud, where the cloud system automatically predicts and displays the disease on the mobile screen. Datta and Gupta. [13] proposed a deep CNN for accurately detecting 5 types of tea leaf diseases and healthy leaf. The proposed model achieved an impressive accuracy of 96.56% in identifying the specific disease. Hu et al. [14] proposed a low-shot learning approach for tea disease recognition. Their method involved segmenting disease spots on tea leaves using support vector machines to remove background interference. Additionally, an improved Conditional Deep Convolutional Generative Adversarial Network (C-DCGAN) was employed to address the limited training sample issue. Finally, the VGG16 model was trained to classify tea diseases. Hu et al. [15] improved the CIFAR10-quick model by adding a multi-scale feature extraction module. This enhancement helped the model automatically extract various tea disease image features. They also utilized depth-wise separable convolution to reduce model parameters and accelerate computation. Consequently, their approach achieved better average identification accuracy. Chen et al. [16] proposed a CNNs model called LeafNet for automatically extracting tea tree disease features from images. The recognition accuracy for the seven types of tea diseases reached as high as 90.16%. Mukhopadhyay et al.[17] proposed a method for detecting disease spots in tea leaves using a Non-dominated Sorting Genetic Algorithm (NSGA-II) based image clustering approach. They further employed Principal Component Analysis (PCA) and multi-class

SVM to reduce features and identify tea diseases, achieving a recognition accuracy of 83% for 5 types of tea diseases. Nath et al. [18] designed a tea disease identification model based on automatic learning, which combines a CNNs architecture using deep separable convolutions and residual networks with SVM. It achieves a recognition accuracy of 99.28% for healthy tea leaves and 3 types of tea disease images.

Although CNN has shown promising results in tea disease recognition, there is still faces significant challenges in detecting diseases with small lesions and complex backgrounds. Moreover, the above models have problems such as large model parameters, which are not conducive to deployment on mobile devices. In this article, we have made improvements to the MobilenetV2 model in order to build a lightweight, high-accuracy tea disease recognition model, providing strong support for the intelligent development of the tea industry. The key contributions of this study include:

(1) According to the needs of the proposed method, a dataset containing 5 types of tea diseases with complex backgrounds and healthy tea leaves was constructed.

(2) We introduce I-MobileNetV2, an enhanced MobileNetV2 model, for accurately recognizing tea diseases in natural scene images. This model effectively resolves issues related to recognition accuracy and excessive parameter quantity.

(3) The proposed method is suitable for deployment on mobile devices, enabling the diagnosis of tea diseases.

2. Materials and method

2.1 Dataset introduction

At the beginning of our work, due to the absence of publicly available tea disease datasets online, we invested a lot of manpower in collecting tea disease data in Yangai Tea Plantation in Huaxi District, Guiyang City, Guizhou Province. We used an OPPO Reno Ace smartphone with a resolution of 4000×3000 pixels to capture 2544 images of healthy tea leaves and five common tea diseases. Examples of healthy leaf image and tea disease image are shown in Fig. 1.

To enhance the classification accuracy and robustness of the tea disease recognition model while minimizing overfitting risks, we incorporated four offline data augmentation techniques, including brightness adjustment (0.8–1.2), chromaticity adjustment (1.2), rotation (90 degrees) and adding Gaussian noise, on top of the existing training set to expand the data. The augmented dataset was then split into a training set and a test set with a ratio of 7:3, as shown in Table 1, which served as the experimental dataset.

Table 1
Detailed information of tea leaf disease dataset

Categories	Training set	Validation set	Total number
HL	1218	522	1740
TLB	1078	462	1540
TALS	1341	575	1916
TCS	1022	438	1460
TBB	1170	502	1672
TGB	1294	554	1848
Total	7123	3053	10176

where, HL represents healthy leaf, TLB represents tea leaf blight, TALS represents tea aglae spot, TCS represents tea cercospora scab, TBB represents tea bud blight, TGB represents tea grey blight.

2.2 Basic model

The MobileNetV2 network, proposed by Google in 2018, is a lightweight convolutional neural network [19]. Compared with MobileNetV1 [20], it has higher accuracy and lower model parameter quantity. The structure of MobileNetV2 is shown in Fig. 2, which continues the idea of using depth-wise separable convolution to reduce model parameters in MobileNetV1 and has two improvements: inverted residual blocks and linear bottleneck layers. The traditional residual block first reduces and then increases the feature channel dimension. In contrast, the inverted residual block first expands the feature channel dimension, extracts feature information from each channel, and then reduces it. This method compensates for the problem of reduced feature information caused by depth-wise separable convolution and ensures the feature extraction ability of the model. The structure of the inverted residual block is shown in Fig. 3.

In addition, ReLU activation function can cause a lot of loss of low-dimensional feature information. MobileNetV2 replaces the last convolutional layer of the inverted residual structure with a linear ReLU6 activation function, thereby reducing the loss of low-dimensional feature information and ensuring the recognition performance of the network.

2.3 Coordinate attention mechanism

To effectively extract disease spot features and improve recognition accuracy, we considered adding an attention mechanism to the MobileNetV2 network. To address the limitation of existing attention mechanisms in capturing location and channel information, which are essential for identifying target objects in visual tasks, we incorporated a hybrid domain coordinate attention mechanism (CA) [21]. By integrating location information into the channel attention, our model was able to gather information across a broader range and accurately identify the target region, leading to successful recognition outcomes.

The CA attention mechanism is based on two steps: embedding coordinate information and generating coordinate attention, which encode channel relationships and long-range dependencies. the specific structure is illustrated in Fig. 4. For the input feature map *X*, pooling convolutional kernels of size (*H*, 1) and (1, *W*) are used to encode each channel along the horizontal and vertical coordinates. After calculating the average pooling, the expression for the encoded representation of the *c*-th channel at height *h* and width *w* is obtained:

$$z_c^h\left(h\right)=\frac{1}{W}\sum_{0\le i<W}x_c(h,i)$$

1

$$z_c^w\left(w\right)=\frac{1}{H}\sum_{0\le i<H}x_c(j,w)$$

2

At this point, the coordinate information embedding process is complete. Continue to concatenate $z_c^h\left(h\right)$ and$z_c^w\left(w\right)$ and do the convolution operation to compress the channels, and then after batch normalization process and non-linear activation operation to get the intermediate feature map, as shown in Eq. (3).

$$f=\delta\left(F_1(\left[z^h,z^w\right])\right.$$

3

In Eq. (3), [.,.] indicates the spatial concatenation operation, δmeans the non-linear activation function, and F_1 represents channel compression. Next, the function f is divided into two independent tensors along the spatial dimension, and then convolved again to adjust the channel numbers of the two tensors to match the channel numbers of the input *X*. After passing through the activation functionσ, the attention weights on the height and width dimensions are obtained as follows:

$$g^h=\sigma\left(F_h\left(f^h\right)\right)$$

4

$$g^w=\sigma\left(F_w\left(f^w\right)\right)$$

5

In Eqs. (4)-(5), $\sigma$is the sigmoid activation function, F_h andF_w represent convolutional operations on the height and width dimensions. After performing a weighted operation on g^h and g^w with the input *X*, attention generation is complete, as shown in Eq. (6).

$$y_c(i,j)=x_c(i,j)\times g_c^h\left(i\right)\times g_c^w\left(j\right)$$

6

To fully exploit the effectiveness of the coordinate attention, this study embeds the CA module into different positions of the MobileNetV2 stride-1 inverted residual structure to explore its performance at different positions, as shown in Fig. 5. The accuracy and F1-score obtained from the comparative experiments with CA embedding in Section 4.4, this experiment selects the embedding method of Fig. 5(d) to embed the CA module.

2.4 Multi-branch parallel convolution

The standard MobileNetV2 network uses fixed convolution3×3 kernels, which have limited receptive fields and can only capture local information of lesions, failing to capture rich detail and contextual information. Enlarging the receptive field can capture more multi-scale contextual information. Considering that dilated convolution does not increase additional parameters while enlarging the receptive field, different receptive fields can be obtained by setting the dilation rate of the dilated convolution [22]. The calculation formula for the receptive field size of a dilated convolution with dilation rate *r* and kernel size $k \times k$ is:

$$k' = k + (k - 1) \bullet (r - 1)$$

7

Therefore, in this paper, we combine the advantages of dilated convolution and design a Multi-branch Parallel Convolution Module (MPCM), which replaces the stride-2 inverted residual structure in MobileNetV2. The module structure can be seen in Fig. 6.

Firstly, the input undergoes a 1x1 point convolution, followed by parallel extraction of multiple scale features using dilated convolutions with dilation rates of 1, 2, and 3. Secondly, considering that dilated convolutions may not extract features from the dilated parts, 3x3 and 5x5 depth convolutions are added after the branches with dilation rates of 2 and 3 respectively, to extract features from the dilated parts. Finally, each branch is processed with a 1x1 point convolution and the features are fused. Batch normalization and ReLU6 activation function are used after each convolution in the figure to accelerate model convergence and alleviate gradient disappearance.

2.5 Model lightweighting

Model pruning is a model compression technique that mainly involves removing channels and layers with minimal impact on network performance to achieve an optimal balance between model size and recognition performance. Traditional model compression relies on human expertise, requiring domain experts to manually adjust the search for the optimal configuration, making the process tedious and time-consuming. However, the Automatic Machine Learning (AMC) for the Model Compression algorithm can fully automate model pruning and achieve excellent pruning results [23]. To enhance the efficiency of model pruning, this paper uses the AMC algorithm to lighten the original model. Figure 7 shows the algorithm framework of AMC.

The AMC algorithm consists of two parts: Agent and Environment. The Agent part uses the Deep Deterministic Policy Gradient (DDPG) algorithm, consisting of three parts: Embedding, Actor, and Critic. The pruning algorithm follows the steps below:

(1) The agent receives the encoded state of layer L_t from the environment and then trains reinforcement learning to search for the sparsity ratio of model layer L_t.

(2) Pruning operations are performed based on the sparsity ratio obtained from the previous step.

(3) Repeat steps 1 and 2 until the pruning strategy search for all layers is completed.

(4) The pruned model is evaluated for accuracy on the validation set and the results are returned to the agent.

After completing the above steps, the model undergoes fine-tuning, where it is retrained to restore the accuracy of the pruned model.

3. Results and discussion

3.1 Experiment setup

The experiment was developed using Python language and based on PyTorch 1.8.0 deep learning framework. The CPU was Intel(R) Core (TM) i5-11400F, and the operating system was Windows 10. NVIDIA GeForce RTX 3060 GPU with 8GB memory was used for accelerated training. The input disease images were cropped to 224×224, and the Adam optimizer was employed to update gradients with an initial learning rate of 0.001. Each processing batch had a size of 16, and the model was iterated for a total of 200 epochs.

3.2 Evaluation index

There are various methods to evaluate the performance of the model, and this study primarily focuses on Accuracy, Precision, Recall, F1-score, and the associated issues with deploying the model on mobile devices. The following six evaluation methods were used to assess the model performance.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \times 100\%$$

8

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

9

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

10

$$F1 - score = \frac{2^{*}P^{*}R}{P + R} \times 100\%$$

11

In Eqs. (8)-(11), TP stands for true positive, TN indicates true negative, FP is false positive, and FN is false negative of the predicted class.

The model parameter count, also known as params, refers to the number of parameters in the model. It not only directly determines the size of the model file but also affects the amount of computer memory used during model inference, which can be limited in resource-constrained mobile platforms. The computational complexity of the model can be quantified in terms of Floating Point Operations (FLOPs), where a lower FLOPs value indicates a smaller computational load, faster execution speed, and shorter disease detection time.

3.3 Performance comparison of embedded CA

In order to verify the excellence of the CA embedding method used in this paper, a comparative experiment of the four embedding methods in Fig. 5 of Section 2.3 was designed. The experimental results on the tea disease dataset are shown in Table 2.

Table 2
Performance of different embedding methods of CA module

Basic Model	Embedding method	Accuracy/%	F1-score/%
MobileNetV2	-	94.21	94.11
	CA-First	94.51	94.69
	CA-Second	94.92	94.75
	CA-Third	93.88	94.17
	CA-Last	**95.54**	**95.26**

From the data presented in Table 2, it is apparent that the CA-First embedding method improved the accuracy by 0.3% and the F1-score by 0.58% compared to the baseline model. For the CA-Second embedding method, its accuracy was 0.71% higher than the baseline model, while its F1-score was 0.64% higher. However, for the CA-Third embedding method, after embedding the CA module, its accuracy was 0.33% lower than the baseline model, and the F1-score was only 0.06% higher than the baseline model. For the CA-Last embedding method, its accuracy and F1-score were 1.33% and 1.15% higher than the baseline model, respectively. Overall, by introducing the CA module after the last 1×1 pointwise convolution in the MobileNetV2 inverted residual module, it can strengthen the connection between channels and compress the number of channels. Therefore, this experiment adopted the CA-Last embedding method to embed the CA module.

3.4 Comparison of pruning results

To demonstrate the effectiveness of the pruning method employed in this paper, we conducted pruning experiments on two other excellent pruning methods under the same experimental conditions. The pruning effects of the three pruning methods are shown in Table 3.

Table 3
Pruning results

Algorithms	Params/M	FLOPs/G	Accuracy/%
Molchanov et al. [24]	1.04	0.21	**93.23**
Liu et al. [25]	0.91	**0.13**	91.97
AMC	**0.83**	0.14	93.15

Table 3 reveals that while Molchanov et al.'s approach maintains a higher accuracy, the model still has a large number of parameters and FLOPs. On the other hand, the method of Liu et al. has a similar number of parameters and FLOPs as the pruning method discussed in this paper, but it experiences a notable decrease in accuracy. Overall, the pruning method proposed in this paper has a higher compression rate and recognition accuracy, rendering it more suitable for deployment on edge devices.

3.5 Ablation experiment

To verify the impact of each improvement of this network on model performance, the ablation experiments were conducted based on the original MobileNetV2 network. The ablation experiments mainly included embedding the coordinate attention mechanism, introducing a multi-branch parallel convolution module, and using an automated pruning strategy in the base network to design comparison experiments, and the experimental results are shown in Table 4. During the training iteration process, the accuracy and loss value curves of the original and improved models are shown in Fig. 8.

Table 4
Ablation experiment results

Model	CA	MPCM	AMC	Accuracy/%	Params/M
MobileNetV2	-	-	-	94.21	2.23
	√	-	-	95.54	2.27
	-	√		95.79	4.28
	-	-	√	93.15	**0.83**
I-MobileNetV2	√	√	√	**96.12**	1.35

Based on the experimental data presented in Table 4, it is evident that after embedding the coordinate attention mechanism, the models' parameters only increased by 0.04M, while the accuracy improved by 1.33%. This indicates that the embedding of the coordinate attention mechanism weakened irrelevant information, strengthened the representation of disease information, and improved network performance. In addition, after adding the Multi-Branch Parallel Convolution Module (MPCM), the accuracy increased by 1.58%, while the models' parameters increased by 2.05M. This suggests that the fusion of features from three branches obtained multi-level feature information at different scales, thereby improving the feature representation capability, but also resulting in many parameters. Finally, using automated model pruning to compress the model's resulted in a 1.06% decrease in accuracy, but it greatly simplified the model complexity. After pruning, the number of model parameters was approximately 1/3 of the original model's parameters. Therefore, it can be seen that the various improvements added in this article have a positive impact on the model, and the model's performance after incorporating all three improvements is better than that of using a single or two improvements.

In addition, the confusion matrix contains the number of correctly and incorrectly predicted samples in the test set, which can help us visually evaluate the classification accuracy of each category. To analyze the differences in the improved model's recognition performance for different categories of tea diseases, we plotted the confusion matrix of the I-MobileNetV2 model on the test set, as shown in Fig. 9.

From Fig. 9, it can be seen that the confusion matrix displays the recognition accuracy and misclassification of each class in the test set, with high misidentification rates for Tea Cercospora Scab Disease and Tea Grey Blight Disease. This is because when the lesion area of Tea Cercospora Scab Disease is large and there are few lesions, it is highly similar to Tea Grey Blight Disease, leading to confusion between the two diseases. Although a small number of disease samples were misclassified among similar samples, the recognition effect was not ideal. overall, the improved model could correctly identify most disease samples, and misclassifications and omissions only occurred in a very small number of samples, getting a high overall recognition accuracy.

3.6 Model comparison

To validate the effectiveness of the proposed algorithm on the tea disease dataset, experimentation was conducted using 7 popular convolutional neural networks. Table 5 provides a performance comparison of training VGG16 [26], GoogLeNet [27], ResNet18 [28], ShuffleNetV2 [29], EfficientNet_B0 [30], GhostNet [31], RepVGG [32] and the I-MobileNetV2 on the tea disease dataset.

Table 5
Performance comparison of different models

Model	Accuracy/%	Precision/%	Recall/%	F1-score/%	Params/M	FLOPs/G
VGG16	95.67	97.15	94.37	95.43	134.29	15.95
GoogLeNet	94.40	94.95	93.22	94.13	10.32	1.53
ResNet18	95.72	96.53	95.28	95.66	11.18	1.78
ShuffleNetV2	95.43	95.11	**96.74**	95.46	1.26	0.16
EfficientNet_B0	96.35	98.23	95.88	96.51	4.02	0.47
GhostNet	95.51	97.35	96.15	96.50	**1.19**	**0.14**
RepVGG_A0	**96.74**	97.42	96.51	96.69	18.74	1.92
I-MobileNetV2	96.12	**98.57**	95.36	**97.15**	1.35	0.21

From Table 5, it can be seen that I-MobileNetV2 has the highest Precision and F1-score, with the third highest recognize accuracy. It has relatively less number of parameters and FLOPs. Compared to RepVGG_A0, the I-MobileNetV2 model's performance is slightly lower in terms of recognize accuracy by 0.62%, but it outperforms RepVGG_A0 in terms of F1-score, parameter, and FLOPs. ShuffleNetV2 has a higher Recall score than I-MobileNetV2 model, and its parameter and FLOPs are also lower, but its recognize accuracy is 0.69% lower than the I-MobileNetV2 and its parameter and FLOPs are comparable. Overall, the I-MobileNetV2 has a competitive advantage. Additionally, in terms of model lightweight property and FLOPs, the GhostNet model is more competitive, but considering overall recognize accuracy and other evaluation metrics, the improved model still remains the best.

In summary, the improved model not only improves recognition accuracy but also significantly reduces the number of parameters and FLOPs, this provides good conditions for real-time deployment. The comprehensive performance is more superior compared to the above seven mainstream models.

4. Conclusion

This article focuses on the identification of tea diseases in natural environments, selecting healthy tea leaves and five common diseases as experimental objects. Considering the complexity of tea disease backgrounds and the difficulty of identifying small lesions, we improved the lightweight network MobileNetV2, effectively solving these issues. The improved model first embedded a coordinate attention mechanism to reduce background noise and other interference, making the network more focused on disease information. Secondly, a multi-branch parallel convolution module was added to enhance the model's ability to extract features of different scales. Finally, an automated model pruning algorithm was used to make the model lightweight and easy to deploy on edge devices. As a result of these improvements, the recognition accuracy of the improved network increased by 1.91%, and the parameter quantity was reduced by about 40% compared to the original model. The experimental results show that the improved network achieved a good balance between recognition accuracy and model parameter quantity on the tea disease dataset. Compared with popular neural networks, the improved network has a better recognition effect on tea diseases and fewer model parameters, which can meet the needs of multi-class tea disease identification in real environments and bring hope for deploying the model to field devices.

In the future, we will further improve the model and plan to deploy it on mobile devices for practical application in tea production processes. At the same time, we will collect more tea disease categories and image data in natural environments, striving to build a larger-scale dataset.

Declarations

Funding This work was supported by National Key Research and Development Program (No. 2016YFD0201305-07), Guizhou Provincial Basic Research Program (Natural Science) (No. ZK [2023]060), Open Fund Project in Semiconductor Power Device Reliability Engineering Center of Ministry of Education (No. ERCMEKFJJ2019-06).

Conflict on Interests The authors have declared no conflict of interest.

Data availability The data included in this study can be obtained by contacting the corresponding author upon request.

Author contributions Z.L. gave the idea, did the experiments, interpreted the results and wrote the paper, C.Y. did result validation and review Y.L. drew some figures P.Y., X.L., and M.Y. helped with data collection, and data processing. T.W. organized and check the manuscript. B.X. is the corresponding authors, supervised the whole project, and revised the paper.

References

1. Hu, G., Yang, X., Zhang, Y., et al.: Identification of tea leaf diseases by using an improved deep convolutional neural network. Sustainable Computing: Informatics and Systems. **24**, 100353 (2019). https://doi.org/10.1016/j.suscom.2019.100353
2. Bao, W., Fan, T., Hu, G., et al.: Detection and identification of tea leaf diseases based on AX-RetinaNet. Scientific Reports. **12**(1), 2183 (2022). https://doi.org/10.1038/s41598-022-06181-z
3. Kamilaris, A., Prenafeta-Boldú, F.X.: Deep learning in agriculture: A survey. Computers and electronics in agriculture. **147**, 70-90 (2018). https://doi.org/10.1016/j.compag.2018.02.016
4. Pantazi, X.E., Moshou, D., Tamouridou, A.A.: Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers. Computers and electronics in agriculture. **156**, 96-104 (2019). https://doi.org/10.1016/j.compag.2018.11.005
5. Jaisakthi, S.M., Mirunalini, P., Thenmozhi, D.: Grape leaf disease identification using machine learning techniques. In: 2019 International Conference on Computational Intelligence in Data Science

(ICCIDS), pp. 1-6 (2019). https://doi.org/10.11109/ICCIDS.2019.8862084

6. Hong, H., Lin, J., Huang F.: Tomato disease detection and classification by deep learning. In: 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), pp. 25-29 (2020). https://doi.org/ 10.1109/ICBAIE49996.2020.00012
7. Yu, H., Cheng, X., Chen, C., et al.: Apple leaf disease recognition method with improved residual network. Multimedia Tools and Applications. **81**(6), 7759-7782 (2020). https://doi.org/10.1007/s11042-022-11915-2
8. Sun, Y., Jiang, Z., Zhang, L., et al.: SLIC_SVM based leaf diseases saliency map extraction of tea plant. Computers and electronics in agriculture. **157**, 102-109 (2019). https://doi.org/10.1016/j.compag.2018.12.042
9. Zou, X., Ren, Q., Cao, H., et al.: Identification of tea diseases based on spectral reflectance and machine learning. Journal of Information Processing Systems. **6**(2), 435-446 (2020). https://doi.org/10.3745/JIPS.02.0133
10. Hossain, S., Mou, R.M., Hasan, M.M., et al.: Recognition and detection of tea leaf's diseases using support vector machine. In: 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), pp. 150-154 (2018). https://doi.org/ 10.1109/CSPA.2018.8368703
11. Pandian, J.A., Nisha, S.N., Kanchanadevi, K., et al.: Grey Blight Disease Detection on Tea Leaves Using Improved Deep Convolutional Neural Network. Computational Intelligence and Neuroscience, (2023). https://doi.org/10.1155/2023/7876302
12. Prabu, S., TapasBapu, B.R., Sridhar, S., et al.: Tea Plant Leaf Disease Identification Using Hybrid Filter and Support Vector Machine Classifier Technique. In: Advances in Data Science and Management: Proceedings of ICDSM 2021, pp. 591-601 (2022). https://doi.org/10.1007/978-981-16-5685-9_58
13. Datta, S., Gupta, N.: A novel approach for the detection of tea leaf disease using deep neural network. Procedia Computer Science. **218**, 2273-2286 (2023). https://doi.org/10.1016/j.procs.2023.01.203
14. Hu, G., Wu, H., Zhang, Y., et al.: A low shot learning method for tea leaf’s disease identification. Computers and Electronics in Agriculture. **163**, 104852 (2019). https://doi.org/10.1016/j.compag.2019.104852
15. Hu, G., Yang, X., Zhang, Y., et al.: Identification of tea leaf diseases by using an improved deep convolutional neural network. Sustainable Computing: Informatics and Systems. **24**, 100353 (2019). https://doi.org/10.1016/j.suscom.2019.100353
16. Chen, J., Liu, Q., Gao, L.: Visual tea leaf disease recognition using a convolutional neural network model. Symmetry. **11**(3), 343 (2019). https://doi.org/10.3390/sym11030343
17. Mukhopadhyay, S., Paul, M., Pal, R., et al.: Tea leaf disease detection using multi-objective image segmentation. Multimedia Tools and Applications. **80**, 753-771 (2021). https://doi.org/10.1007/s11042-020-09567-1
18. Nath, M., Mitra, P., Kumar, D.: A novel residual learning-based deep learning model integrated with attention mechanism and SVM for identifying tea plant diseases. International Journal of Computers and Applications. **45**(6), 471-484 (2023). https://doi.org/10.1080/1206212X.2023.2235750

19. Sandler, M., Howard, A., Zhu, M., et al.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and pattern recognition (CVPR), pp. 4510-4520 (2018). https://doi.org/10.48550/arXiv.1801.04381
20. Howard, A.G., Zhu, M., Chen, B., et al.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv: 1704. 04861 (2017). https://doi.org/10.48550/arXiv.1704.04861
21. Hou, Q., Zhou, D., Feng, J.: Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition (CVPR), pp. 13713-13722 (2021). https://doi.org/10.48550/arXiv.2103.02907
22. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv: 1511. 07122 (2015). https://doi.org/10.48550/arXiv.1511.07122
23. He, Y., Lin, J., Liu, Z., et al.: Amc: Automl for model compression and acceleration on mobile devices. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 784-800 (2018). https://doi.org/10.48550/arXiv.1802.03494
24. Molchanov, P., Tyree, S., Karras, T., et al.: Pruning convolutional neural networks for resource efficient inference. arXiv preprint arXiv: 1611.06440 (2016). https://doi.org/10.48550/arXiv.1611.06440
25. Liu, Z., Li, J., Shen, Z., et al.: Learning efficient convolutional networks through network slimming. In: Proceedings of the IEEE International Conference on Computer Vision (CVPR), pp. 2736-2744 (2017). https://doi.org/10.48550/arXiv.1708.06519
26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556 (2014). https://doi.org/10.48550/arXiv.1409.1556
27. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1-9 (2015). https://doi.org/10.48550/arXiv.1409.4842
28. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), pp. 770-778 (2016). https://doi.org/10.48550/arXiv.1409.4842
29. Ma, N., Zhang, X., Zheng, H.T., et al.: Shufflenet v2: Practical guidelines for efficient cnn architecture design. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 116-131 (2018). https://doi.org/10.48550/arXiv.1807.11164
30. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning (ICML), pp. 6105-6114 (2019). https://doi.org/10.48550/arXiv.1905.11946
31. Han, K., Wang, Y., Tian, Q., et al.: Ghostnet: More features from cheap operations", In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1580-1589 (2020). https://doi.org/10.48550/arXiv.1911.11907
32. Ding, X., Zhang, X., Ma, N., et al.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13733-13742, IEEE (2021).

https://doi.org/10.48550/arXiv.2101.03697

Figures

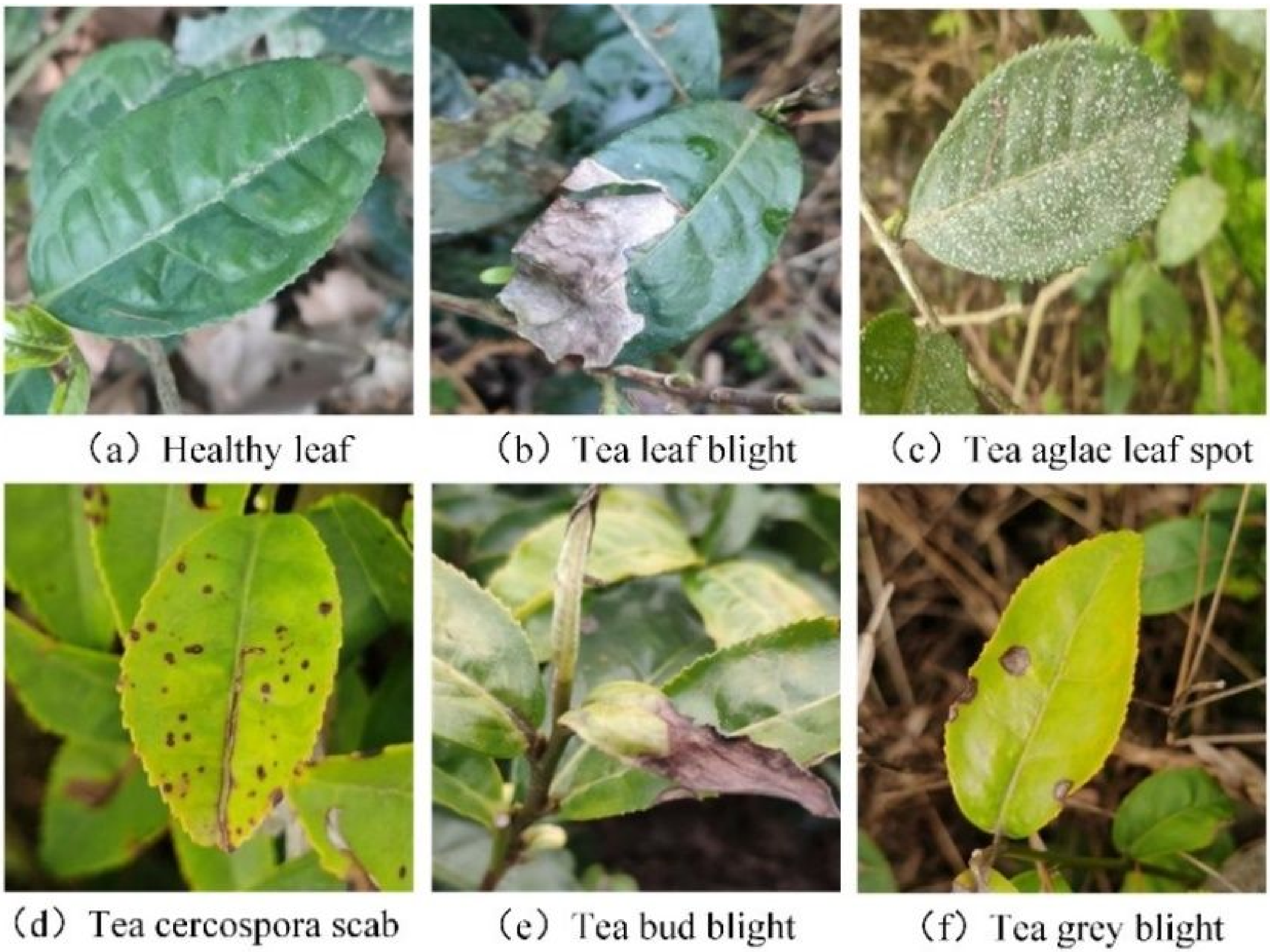

Figure 1

Images of healthy leafand tea leaf disease

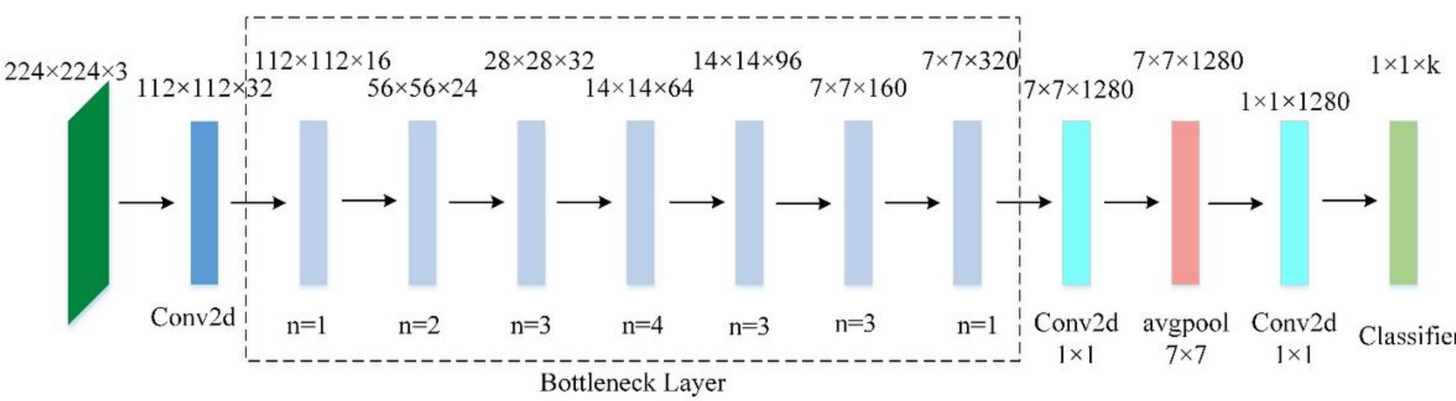

Figure 2

MobileNetV2 structure diagram

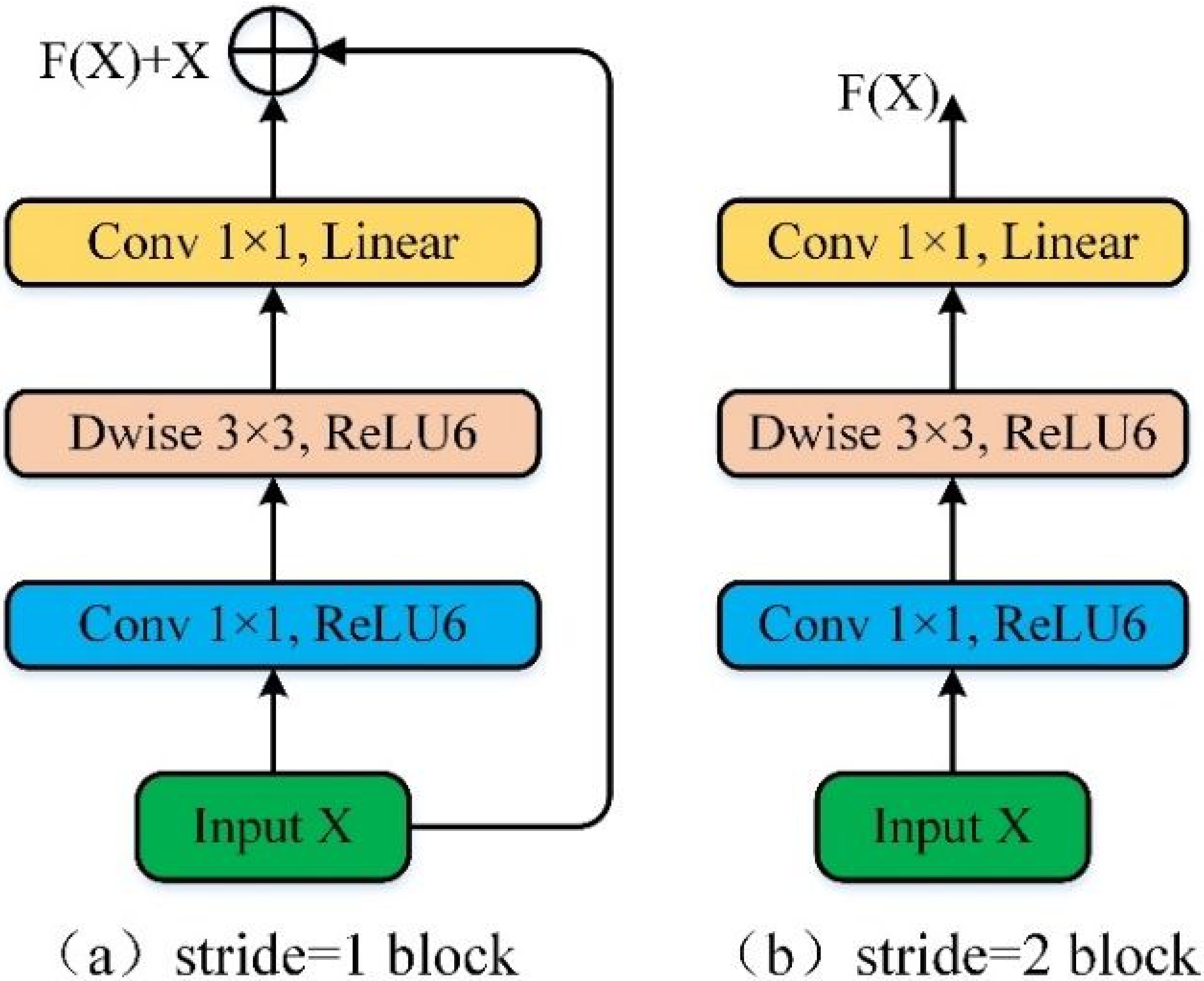

Figure 3

The structure of Inverted residual block

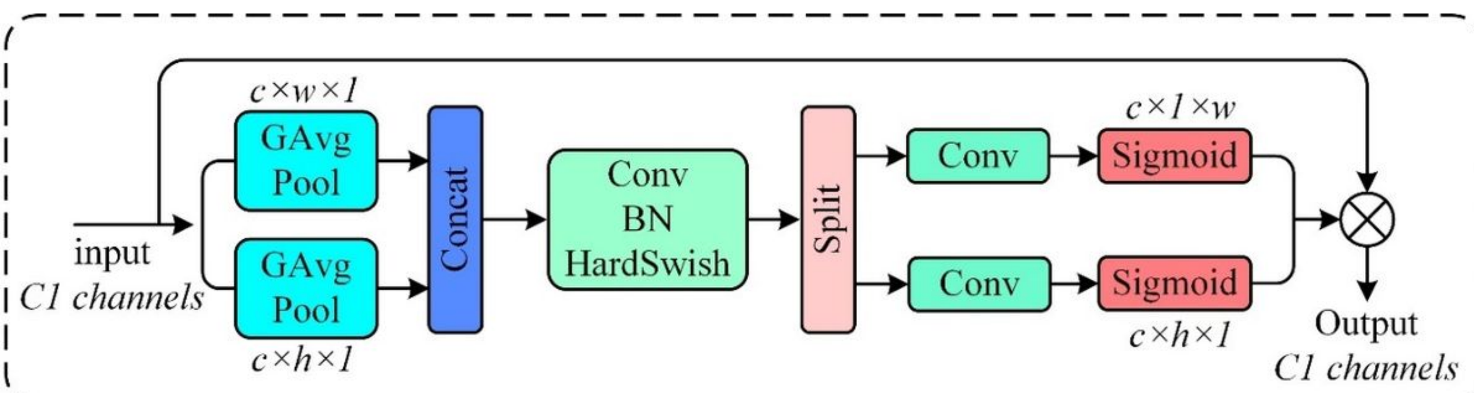

Figure 4

Coordinate attention mechanism

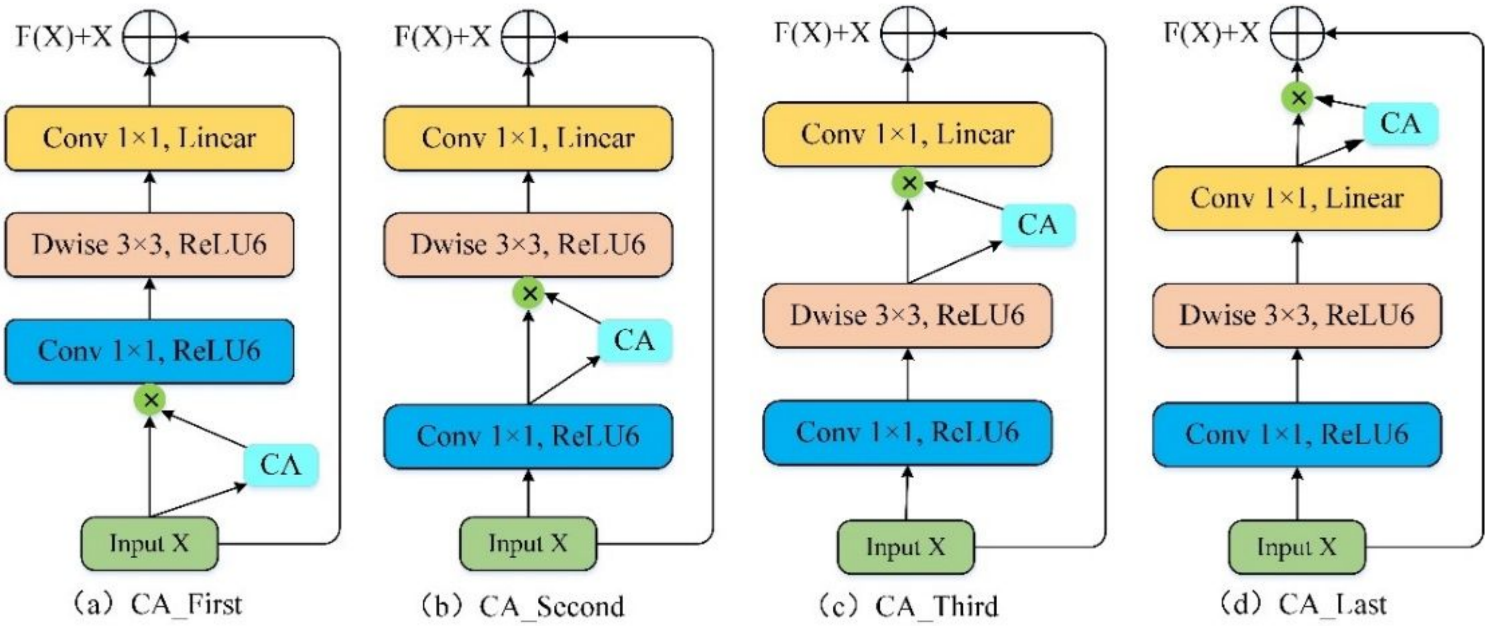

Figure 5

CA modules with different embedding positions

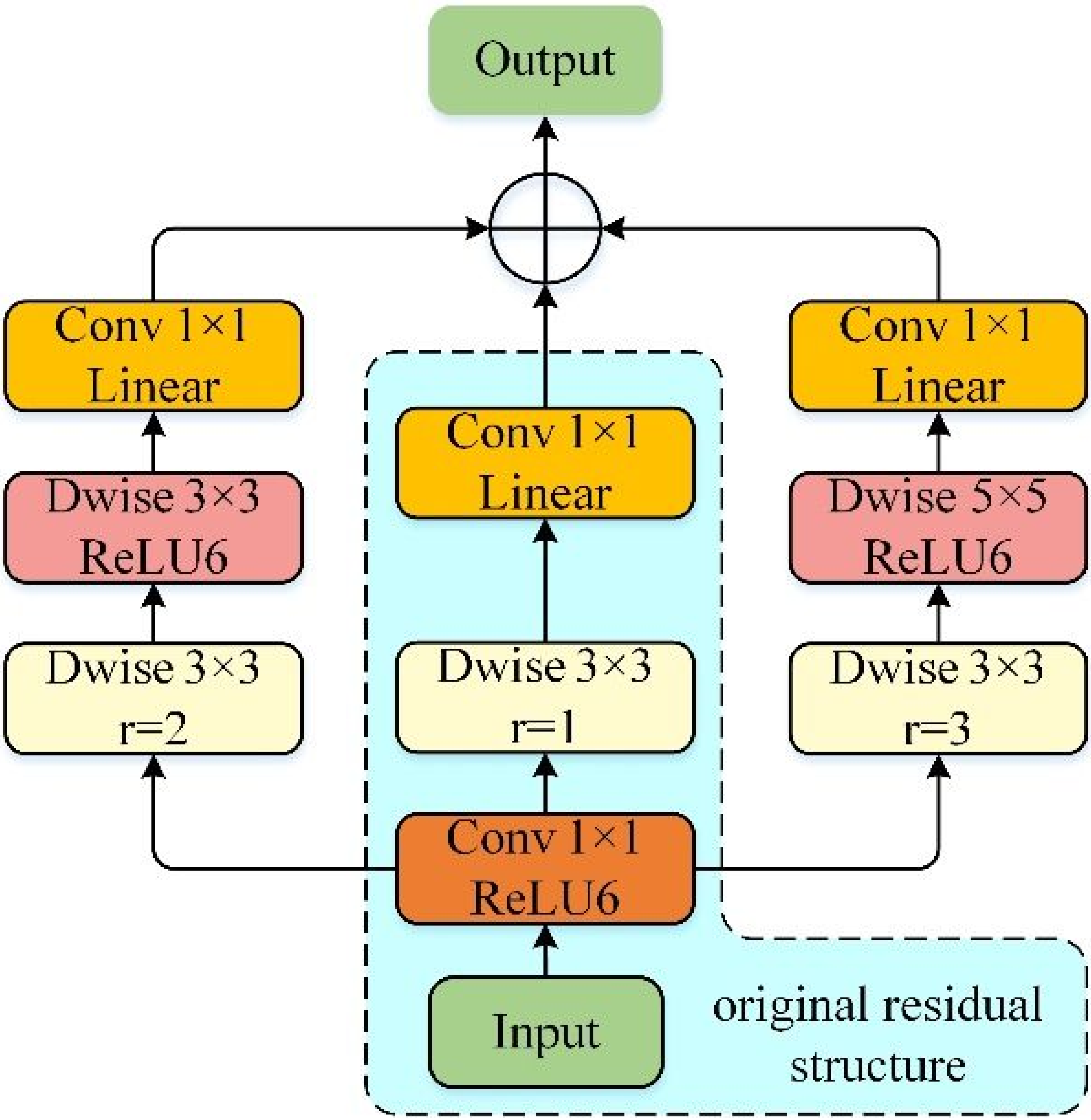

Figure 6

Multi-branch parallel convolution block

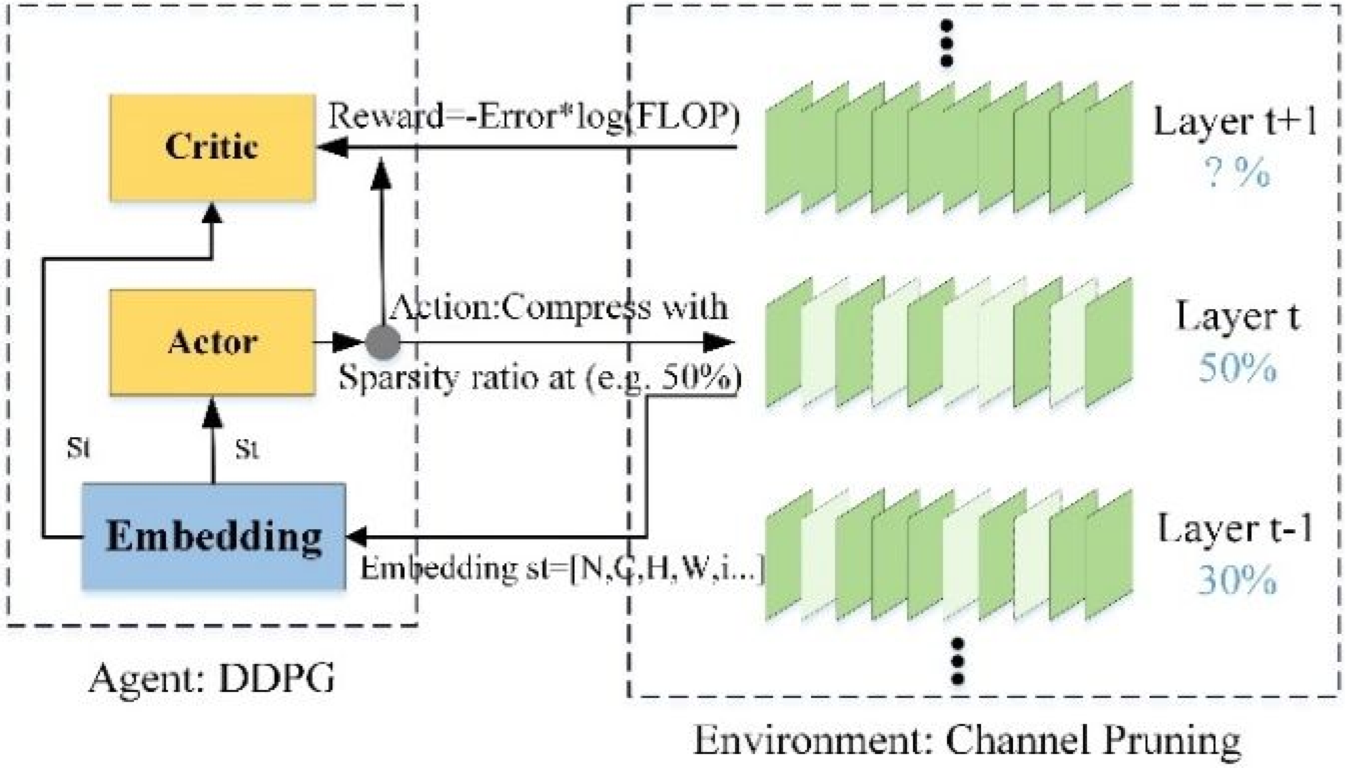

Figure 7

AMC algorithm framework

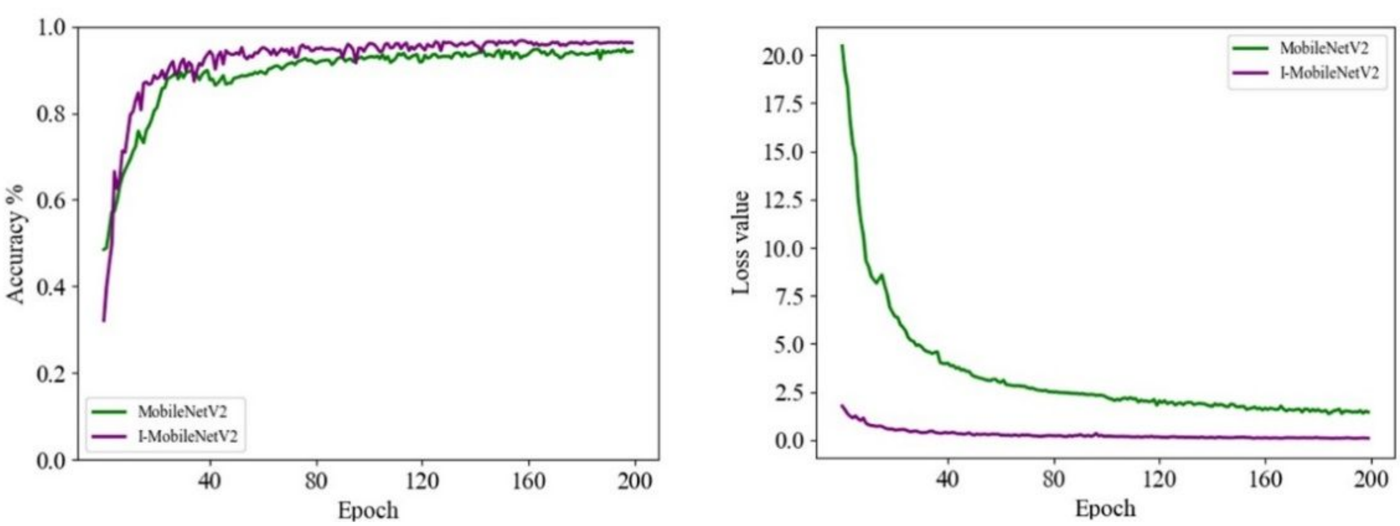

Figure 8

Accuracy and loss value change curves of the original and improved models

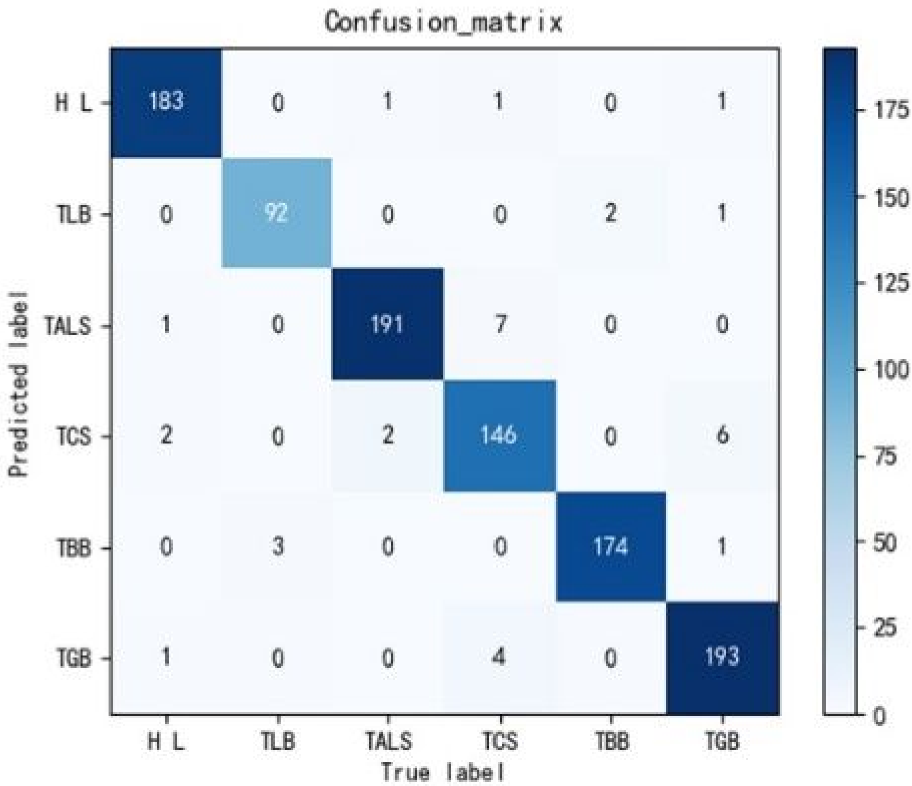

Figure 9

The confusion matrix of the test set