IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received May 2, 2018, accepted May 29, 2018, date of publication June 6, 2018, date of current version June 26, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2844405

Identification of Maize Leaf Diseases Using Improved Deep Convolutional Neural Networks

XIHAI ZHANG, (Member, IEEE), YUE QIAO, FANFENG MENG, CHENGGUO FAN, AND MINGMING ZHANG
School of Electronic Engineering and Information, Northeast Agricultural University, Harbin 150030, China

Corresponding author: Xihai Zhang (xhzhang@neau.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 31101080, in part by the China Postdoctoral Science Foundation through the Project under Grants 2015M580254 and 2017T100221, in part by the Heilongjiang Postdoctoral Science Foundation under Grant LBH-Z15011, in part by the Harbin Science and Technology Innovation Youth Talents Special Fund under Grant 2015RQQXJ094, and in part by the Academic Backbone Project of Northeast Agricultural University under Grant 15XG12.

ABSTRACT In the field of agricultural information, the automatic identification and diagnosis of maize leaf diseases is highly desired. To improve the identification accuracy of maize leaf diseases and reduce the number of network parameters, the improved GoogLeNet and Cifar10 models based on deep learning are proposed for leaf disease recognition in this paper. Two improved models that are used to train and test nine kinds of maize leaf images are obtained by adjusting the parameters, changing the pooling combinations, adding dropout operations and rectified linear unit functions, and reducing the number of classifiers. In addition, the number of parameters of the improved models is significantly smaller than that of the VGG and AlexNet structures. During the recognition of eight kinds of maize leaf diseases, the GoogLeNet model achieves a top - 1 average identification accuracy of 98.9%, and the Cifar10 model achieves an average accuracy of 98.8%. The improved methods are possibly improved the accuracy of maize leaf disease, and reduced the convergence iterations, which can effectively improve the model training and recognition efficiency.

INDEX TERMS Deep learning, deep convolutional neural networks, identification, image processing, leaf diseases.

I. INTRODUCTION

Maize is an important food and feed crop. Its plant area and total output are the largest in the world except for rice and wheat [1]. However, in recent years, the number of species of maize diseases and the degree of harm they cause have increased, mainly due to changes in cultivation systems, the variation of pathogen varieties, and inadequate of plant protection measures. Generally, there are eight types of common leaf diseases, including Curvularia leaf spot, dwarf mosaic, gray leaf spot, northern leaf blight, brown spot, round spot, rust, and southern leaf blight [2]–[6]. Most seriously, maize leaf disease is hazardous and will affect maize production and people's lives.

Maize leaf diseases have various symptoms. It may be more difficult for inexperienced farmers to diagnose diseases than for professional plant pathologists [7]. As a verification system in disease diagnostics, an automatic system that is designed to identify plant diseases by the plant's appearance and visual symptoms could be of great help to farmers. Many efforts have been applied to the quick and accurate diagnosis of leaf diseases. By using digital image processing techniques, support vector machine (SVM), neural networks and other methods, we can detect and classify leaf diseases [8]–[13]. An SVM - based multi - classifier was proposed by Song *et al*. [8] and was applied to identify a variety of maize leaf diseases. The best recognition accuracy was 89.6%. The method of classification using SVM is only applicable to small samples, for a large number of samples, it cannot achieve high recognition accuracy. Chen and Wang [9] proposed a method for the identification of maize leaf diseases based on image processing technology and a probabilistic neural network (PNN). The best recognition accuracy of this method was 90.4%. However, for the PNN classifier, the identification accuracy and speed of this method decreases as the number of training samples increases. A method of maize leaf disease identification based on adaptive weighting multi-classifier fusion was proposed by Xu *et al.* [10]. Seven common types of maize leaf disease were tested by this method. The average recognition rate was 94.71%. Wang *et al*. [11] Qi *et al.* [12], and Zhang [13] proposed different methods using digital image processing techniques based on Fisher discriminant, Retinex algorithm combined with principal component analysis (PCA) and SVM, and quantum neural network (QNN) and combination features for identification of maize leaf disease. The highest recognition accuracy of these studies was 95.3%, but fewer maize diseases were involved in these methods. Different methods are used to identify maize diseases [8]–[13], the best recognition accuracy was 95.3%, which cannot meet the current requirements for high recognition accuracy. Therefore, in the follow - up study, we should focus on how to improve the identification accuracy.

Deep learning has made tremendous advances in the past few years [14]–[24]. It is now able to extract useful feature representations from a large number of input images. Deep learning provides an opportunity for detectors to identify crop diseases in a timely and accurate manner, which will not only improve the accuracy of plant protection but also expand the scope of computer vision in the field of precision agriculture. Lu *et al.* [25] used different pooling operations, filter sizes, and algorithms to identify 10 common rice diseases. The proposed convolutional neural networks (CNNs) - based model achieved an accuracy of 95.48%. Dechant *et al.* [26] trained CNNs to automatically identify northern leaf blight of maize. This approach addressed the challenge of limited data and the myriad irregularities that appear in images of field - grown plants. The identification scheme achieved an accuracy of 96.7%. Some researchers [27]–[29] can improve the identification accuracy of plant diseases to a certain extent by using different convolution neural network models and changing the ratio of training set size to testing set size. These studies [25]–[29] have obtained better results, but more parameters and longer training convergence times have a negative effect on the recognition rate. To obtain a highly maize leaf disease identification accuracy, it is highly significant to design a recognition model with fewer parameters and higher recognition accuracy.

In this study, two improved deep convolution neural network models, GoogLeNet and Cifar10, are presented to increase the recognition accuracy of maize leaf diseases and improve the traditional identification methods with long convergence times and large numbers of model parameters. The two models that are used to train and test 9 kinds of maize leaf images are obtained by adjusting the model parameters, changing the pooling combinations, adding the dropout operation and rectified linear unit (*Relu*) function, and reducing the number of classifiers. Finally, the experimental results are compared with those of the unmodified model.

The rest of this paper is organized as follows. In Section II, it mainly depicts the collection and processing of the image dataset, in addition, two kinds of CNNs structures, GoogLeNet and Cifar10, are introduced in detail. At the same time, some basic concepts and experimental parameters involved in the structure are described in Section II. Then, in Section III, the original and improved structure of GoogLeNet and Cifar10 were used to realize the identification of maize leaf diseases, the accuracy, loss, and parameters are analyzed in detail. In this section, the method proposed in this paper is compared with the traditional method and the relative deep learning method for maize leaf disease. Finally, the paper is concluded in Section IV.

II. MATERIALS AND METHODS

A. DATASET

An appropriate dataset is required at all stages of object recognition research, starting from the training phase to evaluating the performance of recognition algorithms. A total of 500 images are collected from different sources, such as the Plant Village and Google websites, including different periods of occurrence of maize leaf diseases, which are divided into 9 different categories. There are 8 categories representing infected maize leaves and a category representing healthy leaves. Eight kinds of maize leaf diseases are shown in Fig. 1: Curvularia leaf spot, dwarf mosaic, gray leaf spot, northern leaf blight, brown spot, round spot, rust, and southern leaf blight; these are the main diseases investigated in this study.

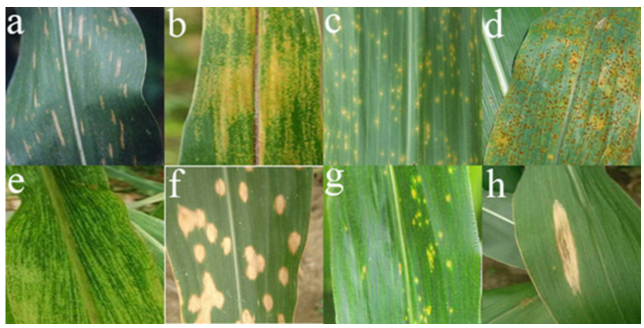


FIGURE 1. Eight common maize leaf diseases a: southern leaf blight; b: brown spot; c: Curvularia leaf spot; d: rust; e: dwarf mosaic; f: gray leaf spot; g: round spot; h: northern leaf blight.

All images downloaded from different sources were cleaned by a developed Python script that applied a comparison procedure. The script removed duplicates by comparing the images' metadata: name, size and date. After automated removal, the images were assessed several times by human experts.

B. AUGMENTATION

Training CNNs requires substantial data. The more data the CNNs has to learn, the more features it can obtain. Since the original leaf image dataset collected in this study is not sufficient, it is necessary to expand the dataset by different methods to distinguish the different disease categories. After the original images are initialized, additional versions are created by rotating the images 90°, 180°, and 270°; by mirroring each rotated image; by cutting the center of the image by the same size; and by converting all processed images to grayscale. The dataset is expanded by the above methods, which helps in reducing over - fitting during the training stage [30]. Partially converted images are shown in Fig. 2. In total, the maize leaf dataset contains 3060 images

– 2248 (80%) for training and 612 (20%) for testing. The dataset for maize leaf disease images is shown in Table 1.

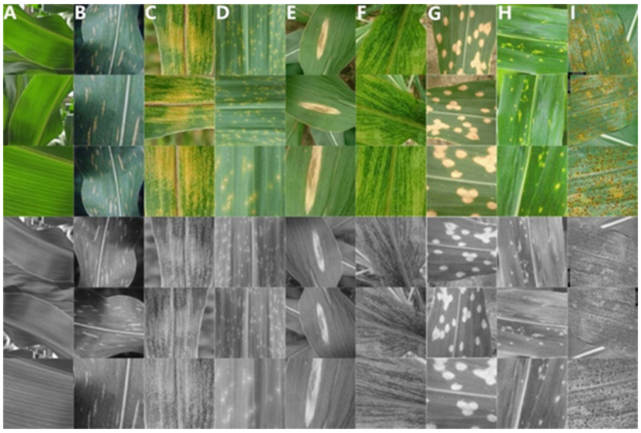


FIGURE 2. Part of the image samples after augmentation process Part A shows a healthy maize leaf after rotation, cutting, and grayscale. Part B-I show eight kinds of maize leaf disease images.

TABLE 1. Dataset for maize leaf disease image.

Class	Number of original images	Total number of images	Number of images for training	Number of images for testing
Northern leaf blight	55	360	288	72
Southern leaf blight	50	360	288	72
Rust	65	360	288	72
Brown spot	55	360	288	72
Round spot	50	360	288	72
Curvularia leaf spot	45	320	256	64
Gray leaf spot	65	320	256	64
Dwarf mosaic	55	360	288	72
Healthy leaf	60	260	208	52
Total	500	3060	2448	612

C. IMAGE PREPROCESSING AND LABELLING

To improve feature extraction and increase consistency, the images in the dataset for the deep CNNs classifier are preprocessed before the model is trained. One of the most significant operations is the normalization of image size and format. In this study, all images are resized to 224 × 224 pixels and 32 × 32 dots per inch, which are automatically computed by Python scripts based on the OpenCV framework.

In the interest of confirming the accuracy of the classes in the dataset, agricultural experts examined leaf images grouped by a keyword search and labeled all the images with the appropriate disease acronym. It is well known that it is essential to use accurately classified images for the training and validation dataset. Only in that can may an appropriate and a reliable model be developed. In this stage, various classes of the dataset as well as the training set and the testing set are marked.

D. CONVOLUTIONAL NEURAL NETWORKS

Caffe [31], a framework based on C++ language designed specifically for deep learning and CNNs - related algorithms, has many advantages, such as faster updates and flexible expansibility. It provides a complete toolkit for training, testing and fine - tuning. The deployment models can run on both central processing units (CPUs) and graphics processing units (GPUs). Integrating Caffe with the cuDNN library can accelerate Caffe models [32].

GoogLeNet structure has 22 layers, which is characterized by going deeper. The GoogLeNet model [33] has more features than previous deep learning structures, because of increases in depth, width, and training data. Nevertheless, GoogLeNet has fewer parameters than the VGG and AlexNet models, which are flexible applications of the network - in - network concept. GoogLeNet uses sparse network structures to improve the disadvantages of over - fitting and over - occupying computing resources. It uses the pyramid model to increase the width and puts forward the concept of an "Inception Module." The main idea of the "Inception Module" is to use dense components to approximate the optimal local sparse structure. A total of nine inception modules are used in the GoogLeNet structure. Each module includes multiple parallel convolutional layers with a size of 1 × 1, 3 × 3, 5 × 5 and a max pooling layer for the capture of different features simultaneously. The improved "Inception Module" can be seen in Fig. 3. Three classifiers are able to measure the top - 1 accuracy, top - 5 accuracy, and system loss. To make the model more adaptable to the sample dataset in this paper, only the first classifier is used to train and test the 9 samples, which reduces the number of model parameters and the time required for convergence without affecting the recognition accuracy. Meanwhile, to improve the identification accuracy of the model, only the top - 1 accuracy is measured in this experiment.

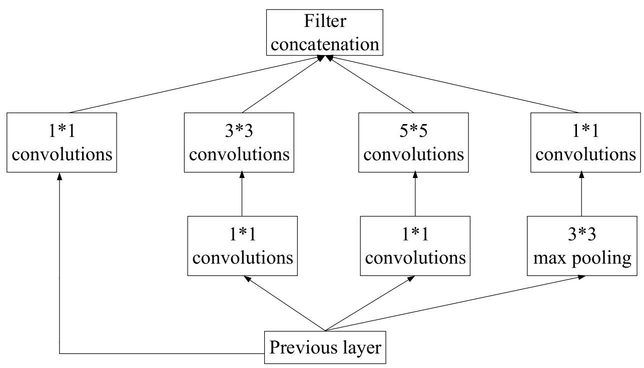


FIGURE 3. The inception module.

The Cifar10 structure is optimized in this study. Specifically, the network contains three convolutional layers, two fully connected layers, and a loss layer. After each convolutional layer in the model, there is a pooling layer and a *Relu* operation. The relationship between different pooling combinations and the recognition accuracy will be explored in this study. The identification accuracy will be improved by adding dropout and *Relu* between the two fully connected layers. A dropout operation with an appropriate probability value can prevent over - fitting of CNNs. The *Relu* function can make the network learn relatively sparse features from the dataset, which creates the effect of automatic dissociation. The modified Cifar10 model is used to train the maize leaf image dataset, and subsequently, the identification accuracy and loss of the model are tested.

1) CONVOLUTION

Convolution is the most important operation in CNNs. The convolution calculation of the two - dimensional image can be mapped to the continuous sliding convolution window to obtain the corresponding convolution value.

In CNNs, each feature map is convoluted by multiple input feature graphs. For an input x of the ith convolutional layer, it computes as (1),

$$h_{ic} = f(W_i * x), \tag{1}$$

where $*$ represents the convolution operation, W_i represents the convolution kernels of the layer, and f represents the activation function. $W_i = [W_i^1, W_i^2, \ldots, W_i^K]$, K is the number of convolution kernels of the layer. Each kernel W_i^K is an $M \times M \times N$ weight matrix with M being the window size and N being the number of input channels [28].

2) ACTIVATE FUNCTION

The *Relu* activation function is an unsaturated nonlinear function that can receive signals by simulating brain neurons. Saturated nonlinear function, such as *Sigmoid* and *Tanh*, have worse performance than unsaturated nonlinear functions when training a network.

In this test, the *Relu* activation function will be added in the Cifar10 model, to prevent the problem of gradient dispersion while accelerating network training and to increase the identification accuracy.

3) POOLING

As the number of convolutional layers increases, the parameters of the network will increase exponentially. The pooling operation can effectively reduce the number of network parameters. To reduce the parameters in all regions, the pooling operation is performed by calculating the statistical characteristics of a region in order to represent the entire region's characteristics.

The effect of different pooling combinations on the identification accuracy of Cifar10 will be explored in this study.

4) DROPOUT

Srivastava *et al.* [34] suggested that dropout can alleviate the situation of fewer training samples in neural networks by preventing the synergies of certain features. For each input sample, the corresponding network structure is different, but all of these different network structures share the weight of hidden nodes at the same time, so that different samples correspond to different models.

To prevent over - fitting and improve the generalization of the model, a dropout operation will be added in the Cifar10 structure in this test.

5) LOSS FUNCTION

The loss function measures the discrepancy between the predicted result and the label of the input, which is defined as (2),

$$E(W) = -1/n \sum_{x_i=1}^{n} \sum_{k=1}^{K} [y_{ik} \log P(x_i = k) + (1 - y_{ik}) \log(1 - P(x_i = k))], \tag{2}$$

where W indicates the weighting matrixes of the convolutional and fully connected layers, n indicates the number of training samples, i is the index of training samples, and k is the index of classes. If the ith sample belongs to the kth class, $y_{ik} = 1$; else $y_{ik} = 0$. $P(x_i = k)$ is the probability of input x_i belonging to the kth class that the model predicts, which is a function of the parameters W. Therefore, the loss function takes W as its parameters.

Network training aims to find the value of W that minimizes the loss function E. In this study, we use a stochastic gradient descent (SGD) algorithm where W is iteratively updated as (3),

$$W_k = W_{k-1} - \alpha(\partial E(W)/\partial W), \tag{3}$$

where α is the learning rate, which is a very important parameter that determines the step size of the learning. The k is the index of classes, its meaning is the same as (2). The value of learning rate should be carefully evaluated [28].

E. HYPER PARAMETERS

The improved Cifar10 and GoogLeNet models' hyper parameters are shown in Table 3 compared with the original one in Table 2. By changing the base learning rate, it can affect the identification accuracy of the network. All experiments are done using the GPUs. The models are optimized by stochastic gradient descent (SGD) algorithm. The method of batch training is to divide the training set and the testing set into multiple batches. Each batch consists of training 10 images. The initial learning rate of the Cifar10 model is fixed at 0.0002. The initial learning rate for the GoogLeNet model is 0.001 and decremented by 0.96 times.

TABLE 2. The original hyper parameters.

Cifar10		GoogLeNet	
Name	Parameters	Name	Parameters
Solver type	SGD	Solver type	SGD
Base learning rate	0.001	Base learning rate	0.01
Momentum	0.9	Momentum	0.9
Learning rate policy	Fixed	Learning rate policy	Step
Weight decay	0.004	Weight decay	0.0002
Batch size	100	Gamma	0.96

TABLE 3. The improved hyper parameters.

Cifar10		GoogLeNet	
Name	Parameters	Name	Parameters
Solver type	SGD	Solver type	SGD
Base learning rate	0.0002	Base learning rate	0.001
Momentum	0.9	Momentum	0.9
Learning rate policy	Fixed	Learning rate policy	Step
Weight decay	0.004	Weight decay	0.0002
Batch size	10	Gamma	0.96

F. EQUIPMENT

The Caffe framework, Visual Studio development environment and Python language are used to train and test the complete model on a computer. The relevant parameters are shown in Table 4.

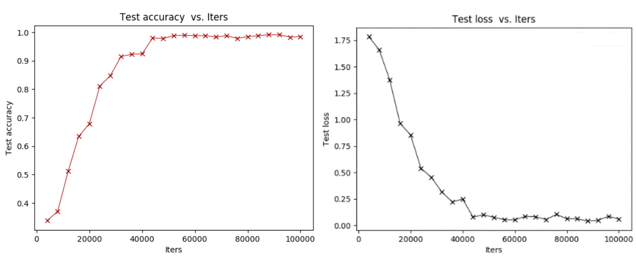
TABLE 4. Hardware and software parameters.

Name	Parameter
Memory	8Gb
Processor	Intel Core i7-4790 CPU @3.40GHz 3.41GHz
Graphics	NVIDIA GeForce GTX 960
Operating system	Windows 10 64 bits
Development environment	Visual Studio 2013, OpenCv
Language	Python, C++

III. RESULTS AND DISCUSSION

A. Goollenet MODEL

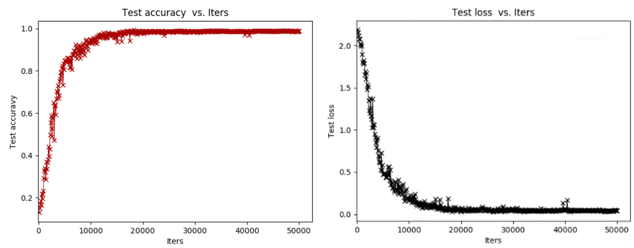
The initial learning rate of the original GoogLeNet model is 0.001, using the "step" method to attenuate the learning rate. After 100000th iterations and classified by the three classifiers, the top - 1 testing accuracy are 98.8%, 98.6%, 98.2%; top - 5 testing accuracy are 99.6%, 99.6%, 99.6%; the loss of the system is 15.8%. Fig. 4 (a) shows the changes of partial top - 1 test accuracy and Fig. 4 (b) shows the curve of the system loss. We can see that the top - 1 identification accuracy and system loss gradually converge after 40000th iterations. The training time and the convergence time of the original model are longer. The original model also has a larger number of parameters.



(a) (b)

FIGURE 4. Experimental results of the original GoogLeNet model.

The first classifier of the GoogLeNet model is used to perform 50000th iterations on 9 samples of the maize leaf dataset in this test. After each 100th iteration, the top - 1 accuracy and the model loss are measured. Fig. 5 (a) shows the changes of top - 1 test accuracy and Fig. 5 (b) shows the curve of the model loss. In this study, the initial learning rate of the GoogLeNet model is 0.001, and the "step" method attenuates the learning rate by 0.96 times every 2000th iterations. As seen from Fig. 5, after 10000th iterations, the top - 1 testing accuracy gradually tends to 1, the loss gradually approaches 0, and both states are stable. Experiments show that the average top - 1 accuracy is 98.9% and the loss is 1.6%, after using the improved GoogLeNet model to train and test the maize leaf image dataset.



(a) (b)

FIGURE 5. Experimental results of the improved GoogLeNet model.

Compared with the original unmodified model, the identification accuracy and system loss of the improved model are better than the original one. The improved model's top – 1 identification accuracy is 0.4% higher than that of the original one, the system loss is 14.2% less than the original one. In Fig. 4 the top - 1 identification accuracy and system loss gradually converge after 40000th iterations, in Fig. 5 after 10000th iterations, the top - 1 testing accuracy gradually tends to 1, the loss gradually approaches 0, and both states are stable. The convergence time have been greatly improved, which can effectively improve the model training and recognition efficiency.

B. Cifar10 MODEL

The *Relu* function and dropout operation will be added between the two fully connected layers of the Cifar10 model. *Relu* function can adaptively learn the parameters of the rectifier and increase accuracy with negligible additional cost. For an input x, the *Relu* activation function is defined as (4),

$$Relu(x) = \begin{cases} 0, & if\ x \leq \\ 0, & if\ x > 0. \end{cases} \tag{4}$$

Dropout operation works by randomly suppressing a certain number of neurons. The suppressed neurons are temporarily not involved in the forward communication of the network. Optimizing the model - related parameters and then initializing the three pooling combinations: Max - Max - Ave (By taking the maximum of the k × k neighborhood in the feature graph, max pooling can calculate the maximum value of the non-overlapping rectangular area for each convolution kernel output. This approach can be used to separate very sparse features, reduce the estimated mean offset error caused by the convolutional layer parameter error, and keep more texture information. The mean pooling is averaged over all the sampling points in the locally accepted domain. It is possible to reduce the error of the variance of the estimated variance increases due to the limited size of the neighborhood, which can retain more image background information.). Considering the fact that different dropout parameters will affect the recognition accuracy, in this test, the relationship between the dropout probability value and the testing accuracy of the improved model is studied. The results are shown in Table 5 (In general, the probability value of the dropout operation is chosen to be 0.5. Since the dataset in this paper is rather special, we try to find the probability value with a greater accuracy near 0.5. From 0.5 to 0.75, several experiments were performed on a 0.05 - percent difference, and it was found that when the probability value was chosen to be 0.65, the best recognition accuracy was achieved within this range.). The maximum testing accuracy of the model is 97.8% when the dropout probability value is 0.65. We fix this value and then experiment with four pooling combinations of three convolutions: Max/Ave/Ave, Max/Max/Ave, Max/Max/Max, Ave/Ave/Ave. The learning rate of this model is fixed at 0.0002. The accuracy and the loss of the model is measured after every 20th iteration, for a total of 50000th iterations. The model's testing accuracy and loss curves are shown in Table 6. As seen from Table 6, the preferred pooling combination is Max - Max - Ave. The original model's testing accuracy and loss are shown in Fig. 6. The improved models' testing accuracies are shown in Fig. 7.

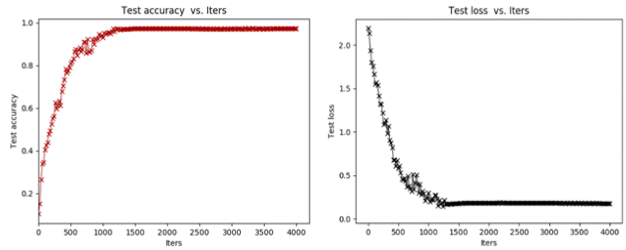
TABLE 5. The relationship between the dropout probability value and testing accuracy.

Dropout probability value	Testing accuracy
0.5	96.2%
0.55	96.8%
0.6	97.2%
0.65	97.8%
0.7	95.2%
0.75	92.6%

TABLE 6. The accuracy and loss of the improved Cifar10 model.

Form of pooling layers	Training accuracy	Testing accuracy	Loss
Max-Ave-Ave	98.2%	85.4%	52.8451%
Max-Max-Ave	98.8%	97.8%	7.6894%
Max-Max-Max	97.7%	97.2%	16.8793%
Ave-Ave-Ave	97.2%	98.4	10.9809%

The original Cifar10 model is used to train the dataset for 4000th iterations. The accuracy and the model loss are measured every 20th iteration. The results are shown in Fig. 6. After 1200th iterations, the testing accuracy and the loss curve tend to be stable. The average testing accuracy of the system is 97.1%, and the loss of the system is 17.8%. According to Table 6 and Fig. 7, the best combination of pooling is Max - Max - Ave, where the dropout parameter is 0.65. In this case, the training accuracy of this model is 98.8%, the testing accuracy is 97.8%, the system loss is 7.6%, and after 20000th iterations, the two curves of Fig. 7 (b) have



(a) (b)

FIGURE 6. Experimental results of the original Cifar10 model.

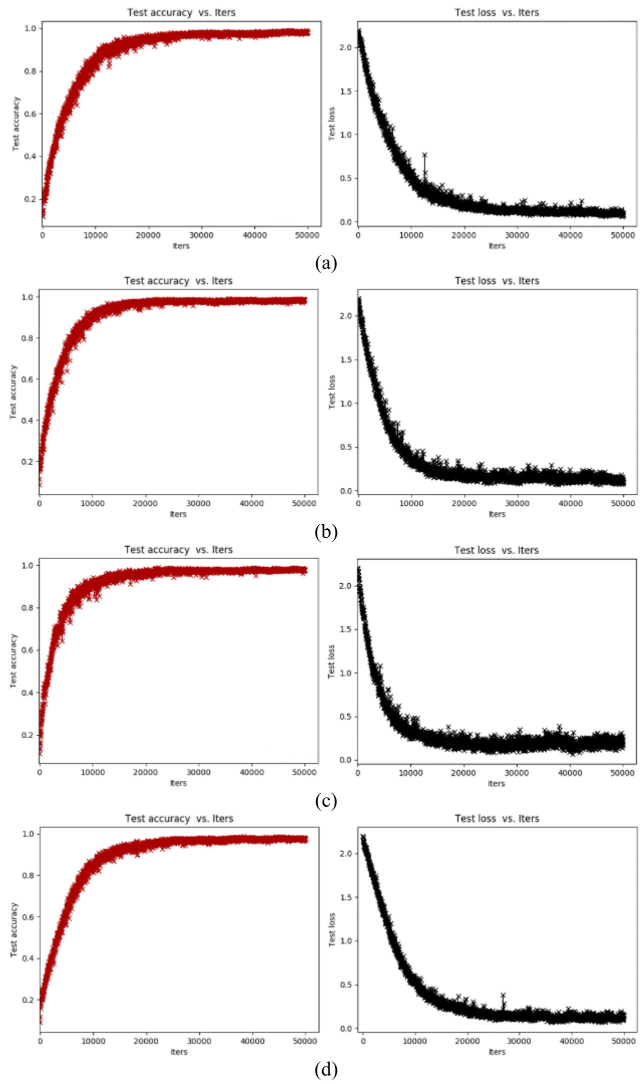


FIGURE 7. Experimental results of the four pooling layer combinations of the Cifar10 model (a) Max-Ave-Ave. (b) Max-Max-Ave. (c) Max-Max-Max. (d) Ave-Ave-Ave.

converged. Therefore, the testing accuracy can be improved by 0.7% and the loss reduced by 10.2% by using the improved Cifar10 model shown in Fig. 8.

In this study, the average top - 1 identification accuracy achieves 98.5% by using the original GoogLeNet structure, the average identification accuracy achieves 97.1% by using

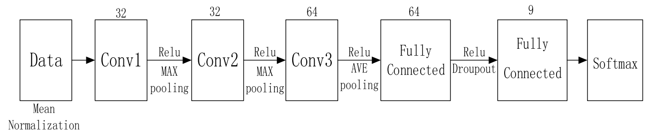


FIGURE 8. The improved Cifar10 model.

the original Cifar10 structure. The two improved deep CNNs models, GoogLeNet and Cifar10, can achieve high identification accuracy, 98.9% and 98.8%, respectively. The improved methods are possibly improved the accuracy of maize leaf disease, and reduced the convergence iterations, which can effectively improve the model training and recognition efficiency. Compared with Song *et al.* and Wang *et al.*'s [8]–[13] methods for maize leaf disease recognition, their research first carries on a series of processing to the image, then extracts the feature, and finally classifies the maize leaf disease by using SVM, PNN or QNN and so on. These research processing steps are more complex and will introduce unnecessary interference at each step. The method proposed in this paper can directly take the image of the dataset as the input of the convolutional neural networks and let it learn and adjust itself to achieve an effective recognition effect. The recognition accuracy and loss are also in a more satisfactory range, and the training and recognition efficiency has been improved.

IV. CONCLUSION

In this study, when identifying 9 types of maize leaves, the two improved deep convolutional neural networks models, GoogLeNet and Cifar10, can achieve high identification accuracy, 98.9% and 98.8%, respectively. When the train - test set is 80 - 20 (80% of the whole dataset used for training, and 20% for testing), the classification algorithms used in this study allow the systems to acquire a diversity of sample conditions with strong robustness. Experiments show that it is possible to improve recognition accuracy by increasing the diversity of pooling operations, the reasonable addition of a *Relu* function and dropout operations, and including multiple adjustments of the model parameters. In future research, we will identify more types of maize diseases and pests and combine new algorithms and other deep learning structures for the training and testing of the model. Meanwhile, in order to enable agricultural producers to make quick and reasonable judgments about crop disease information, the trained model can be combined with mobile devices in a flexible manner.

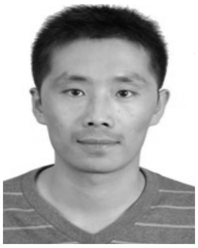
ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful suggestions, which greatly improved the paper.

REFERENCES

[1] G. F. Sprague, *Corn and Corn Improvement*. New York, NY, USA: Academic, 1955, pp. 89–150.
[2] F. M. Latterell and A. E. Rossi, "Gray leaf spot of corn: A disease on the move," *Plant Disease*, vol. 67, no. 8, pp. 842–847, 1983.
[3] L. E. Williams and L. J. Alexander, "Maize dwarf mosaic, a new corn disease," *Phytopathology*, vol. 55, no. 7, pp. 802–804, 1965.
[4] F. C. Dai *et al.*, "Curvularia leaf spot of maize: Pathogens and varietal resistance," *Acta Phytopathologica Sinica*, vol. 28, no. 2, pp. 123–129, Feb. 1998.
[5] X. X. Li *et al.*, "The corn disease remote diagnostic system in China," *J. Food Agricult. Environ.*, vol. 10, no. 1, pp. 617–620, Jan. 2012.
[6] J. M. J. Ward, E. L. Stromberg, D. C. Nowell, and F. W. Nutter, Jr., "Gray leaf spot: A disease of global importance in maize production," *Plant Disease*, vol. 83, no. 10, pp. 884–895, 1999.
[7] S. A. Miller, F. D. Beed, and C. L. Harmon, "Plant disease diagnostic capabilities and networks," *Annu. Rev. Phytopathol.*, vol. 47, no. 1, pp. 15–38, 2009.
[8] K. Song, X. Y. Sun, and J. W. Ji, "Corn leaf disease recognition based on support vector machine method," *Trans. Chin. Soc. Agricult. Eng.*, vol. 23, no. 1, pp. 155-157, Jan. 2007.
[9] L. Chen and L. Y. Wang, "Research on application of probability neural network in maize leaf disease identification," *J. Agricult. Mech. Res.*, vol. 33, no. 6, pp. 145–148, Jun. 2011.
[10] L. F. Xu, X. B. Xu, and H. Min, "Corn leaf disease identification based on multiple classifiers fusion," *Trans. Chin. Soc. Agricult. Eng.*, vol. 31, no. 14, pp. 194–201, 2015.
[11] N. Wang, K. Wang, R. Xie, J. Lai, B. Ming, and S. Li, "Maize leaf disease identification based on fisher discrimination analysis," *Scientia Agricultura Sinica*, vol. 42, no. 11, pp. 3836–3842, 2009.
[12] Z. Qi *et al.*, "Identification of maize leaf diseases based on image technology," *J. Anhui Agricult. Univ.*, vol. 43, no. 2, pp. 325–330, Feb. 2016.
[13] F. Zhang, "Recognition of corn leaf disease based on quantum neural network and combination characteristic parameter," *J. Southern Agriculture*, vol. 44, no. 8, pp. 1286–1290, 2013.
[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
[15] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.
[16] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *Int. J. Robot. Res.*, vol. 34, nos. 4–5, pp. 705–724, 2013.
[17] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning," *Nature Biotechnol.*, vol. 33, pp. 831–838, Jul. 2015.
[18] L. P. Zhang, G. S. Xia, T. Wu, L. Lin, and X. C. Tai, "Deep learning for remote sensing image understanding," *J. Sensors*, vol. 2016, Jun. 2015, Art. no. 7954154.
[19] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[21] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[22] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, 2012, pp. 1097–1105.
[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[24] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[25] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, Dec. 2017.
[26] C. Dechant *et al.*, "Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning," *Phytopathology*, vol. 107, no. 11, pp. 1426–1432, 2017.
[27] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016.
[28] G. Wang, Y. Sun, and J. X. Wang, "Automatic image-based plant disease severity estimation using deep learning," in *Computational Intelligence and Neuroscience*, 2017, pp. 1–8.
[29] D. Oppenheim and G. Shani, "Potato disease classification using convolution neural networks," in *Proc. 11th Eur. Conf. Precis. Agriculture (ECPA)*, 2017, pp. 244–249.

[30] D. M. Hawkins, "The problem of overfitting," *J. Chem. Inf. Comput. Sci.*, vol. 44, no. 1, pp. 1–12, 2004.
[31] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, Orlando, FL, USA, 2014, pp. 675–678.
[32] S. Chetlur *et al.*, "cuDNN: Efficient primitives for deep learning," in *Proc. Deep Learn. Represent. Learn. Workshop*, 2014, pp. 1–9.
[33] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2014, pp. 1–9.
[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

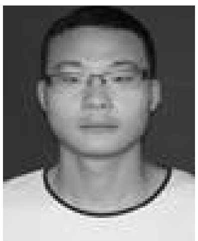


XIHAI ZHANG was born in Harbin, China, in 1979. He received the B.S. and M.S. degrees in electrical engineering from Northeast Agricultural University, Harbin, China, in 2002 and 2005, respectively, and the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2013. He is currently an Associate Professor with the School of Electrical and Information Engineering, Northeast Agricultural University. His current research interests include deep learning and its application to computer vision and pattern recognition, and applications to embedded systems.

YUE QIAO received the B.E. degree in communication engineering from Qufu Normal University, Qufu, China, in 2015. She is currently pursuing the degree in computer application technology with Northeast Agricultural University, Harbin, China. Her current research interest is the application of deep learning to agriculture.

FANFENG MENG received the B.S. in computer science and engineering, the M.S. degree in software engineering, and the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2002, 2005, and 2009, respectively. He is currently a Lecturer with the School of Electrical and Information Engineering, Northeast Agricultural University, Harbin. His current research interests include image target recognition, deep learning applications, and embedded system technology.

CHENGGUO FAN received the B.E. degree in communication engineering from Ludong University, Shandong, China, in 2014. He is currently pursuing the degree in computer application technology with Northeast Agricultural University, Harbin. His current research interest is the application of embedded system.



MINGMING ZHANG received the B.E. degree in automation specialty from the East China University of Technology, Nanchang, in 2017. He is currently pursuing the degree in computer science and technology with Northeast Agricultural University, Harbin. His current interest is intelligent control technology.