Rice Disease Identification System Using Lightweight MobileNetV2

Zhenghua Zhang^a, Yifeng Gu^a, Qingqing Hong^{a,*}

^aYangzhou University, 88 South University Road, Yangzhou City, Jiangsu Province, China

*Corresponding Author: sober_guyf@foxmail.com

Abstract

Rice is one of the main food crops in China, and rice diseases have become an important factor influencing the increase in food production losses in China. Traditional manual identification of rice diseases is time-consuming and labor-intensive. Machine learning algorithms have improved this problem and have been applied to the field of smart agriculture. The convolutional neural network (CNN) in deep learning has a significant effect on rice disease recognition relying on the characteristics of automatically extracting features. Aiming at five major rice diseases such as sheath blight, rice blast, bacterial leaf blight, rice smut and brown spot, this paper proposed a rice disease identification system using lightweight MobileNetV2. The identification results are uploaded and saved to the cloud database. Based on the lightweight model MobileNetV2, the system uses the channel pruning method to further compress the model. Compared with the original model, the memory usage has been reduced by 74%, the number of floating-point operations per second (FLOPS) has been reduced by 49%, the number of parameters has been reduced by 50%, and the accuracy of rice disease identification has increased by 0.16% to 90.84%.

Keywords: Rice disease, MobileNetV2, Channel pruning.

1. Introduction

Smart agriculture is an advanced stage of agricultural development with smart production as the core, refined, intelligent, intensive, and scientific production. It is a new model for the development of agricultural modernization, promoting the quality and efficiency of agricultural products. Rice is one of the main food crops in China, with a wide planting area. Affected by environmental conditions and cultivation techniques, rice diseases and insect pests have become more and more serious, the damage has increased, and the loss of rice yield has also increased⁽¹⁾.

Experts estimate that there are more than 1,600 kinds of diseases, pests, weeds, and rodents that occur all year round in crops in my China. A mong them, there are more than 100 types that can cause serious damage, and 14 million tons of food are lost due to diseases and insect pests each year. In view of these circumstances, the state has given more and more preferential policies and support in the field of smart agriculture. Smart agriculture uses and integrates advanced methods such as Internet of Things (IoT) technology and image processing technology to monitor the production environment and growth status of crops intelligently⁽²⁾. Remote monitoring and automatic disease identification can help prevent rice diseases from occurring on a large scale, laying a foundation for reducing food losses caused by diseases. This is of great significance to our country's goal of stabilizing and increasing grain output and ensuring food security this year. Compared with the traditional artificial visual recognition of crop diseases and continuous monitoring of crop growth, the application of advanced technologies such as machine learning has obvious advantages. Advanced technology saves a lot of labor and greatly improves agricultural production efficiency. In recent years, convolutional neural network (CNN) has been applied to different scene tasks relying on powerful feature extraction capabilities (3), and has achieved a significant improvement in accuracy. However, conventional CNN often requires a huge amount of floating point operations and occupies a large amount of storage space while achieving satisfactory accuracy. It has higher and higher hardware requirements for terminal equipment⁽⁴⁾, and relies on the powerful computing capabilities of GPUs to support the training and inference of network models. Therefore, how to use the redundancy of CNN structure and parameters to compress the model, obtain a model with fewer parameters and a more streamlined structure without affecting the automatic identification of rice diseases has become the key to reducing food loss.

2. Related Work

In recent years, the application of deep learning in the field of smart agriculture has received continuous attention. Many scholars at home and abroad have carried out research on the identification of plant diseases and the compression of CNN models. At present, there are relatively few researches on the identification of rice diseases, and most of them rely on machine learning algorithms such as support vector machine (SVM) for identification. Guan et al. used color features and outer contours to segment diseased spots, extracted 63 parameters of diseased spots morphology, color, and texture features, and used Bayesian discriminant method to identify three common diseases in rice(5). Liu et al. used the mean shift image segmentation algorithm to extract the diseased spots in the leaf image, and used the SVM model to achieve accurate identification of 15 main diseases in rice leaves (6). Suresha et al. used the global threshold method and K-Nearest Neighbor (KNN) classifier to classify rice blast and brown spot⁽⁷⁾. Xu et al. designed a set of Augmented Reality (AR) assisted recognition schemes based on histogram of oriented gradient and support vector Machine (HOG-SVM), and it can give faster training results and recognition results under the premise of small training samples⁽⁸⁾. Has an et al. trained an SVM classifier to identify and classify 9 different types of rice diseases⁽⁹⁾. Machine learning algorithms are demanding on disease images and need to extract specific features. Zhou et al. proposed a method for detecting rapid rice disease based on K-Means clustering algorithm (FCM-KM) and Faster Region-CNN (R-CNN) fusion to address various problems with the rice disease images, such as noise, blurred image edge, large background interference and low detection accuracy (10). CNN in the deep learning algorithm can automatically extract features, but its application in rice disease recognition is relatively scarce. The current mainstream model compression methods are as follows: parameter pruning, parameter quantization, low-rank decomposition, parameter sharing, compact network and knowledge distillation(11). Lebedev et al. used the OBD algorithm to treat the convolution operation as a matrix multiplication calculation, made the convolution kernel sparse in a group manner, and turned it into a sparse matrix multiplication, and this method improved the calculation speed⁽¹²⁾. Courbariaux et al. first proposed binarized neural network (BNN), and it quantized the weight and activation value to

±1 to achieved the purpose of compressing the network through binarization⁽¹³⁾; Wen et al. proposed Force regularization to coordinate more weight information into a low-rank space to compress the network⁽¹⁴⁾. Howard et al. proposed MobileNet, it splits ordinary convolution into depth-wise convolution (DWC) and point-wise convolution (PWC), and reduces the number of multiplications to reduce the amount of model calculations (15). Buciluă et al. proposed knowledge distillation to train a compression model of a strong classifier with pseudo-data labels and copy the output of the original classifier⁽¹⁶⁾. Polino et al. proposed a quantitative training method that adds knowledge distillation loss. There are floating-point models and quantized models. The quantized model is used to calculate the forward loss, and the gradient is calculated to update the floating-point model. Before each forward calculation, it updates the quantization model with the updated floating-point model⁽¹⁷⁾. The quantification of the model to a special bit width requires the design of a special system architecture, and it is not very flexible. The low-rank decomposition operations are costly. Knowledge distillation, compact networks, etc. need to build new small networks, and it is more cumbersome. Li et al. proposed to calculate the L1 norm of the convolution kernel, filter out the feature map corresponding to the convolution kernel with the smaller L1 norm, and then train the network after pruning⁽¹⁸⁾. The current pruning methods are mostly performed on heavyweight models such as VGG and ResNet, and very few on MobileNet, ShuffleNet, etc. Aiming at the deficiencies of the above methods, this paper proposes a rice disease recognition system using lightweight CNN MobileNetV2 to prevent the large-scale occurrence of rice diseases. The system adopts the method of channel pruning to further compress the model. It reduces the amount of model calculations and facilitates lightweight deployment.

3. Proposed Work

The rice disease identification system proposed in this paper is mainly completed in the following three steps: dataset production, MobileNetV2 model training and compression, and model deployment. The specific process is shown in Figure 1.

3.1 Rice disease dataset

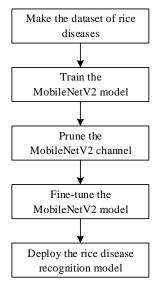


Fig. 1. System implementation steps.

After reading and searching a large number of related agricultural literatures, this paper selects five common rice diseases as the target categories, including three major diseases of rice: sheath blight, rice blast, and bacterial blight, as well as rice smut and brown spot. As we all know, CNN can automatically extract features. What training CNN really does is to adjust parameters and find the best point with lower model loss so that the network can map input (such as images) to output (such as labels). The amount of network parameters is directly proportional to the complexity of the task. Therefore, training CNN requires a large amount of sample data. The larger the number of samples, the better the effect of the trained model and the stronger the generalization ability of the model. Through field shooting and data searching, we found that the image data of rice disease is less. When the dataset is small, too many parameters will fit all the characteristics of the dataset, rather than the commonality between the data. In order to prevent the model from overfitting, accelerate the convergence speed of the model, and enhance the robustness of the model, this paper first preprocesses rice disease data, including data enhancement operations such as random cropping, random rotation, and horizontal flipping, and then normalizes the entire dataset.

3.2 MobileNet model

MobileNet describes an efficient new network architecture. It allows the direct construction of small models with low latency and is easy to meet the requirements of embedded devices. MobilenetV1 uses depth-wise separable convolution (DSC). The core is to

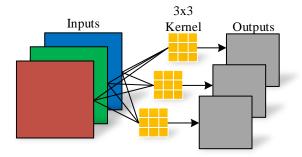


Fig. 2. Standard convolutional layer.

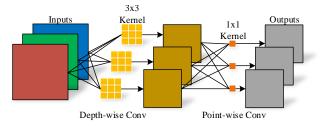


Fig. 3. Depth-wise separable convolutions with depth-wise and point-wise layers.

factorize the original standard convolution operation into DWC and 1x1 PWC. DWC filters the input channels without increasing the number of channels. PWC connects different channels, and it can increase the number of channels. Through this decomposition method, the amount of calculation can be significantly reduced. The standard 3x3 convolution and the DSC structure used in MobileNet are shown in Figure 2 and Figure 3 respectively.

Assuming that the input feature map size is (DF, DF, M), the traditional convolution kernel parameters are (DK, DK, M, N). The corresponding DWC convolution kernel parameters are (DK, DK, 1, M), and the PWC convolution kernel parameters are (1, 1, M, N). Where DF is the input size, DK is the convolution kernel size, M is the number of input channels, and N is the number of output channels. The calculation formulas of traditional convolution and DSC are respectively as follows:

$$DK \times DK \times M \times N \times DF \times DF$$
 (1)

$$DK \times DK \times M \times DF \times DF + M \times N \times DF \times DF$$
 (2)

Compared with traditional convolution, when a 3x3 convolution kernel is used, DSC can reduce the amount of calculation by 8 times. MobileNetV2 is an improvement based on MobileNetV1. The difference between the two is shown in Figure 4.

Fig. 4. Improvements of MobileNetV2 compared to MobileNetV1.

Both of them use DWC and PWC to extract features. The calculation characteristics of DWC determine that it does not have the ability to change the number of channels. If the number of channels given by the upper layer is small, DWC can only extract features in low-dimensional space, and the effect is not good enough. MobileNet V2 added a new PWC before DWC to improve this problem. The new PWC is specially used to increase the dimension. MobileNet V2 adds a new PWC before DWC to increase the dimension specifically. It defines the dimensionality coefficient t=6, and then performs convolution and dimensionality reduction t=10 migrove this problem.

(a) Model training

Nowadays, CNN is more and more popular, such as AlexNet, VGG, ResNet, etc. Although the use of these networks for recognition tasks is effective, the model has a large amount of parameters, calculations, and memory footprint, and is not suitable for running on mobile terminals and embedded devices. This paper puts the rice disease dataset into the MobileNetV2 network for training and saves the best model.

(b) Model compression

At this stage, the work of building small and efficient neural networks can usually be divided into two types of work: pre-training model compression and small model direct training. MobileNetV2 reconstructs the model from the perspective of DSC, reduces the size of the model, and reduces the amount of parameters and calculations. This paper uses parameter pruning to compress MobileNetV2, and trims redundant parameters that do not contribute much to the final output to further improve the operating efficiency of the model. According to different sparsity, pruning can be implemented at different levels, for example, weight level, kernel level, channel level or layer level. Fine-grained level (such as weight level) sparsity provides the highest flexibility, versatility leads to higher compression ratios, but it usually requires special software or hardware accelerators to quickly infer sparse models. On

the contrary, coarse-grained (such as layer level) sparsity does not require special packages to obtain inference acceleration, and it is less flexible because it needs to trim some complete layers. In contrast, channel level sparsity provides a good compromise between flexibility and ease of implementation, so this paper uses channel pruning to perform model compression.

This paper uses L1 regularization to achieve sparse connections. L1 regularization refers to adding a norm penalty to the original loss function. The L1 norm is the absolute value of the weight parameter. Sparsity is equivalent to a feature selection for the model, leaving only some of the more important features. The L1 regularization formula is as follows:

$$L(X, y) = E_{in}(X, y) + \lambda \|\omega\|_{1}$$
(3)

Where X is the input data, y is the label, L is

the total loss function, E is the experience loss, ω is the parameter of the model, and λ is the hyperparameter used to adjust the relative contribution of the norm penalty and the experience loss. When predicting or classifying, so many features are obviously difficult to choose. But if the model obtained by substituting these features is a sparse model, it means that only a few features contribute to the model, and most of the features contribute little. At this time, we can only focus on the features with non-zero coefficients. L1 regularization helps to generate a sparse weight matrix, and it can be used for feature selection. Channel pruning requires designing evaluation criteria for the importance of parameters. The key lies in the selection of the scaling factor. The function of the scaling factor is to select the channel. If batch normalization (BN) layer is not used and a zoom layer is added after the convolutional layer, the value of the zoom factor has no meaning for evaluating the importance of a channel, because the convolutional layer and the zoom layer are just a linear transformation. If a zoom layer is added before the BN layer, the effect of the zoom layer will be completely masked by the BN layer. If a scaling layer is added after the BN layer, there will be two consecutive scaling factors for each channel. Therefore, this paper directly uses the scaling factor γ in the BN layer as a factor to evaluate the contribution of the upper layer output (lower layer input). That is, the smaller γ , the less important the corresponding neuron, and it can be cut out⁽²⁰⁾. The BN layer formula is as follows:

$$y^{(k)} = \gamma^{(k)} x^{(k)} + \beta^{(k)}$$
 (4)

Among them, k is the number of layers, x is the neurons of this layer, y is the output of this layer, y and β are the learnable parameters. According to the set channel pruning ratio, calculate the number of channels to be clipped. The zoom factor is sorted by size to determine which indexed channels of each layer should be pruned. After cutting the channel with small weight, you can finally get a compact CNN model with fewer parameters and small runtime memory. The essence of channel pruning is to cut all the input and output connections related to this channel, and some eigenvalues are easy to lose, and this will cause the accuracy of the model to drop to a certain extent. Therefore, this paper puts the rice disease dataset into the pruned model to fine-tune the parameters to improve the accuracy of the model.

3.3 Model deployment

The fine-tuned lightweight model will be deployed to a high-performance computer to realize real-time shooting and identification of five major rice diseases. The system interface will display the real-time images taken by the camera and the rice disease identification results, and output the parameter information of the system operation. At the same time, the rice disease identification results will be uploaded and saved to the cloud database to facilitate researchers to view historical data.

4. Experiment And Discussion

In order to verify the feasibility of the system, a large number of experiments and evaluations were carried out. The computer processor used in this experiment is Intel core i7-9750H, the graphics card is NVIDIA GTX1660Ti, the memory capacity is 16G, the hard disk capacity is 512G, the operating system and environment are Windows10, the framework used is Pytorch, and the writing language is Python.

4.1 Model training

After data enhancement, normalization and other preprocessing operations, this paper obtained 12,612 sample images, and all image data were labeled. Figure 5 shows the established rice disease dataset and some samples. In this paper, 30% of the dataset is divided into the

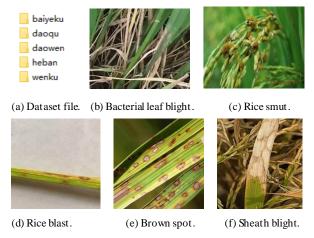


Fig. 5. Rice disease dataset.

test set, and the remaining 70% is the training set. Put it into Mobile Net V2 to train the original model and save the best model obtained through training. The test accuracy rate reaches 90.68%.

4.2 Channel pruning

In this paper, the channel pruning ratio is gradually increased from 0.1 to 0.9 with an interval of 0.1. At different ratios, the accuracy of the model with only pruning and the accuracy of the model with fine-tuning after pruning are shown in Figure 6.

As the proportion of channel pruning continues to increase, the accuracy of the pruned MobileNetV2 model continues to decrease. Starting from 0.5, the decreasing proportion tends to increase. It can be seen from the figure that when the pruning ratio is 0.1 to 0.5, after fine-tuning, the accuracy of the model has increased significantly, and the accuracy of the best model even exceeds 0.16% of the original model. However, when the proportion of pruning is

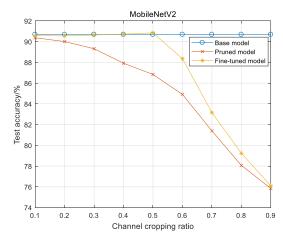


Fig. 6. The accuracy of models under different channel pruning scales.

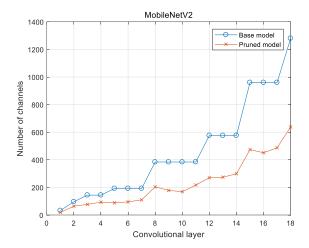


Fig. 7. Comparison of MobileNetV2 channel numbers before and after pruning.

too large, the accuracy of the model is not greatly improved after fine-tuning. This is because there are too many missing features and the damage to the model cannot be recovered. After experiments, this paper chooses the pruning ratio to be 0.5, and prunes the channel with a small scaling factor. The number of channels in each layer of MobileNetV2 before and after pruning is shown in Figure 7. After finishing the pruning, fine-tune the model and save the best model.

Table 1 compares the memory footprint, floating point operations per second (FLOPS) and parameter amount of the MobileNetV2 model before and after channel pruning. It can be seen that the channel pruning reduces the memory usage of the model by 74%, the FLOPS reduces by 49%, and the parameter amount is reduced by 50%. A model with fewer parameters and a more streamlined structure is obtained, and it has a certain promotion effect on the mobile terminal deployment of the MobileNetV2 model.

4.3 System Architecture

In this paper, the lightweight MobileNetV2 model is deployed on a computer with NVIDIA GTX1660Ti graphics card, and a rice disease identification system interface is designed. As shown in Figure 8, the image taken by the camera and the rice disease identification result can be displayed in real time, and the parameter

Table 1. Parameter comparison.

Tuole 1. Turumeter comparison.			
	Model size/M	FLOPS/M	Parameter/M
Base model	17.30	313.47	2.23
Pruned model	4.41	158.84	1.11



Fig. 8. Main interface of rice disease identification system.

information of the system operation, including the current frame rate, image size, running time, file saving path, etc., will also be output on the interface.

As shown in Figure 9, this paper uses SQL Server to design a database to save the rice disease identification results. The database saves the running time of the system, the names of rice disease images and the recognition results, and it is convenient for researchers to view and analyze historical data.

5. Conclusions

This paper proposes a rice disease recognition system using lightweight MobileNetV2. A dataset containing five main rice diseases is produced, and the MobileNetV2

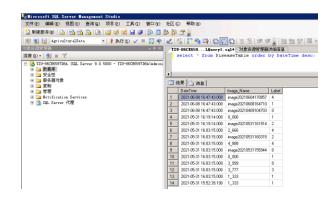


Fig. 9. Database query history record interface.

model is channel pruned. Finally, the model is fine-tuned to improve accuracy. Experiments have shown that this method can reduce the model's memory footprint by 74%, FLOPS by 49%, and the number of parameters by 50%. The accuracy of the best model is increased by 0.16% compared to the original MobileNetV2 model. At the same time, this paper deploys the MobileNetV2 lightweight model, uploads and saves the recognition results to the cloud database, so that researchers can view and analyze historical data.

6. Future Work

This paper realizes the lightweight of MobileNetV2, and uses this model to identify rice diseases. It has a positive effect on the mobile terminal deployment of MobileNetV2 and the prevention of the large-scale occurrence of rice diseases. The next step will be to supplement and improve the rice disease dataset and continue to improve the accuracy of the model. In addition, other model compression methods will be combined to further streamline the model without affecting the completion of the task. Speed up the operation of the model and deploy it to the mobile terminal for optimization.

References

- (1) Liu Tingting, Wang Ting, Hu Lin, "Rhizocotonia Solani Recognition Algorithm Based on Convolutional Neural Network", Chin J Rice Sci, Vol. 33, No. 1, pp. 90-94, 2019 (in Chinese)
- (2) Zhou Bin, "The development status, problems and strategic countermeasures of smart agriculture in China", Agricultural Economy, Vol. 4, No. 1, pp. 6-8, 2018 (in Chinese)
- (3) Cai Ruichu, Zhong Chunrong, Yu Yang, Chen Bingfeng, Lu Ye, Chen Yao, "CNN quantization and compression strategy for edge computing applications", Journal of Computer Applications, Vol. 38, No. 9, pp. 2449-2454, 2018 (in Chinese)
- (4) Wang Jun, Feng Suncheng, Cheng Yong, "Research progress of lightweight network architecture in deep learning", Computer Engineering, pp. 1-14, 2021 (in Chinese)
- (5) Guan Zexin, Tang Jian, Yang Baojun, Zhou Yingfeng, Fan Deyao, Yao Qing, "Study on Recognition Method of Rice Disease Based on Image", Chin J Rice Sci, Vol. 24, No. 5, pp. 497-502, 2010 (in Chinese)

- (6) Liu Tao, Zhong Xiaochun, Sun Chengming, Guo Wenshan, Chen Yingying, Sun Juan, "Recognition of Rice Leaf Diseases Based on Computer Vision", Scientia Agricultura Sinica, Vol. 47, No. 4, pp. 664-674, 2014 (in Chinese)
- (7) M. Suresha, K. N. Shreekanth and B. V. Thirumalesh, "Recognition of diseases in paddy leaves using knn classifier," 2017 2nd International Conference for Convergence in Technology (I2CT), pp. 663-666, 2017
- (8) XU Shipu, LI Linyi, JIA Jinyuan, WANG Yunsheng, LIU Chang, LIU Yong, MA Chao, "AR-assisted intelligent analysis and identification system for mobile rice diseases based on HOG-SVM", JOURNAL OF GRAPHICS, Vol. 42, No. 3, pp. 454-461, 2021 (in Chinese)
- (9) M. J. Hasan, S. Mahbub, M. S. Alom and M. Abu Nasim, "Rice Disease Identification and Classification by Integrating Support Vector Machine With Deep Convolutional Neural Network", 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), pp. 1-6, 2019
- (10) G. Zhou, W. Zhang, A. Chen, M. He and X. Ma, "Rapid Detection of Rice Disease Based on FCM-KM and Faster R-CNN Fusion", in IEEE Access, vol. 7, pp. 143190-143206, 2019
- (11) Gao Han, Tian Yulong, Xu Fengyuan, Zhong Sheng, "Survey of Deep Learning Model Compression and Acceleration", Journal of Software, Vol. 32, No. 1, pp. 68-92, 2021 (in Chinese)
- (12) V. Lebedev and V. Lempitsky, "Fast ConvNets Using Group-Wise Brain Damage", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2554-2564, 2016
- (13) Courbariau x M, Bengio Y, David J P. "BinaryConnect: Training Deep Neural Networks with binary weights during propagations", CoRR, pp. 3123-3131, 2015
- (14) W. Wen, C. Xu, C. Wu, Y. Wang, Y. Chen and H. Li, "Coordinating Filters for Faster Deep Neural Networks", 2017 IEEE International Conference on Computer Vision (ICCV), pp. 658-66, 2017
- (15) Howard A G, Zhu M, Chen B, et al. "Mobile Nets: Efficient Convolutional Neural Networks for Mobile Vision Applications", CoRR, 1704.04861, 2017
- (16) Buciluă, C., Caruana, R., & Niculescu-Mizil, A. "Model compression", In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 535–541,

- 2006
- (17) Polino A, Pascanu R, Alistarh D. "Model compression via distillation and quantization", CoRR, 1802.05668, 2018
- (18) Li H, Kadav A, Durdanovic I, et al. "Pruning Filters for Efficient ConvNets", CoRR, 1608.08710, 2016
- (19) M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. Chen, "Mobile Net V2: Inverted Residuals and Linear Bottlenecks", 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4510-4520, 2018
- (20) Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan and C. Zhang, "Learning Efficient Convolutional Networks through Network Slimming", 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2755-2763, 2017