

# Identification of Soybean Leaf Diseases via Deep Learning

Qiufeng Wu<sup>1</sup>  · Keke Zhang<sup>2</sup> · Jun Meng<sup>1</sup>

Received: 27 January 2019 / Accepted: 20 July 2019 / Published online: 27 July 2019  
© The Institution of Engineers (India) 2019

**Abstract** We propose a novel approach for identifying soybean leaf diseases in the natural environment by convolutional neural network (CNN). AlexNet, GoogLeNet and ResNet were utilized for transfer learning. Firstly, 27 models were obtained by setting different batch sizes and the number of iterations. Then, the effects of CNN structure on identification performance were explored. The optimal model is based on ResNet and has the highest accuracy of 94.29%. In the parameter settings of the optimal network, the number of iterations and batch size are 1056 and 16, respectively, and the training depth is 140. Overall, the proposed method is effective for identifying soybean leaf diseases in the natural environment.

**Keywords** Soybean leaf diseases · Deep learning · Convolutional neural network · Image recognition

## Introduction

Soybean is one of the important grain and oil crop in the world. With the growth of population and economy, the demand of soybean yield is growing. Soybean leaf diseases have many characteristics, such as large variety, large impact and local outbreaks, which have been the important

factors for restricting the sustainable development of high-yield and high-quality soybean industry [1]. Thus, real-time and precision identification of soybean leaf diseases is urgently needed. With the development of precision agricultural technology, the use of image processing technology to diagnose and control crop diseases has become one of the important research contents of intelligent agriculture [2].

Researchers have made many achievements in the diagnosis and prevention of crop diseases by image recognition technology. For example, Guan et al. used k-nearest neighbor (KNN) to identify weeds [3]. They extracted multi-type features and processed them to identify weeds. The optimal accuracy 88% was obtained in identifying five kinds of weeds. Cheng and Matson proposed a feature-based approach to discriminate weed from rice [4]. They obtained the best precision value of 98.2% and the best recall value of 97.7% by decision tree. Ma et al. extracted 14 dimensional geometry, color and texture features of soybean leaf disease area [5]. In addition, they established two slopes cascade neural network model to identify soybean disease categories and obtained a simulation accuracy of 97.67%. Singh et al. proposed an image segmentation algorithm based on genetic algorithm. Meanwhile, they extracted features from the segmented image by color co-occurrence method [6]. Then, they classified plant diseases and obtained an accuracy of 93.63% by support vector machine (SVM). Pedro et al. proposed a combined method to discriminate monocot and dicot weeds [7]. They obtained the best accuracy of 92.9% by fuzzy multi-criteria decision-making (FMCDM) approach. Deng et al. extracted shape and texture features from wheat *Tilletia* disease images, and then minimum distance method, (back propagation) BP neural network and SVM were used for classifying three types of *Tilletia*

✉ Qiufeng Wu  
qfwu@neau.edu.cn

<sup>1</sup> College of Science, Northeast Agricultural University, No. 600 Changjiang Street, Xiangfang District, Harbin 150030, China

<sup>2</sup> College of Engineering, Northeast Agricultural University, No. 600 Changjiang Street, Xiangfang District, Harbin 150030, China

diseases [8]. The optimal recognition accuracy 82.9% was obtained by SVM.

The methods above for identifying crop diseases are mainly divided into two independent stages, i.e., feature extraction (such as color, texture and shape) and classification (such as KNN [9, 10], SVM and BP neural network [8]). In the feature extraction phase, it is necessary for the agricultural experts to point out the main differences of various diseases and then to extract the corresponding features. On the one hand, feature extraction is subjective. On the other hand, it is difficult to select the optimal feature extractor and classifier. Deep learning can automatically implement the classification process through its end-to-end characteristic, which overcomes the limitations of traditional pattern recognition methods in feature extraction and classification [11]. Recently, deep learning has achieved great success in many fields, such as pavement cracks detection [12], biomedical image application [13], computer vision and signal processing [14], pedestrian detection [15] and face attributes prediction [16], especially in the agriculture field, like pest identification [17], weed detection [18, 19], fruit detection [20], tomato diseases

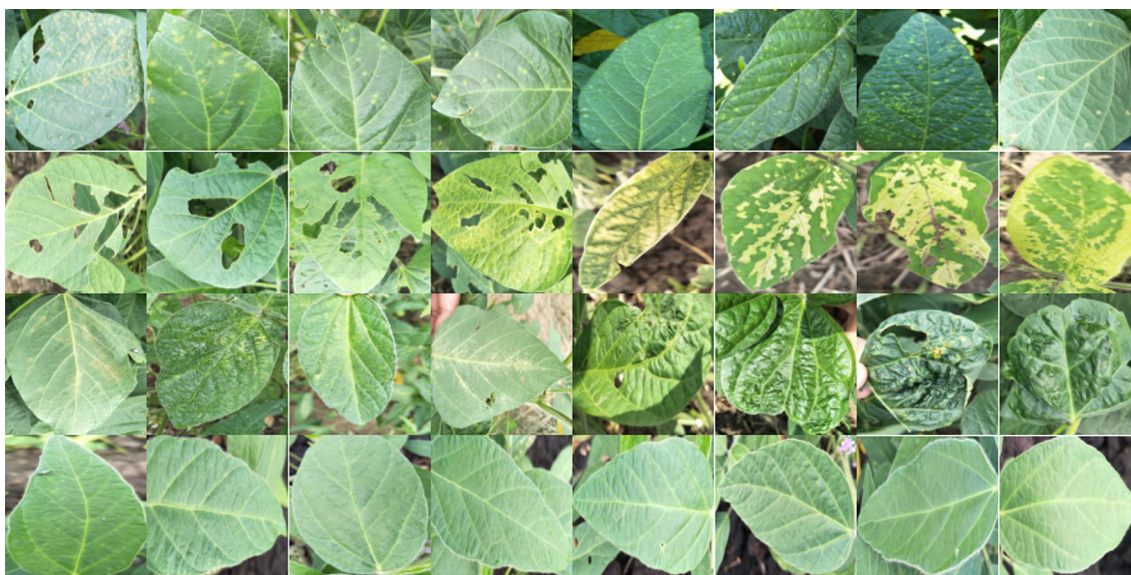
identification [21], cucumber diseases identification [22], crop species and diseases identification [23].

This paper applies deep learning in identifying soybean leaf diseases. AlexNet [24], GoogLeNet [25] and ResNet [26] were selected as backbone of soybean leaf diseases identification model, and transfer learning was used. Furthermore, the effects of adjusting hyperparameters and network structure on the identification of soybean leaf diseases were explored. Finally, the evaluation of the optimal model was carried out.

## Materials and Methods

### The Dataset

From June to August 2017, the soybean leaf diseases dataset was collected from the Xiangyang farm, Nenjiang farm and Jiusan farm of Northeast Agricultural University in Heilongjiang Province. This work collected 1470 raw soybean leaf images (See Fig. 1 and Table 1). There are 200 images of soybean leaves with bacterial disease (See



**Fig. 1** Soybean leaf images

**Table 1** The soybean leaf dataset

Category	Bacterial	Downy mildew	Health	Pest	Pesticide	Spider mite	Virus disease	Total
Original								
Training set	100	155	100	100	80	135	100	770
Test set	100	100	100	100	100	100	100	700
Augmented								
Training set	1000	1550	1000	1000	800	1350	1000	7700

the first to fourth images in Fig. 1 first row), 255 images of soybean leaves with downy mildew (See the fifth to eighth images in Fig. 1 first row), 200 images of health soybean leaves (See Fig. 1 forth row), 200 images of soybean leaves with pest (See the first to fourth images in Fig. 1 second row), 180 images of soybean leaves with pesticide (See the fifth to eighth images in Fig. 1 second row), 235 images of soybean leaves with spider mite (See the first to fourth images in Fig. 1 third row) and 200 images of soybean leaves with virus disease (See the fifth to eighth images in Fig. 1 third row).

Consider the following two problems: (1) The pre-trained networks are deep convolutional neural networks. The network parameters will increase rapidly as the number of network layers increases. (2) The number of the manually collected soybean leaf dataset is small. Massive parameters and small dataset will cause the network to prone to overfitting. The effective method to avoid overfitting is to expand the size of dataset manually. Therefore, data augmentation was conducted.

**Data Augmentation**

In this work, since the collected images are limited, 100 images in each class, totally 700 images, were randomly selected as test set and then data augmentation was conducted in the remaining 770 training images. It should be noted that the augmented 7700 images plus the original 770 training images was used as the final training set subsequently. The number of the final training set is 8470. Ten data augmentation procedures were used: (1) flip images horizontally, vertically and diagonally; (2) adjust brightness, contrast, hue and saturation of images; (3) rotate images by 90°, 180° and 270°.

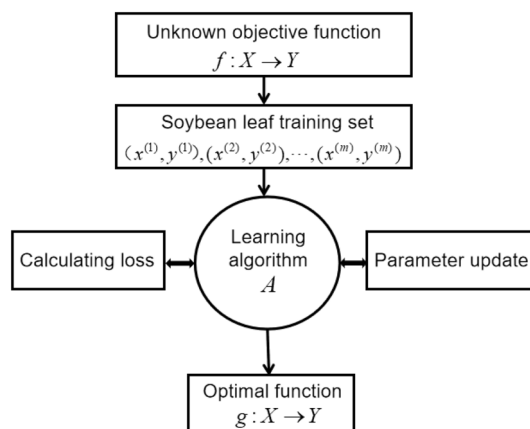
**Model Building**

The main process of soybean leaf diseases identification is shown in Fig. 2. First, the unknown objective function is set to  $f : X \rightarrow Y$ . Then, the soybean leaf training set is sent to the learning algorithm. The optimal objective function  $g : X \rightarrow Y$  will be obtained by calculating the loss and updating parameters by iterations.

The basic process of CNN can be expressed by the following formulas. First, convolution operation is conducted,

$$M_j^l = f \left( \sum_{i \in N_j} M_i^{l-1} \times w_j^l + b_j^l \right) \tag{1}$$

where  $f(\cdot)$  represents an activation function. This work uses the rectified linear unit (ReLU) [27].  $N_j$  denotes the number of filters.  $M_i^{l-1}$  denotes feature map.  $w_j^l$  denotes weight matrix, and  $b_j^l$  denotes bias term.



**Fig. 2** The mathematical model of soybean leaf diseases identification

Max pooling is used in this work. Finally, all features are combined in the fully connected layer. Then, the predicted labels will be obtained by a classification function. Softmax function is used here, and the probability of an input  $x$  classified to a class  $i$  can be calculated by the following equation:

$$p(y = i|x; \theta) = \frac{e^{\theta_i^T x}}{\sum_{j=1}^k e^{\theta_j^T x}} \tag{2}$$

where  $\theta$  denotes the parameters of model and  $k$  represents the total number of categories.

**AlexNet**

AlexNet sparked a wave of research in deep learning after winning ImageNet large-scale visual recognition challenge (ILSVRC) 2012. The AlexNet architecture is directly connected between layers. In addition, dropout layers are added after the fully connected layers to reduce overfitting [24]. The AlexNet used in this work is shown in Fig. 3. The symbols  $a \times a \times b$  denote  $ba \times a$  feature maps obtained after convolutional operation. The number 4096 indicates that there are 4096 neurons in the corresponding fully connected layer. The number 7 in the final fully connected layer represents the number of categories that need to be identified in this work.

**GoogLeNet**

GoogLeNet won the first place in ILSVRC 2014, which is inspired by network-in-network approach [28]. GoogLeNet is an inception architecture with 6.8 million parameters and nine inception modules. Six convolutional layers and one max pooling layer compose each inception module. There are three types of convolutional layers for each module, that is,  $1 \times 1$ ,  $3 \times 3$  and  $5 \times 5$ . The  $1 \times 1$  convolutional

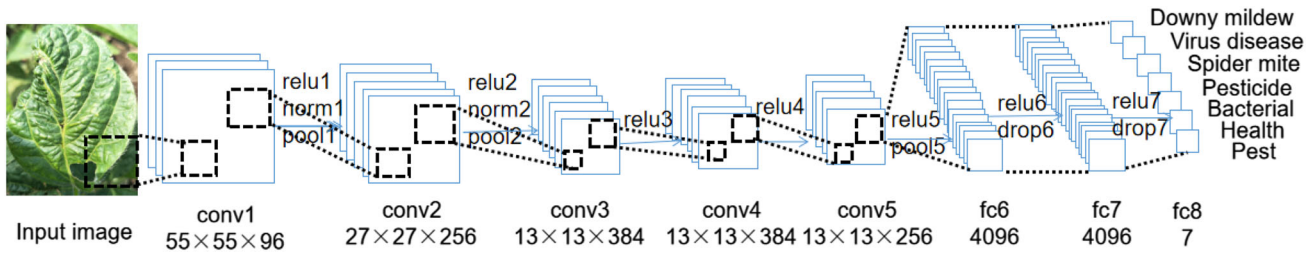
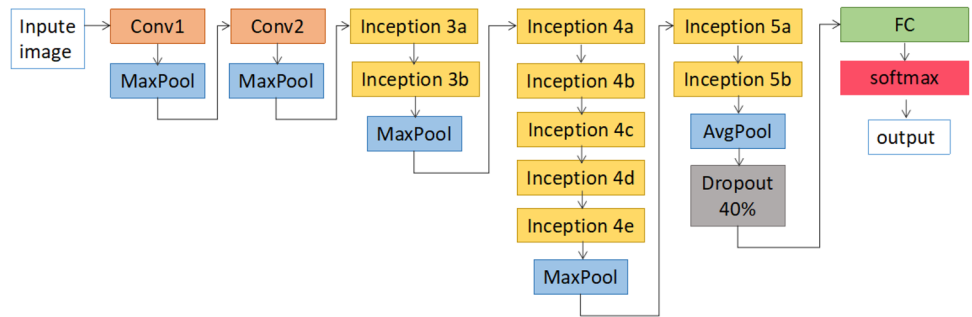


Fig. 3 The architecture of AlexNet

Fig. 4 The architecture of GoogLeNet



layer is designed for reducing the spatial dimension and limiting the scale of GoogLeNet. The amount of neurons in the fully connected layer was revised to 7. Figure 4 shows the GoogLeNet architecture used in this paper.

**ResNet**

ResNet swept first places in ILSVRC and COCO 2015 competitions [26, 29]. ResNet is a very deep network using residual connections. Figure 5 displays a ‘bottleneck’ structure designed for ResNet-50/101/152. The 1 × 1 layers are designed for reducing and then restoring

dimensions. In this work, the ResNet-50 was used, and the ResNet-50 was denoted by ResNet subsequently.

**Results**

The experiment platform utilized in this work is MATLAB. In addition, NVIDIA GTX1050 was used to accelerate the experimental process. In this work, overall accuracy, precision, recall and *F1* score were selected as the quantitative evaluation indices.

**Experiments on Different Pre-trained Models**

The classification accuracy of AlexNet, GoogLeNet and ResNet in various configurations is given in Table 2. In Table 2, the first column displays nine different configurations. The first digit in parentheses in the notation ‘Configuration#’ represents the batch size, and the second digit represents the number of iterations. In addition, the corresponding classification accuracy of AlexNet, GoogLeNet and ResNet is given in column 2, column 3 and column 4, respectively. The optimization method was set to stochastic gradient descent (SGD) with a momentum of 0.9. We used an initial learning rate of 0.001 and then dropped by 0.5 times per 528 iterations.

By comparing the classification accuracy of the three pre-trained networks, it can be found that the performance of ResNet model is relatively optimal. The highest accuracy is obtained by ResNet in configuration2, that is, the

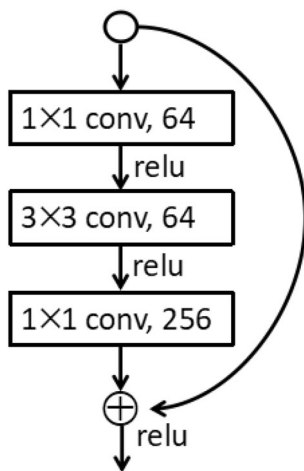


Fig. 5 Bottleneck building block for ResNet



**Table 2** Classification accuracy (%) of different configurations

Configurations	AlexNet	GoogLeNet	ResNet
Configuration1 (16, 528)	76.57	90.14	91.86
Configuration2 (16, 1056)	85.86	91.57	93.71
Configuration3 (16, 2112)	87.29	93.43	93.14
Configuration4 (32, 528)	83.14	88.43	92.57
Configuration5 (32, 1056)	89.29	92.00	93.14
Configuration6 (32, 2112)	89.57	92.43	93.29
Configuration7 (64, 528)	87.14	91.29	92.43
Configuration8 (64, 1056)	89.71	91.86	92.57
Configuration9 (64, 2112)	89.43	92.14	91.86

batch size and the number of iterations are 16 and 1056, respectively. Therefore, the rest works will be conducted based on ResNet.

**Effects of Batch Size**

To clearly discuss the effects of batch size in classification performance, the results of the ResNet in Table 2 were rearranged (see Table 3).

Table 3 shows the rearranged ResNet results from Table 2. There are three types of batch size which are 16, 32 and 64, respectively. In Table 3, the change of batch size has no definite effect on accuracy when the number of iterations is fixed. When the number of iterations is 528, the accuracy increases only when the batch size increases from 16 to 32. However, when the batch size continues to increase, the accuracy decreases. The accuracy decreases as the batch size increases when the number of iterations is 1056. When the number of iterations is 2112, as the batch size increases, the accuracy increases first and then decreases.

In fact, the setting of batch size depends on the dataset and the performance of the computer used. In a certain range, increasing the batch size will increase the accuracy of the model, so the parameters of the network learning are

approximately the same as the training using the entire dataset. However, if the batch size is increased blindly, the demand for computing resources will increase. In addition, the network will more easily converge to the sharp minima, and the generalization ability of the trained model will be weakened [30]. In Table 3, the optimal classification accuracy of ResNet is obtained when the batch size is 16.

**Effects of Number of Iterations**

To clearly discuss the effects of number of iterations on the classification accuracy, the results of the ResNet in Table 2 were rearranged (see Table 4).

Table 4 presents the rearranged ResNet results from Table 2. There are three types of number of iterations which are 528, 1056 and 2112, respectively. When the batch size is 16, the accuracy increases first and then decreases as the number of iterations increases. The accuracy increases as the number of iterations increases when the batch size is 32. When the batch size is 64, the accuracy increases first and then decreases with the number of iterations increases. From Table 4, the optimal accuracy is obtained when the batch size is 16 and the number of iterations is 1056. As discussed above with respect to the effects of batch size on accuracy, there is no definitive

**Table 3** Rearranged ResNet results (fixed number of iterations)

Number of iterations	Batch size	Accuracy (%)
528	16	91.86
	32	92.57
	64	92.43
1056	16	93.71
	32	93.14
	64	92.57
2112	16	93.14
	32	93.29
	64	91.86

**Table 4** Rearranged ResNet results (fixed batch size)

Batch size	Number of iterations	Accuracy (%)
16	528	91.86
	1056	93.71
	2112	93.14
32	528	92.57
	1056	93.14
	2112	93.29
64	528	92.43
	1056	92.57
	2112	91.86

answer to the discussion about effects of number of iterations on accuracy.

In practical classification task, there are no fixed rules for the setting of hyperparameters. On the one hand, it is necessary to consider the computing resources the researcher owned. On the other hand, the characteristics of the used dataset also need to be considered. The above discussion provides researchers with a reference for setting batch size and number of iterations.

### Effects of Network Structure

This section explores the impact of network structure on the identification of soybean leaf diseases in the natural environment. In practice, a deep CNN usually has massive parameters. Full training a deep CNN will consume massive time and computing resources. Furthermore, full training a deep CNN requires extensive dataset. Thus, the effects of changing the ResNet structure on the performance of identifying soybean leaf diseases were explored.

The optimal number of iterations and batch size obtained from the above experiments were used. In MATLAB software, the ResNet owns 177 layers. For convenience, the fully connected layer of ResNet was recorded as 'fc.' We used an initial learning rate of 0.001 and dropped by 0.5 times per 528 iterations. The change in network structure was implemented by freezing the weights of the layers that are not participating in training, that is, the learning rate of the layers that are not participating in the training is set to 0, so the parameters of these layers are not updated during the training. Freezing weights of the pre-trained network not only improves training speed, but also reduces overfitting on this small dataset. The experimental results are shown in Table 5. The training layers of network were numbered in the opposite way, that is, the original last layer of the ResNet was recorded as the first layer. In the symbol 'fc'-N in Table 5, the larger N is, the deeper the depth of the network is.

In Table 5, ResNet ('fc'-140) obtains the optimal accuracy of 94.29%, which is 0.58 percentage points higher

than the accuracy of the full training network ResNet ('fc'-177). Meanwhile, the accuracy of ResNet ('fc'-118) is equal to the accuracy of the full training network ResNet ('fc'-177). In addition, the accuracy of both ResNet ('fc'-98) and ResNet ('fc'-76) is 93.43%, which is only 0.28 percentage points lower than the accuracy of the full training network. The experiments show that when fine-tuning a pre-trained network, it may not be necessary to use all layers of the network. In this work, ResNet ('fc'-140) obtains the optimal accuracy, but if massive data are available, ResNet ('fc'-98) and ResNet ('fc'-76) may obtain an acceptable performance.

### Analysis of the Optimal Network

Precision and recall were used to quantify the performance of the optimal model ResNet ('fc'-140) for each class of soybean leaf diseases identification. The precision is the percentage of the correct portion of the test results. The recall is the percentage of the correct part of the test results to the actual correct part. In addition, the *F1* score was given to evaluate the overall model performance. The calculation formulas are:

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \quad (3)$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \quad (4)$$

$$F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (5)$$

A total of 700 samples from seven categories of the test set were tested, and the results are shown in Table 6. In addition, t-distributed stochastic neighbor embedding (t-SNE) [31] was used to visualize the features extracted from 'fc' in the optimal ResNet ('fc'-140) (as shown in Fig. 6).

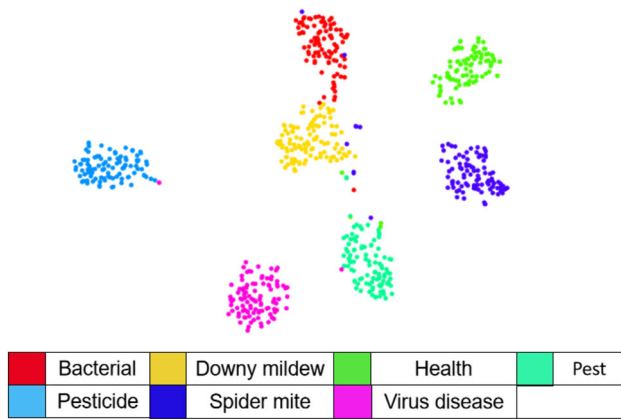
From Table 6 and Fig. 6, the *F1* score of the identification of pesticide reaches 95.78%, which indicates that the optimal model can accurately identify the pesticide of soybean leaves. Meanwhile, the light blue dots in Fig. 6

**Table 5** Classification accuracy of different network structures

Network	Accuracy (%)
ResNet ('fc'-177)	93.71
ResNet ('fc'-140)	94.29
ResNet ('fc'-118)	93.71
ResNet ('fc'-98)	93.43
ResNet ('fc'-76)	93.43
ResNet ('fc'-56)	92.14
ResNet ('fc'-36)	91.00
ResNet ('fc'-14)	88.43

**Table 6** Quantification evaluation of the optimal model recognition results

Category	Precision (%)	Recall (%)	<i>F1</i> score (%)
Bacterial	89.00	92.20	90.57
Downy mildew	98.00	92.36	95.10
Health	91.00	92.47	91.73
Pest	90.00	92.51	91.24
Pesticide	99.00	92.77	95.78
Spider mite	96.00	92.75	94.35
Virus disease	97.00	92.68	94.79



**Fig. 6** t-SNE feature visualization of ‘fc’ in ResNet (‘fc’-140)

representing the pesticide are also completely gathered together. Since the symptoms of bacterial are similar to those of downy mildew and early onset of spider mite, the model has certain errors in identifying bacterial, and the  $F1$  score of the identification of bacterial is 90.57%. The  $F1$  score of the identification of health is 91.73%, which is caused by the fact that the symptom of the initial stage of bacterial, downy mildew and spider mite are minor, and similar to health. In general, the optimal model ResNet (‘fc’-140) can precisely identify the soybean leaf diseases in the natural environment.

## Conclusions

In this paper, the convolutional neural networks were used to identify soybean leaf diseases in the natural environment. AlexNet, GoogLeNet and ResNet were utilized for transfer learning. Firstly, the classification performance based on the three pre-trained models was explored, and then the effects of number of iterations and batch size on network classification performance were discussed. ResNet obtains the highest classification accuracy with 93.71% in the three pre-trained networks. On the basis of ResNet, the effects of network structure on classification performance were explored. The results suggest that the optimal accuracy of 94.29% is obtained when the training depth is 140. Quantify the performance of the optimal model ResNet (‘fc’-140), the highest  $F1$  score is 95.78%, which is obtained from the identification of pesticide. The lowest  $F1$  score is 90.57%, which is obtained from the identification of bacterial. In a word, the optimal network obtained in this work can accurately identify soybean leaf diseases in the natural environment.

**Acknowledgements** This work was supported by the Public Welfare Industry (Agriculture) Research Projects Level-2 under Grant

201503116-04-06; Postdoctoral Foundation of Heilongjiang Province under Grant LBHZ15020; Harbin Applied Technology Research and Development Program under Grant 2017RAQXJ096; and National Key Application Research and Development Program in China under Grant 2018YFD0300105-2.

## References

- J. Meng, Q. Wu, The research of cyclic fluctuation rule and forming reason for soybean production of China. *Ecol. Econ. (Acad. Edit.)* **1**, 175–178 (2008)
- C. Zhao, X. Xue, X. Wang et al., Advance and prospects of precision agriculture technology system. *Trans. Chin. Soc. Agric. Eng.* **19**(4), 7–12 (2003)
- L. Guan, Z. Liu, Q. Wu et al., Multi-type feature fusion technique for weed identification in cotton fields. *Int. J. Signal Process. Image Process. Pattern Recognit.* **9**(2), 355–368 (2016)
- B. Cheng, E.T. Matson, A feature-based machine learning agent for automatic rice and weed discrimination, in *AISC 2015 Conference Proceeding* (2015), pp. 517–527
- X. Ma, H. Guan, G. Qi et al., Diagnosis model of soybean leaf diseases based on improved cascade neural network. *Trans. Chin. Soc. Agric. Mach.* **48**(1), 163–168 (2017)
- S. Vijai, A.K. Misra, Detection of plant leaf diseases using image segmentation and soft computing techniques. *Inform. Process. Agric.* **4**, 41–49 (2017)
- P.J. Herrera, J. Dorado, Á. Ribeiro, A novel approach for weed type classification based on shape descriptors and a fuzzy decision-making method. *Sensors* **14**, 15304–15324 (2014)
- J. Deng, M. Li, Z. Yuan et al., Feature extraction and classification of Tilletia diseases based on image recognition. *Trans. Chin. Soc. Agric. Eng.* **28**(3), 172–176 (2012)
- Z. Liu, X. Xu, Q. Wu, A novel K-nearest neighbor algorithm based on I-divergence criterion. *ICIC Express Lett. B: Appl.* **4**(2), 243–248 (2013)
- X. Xu, Z. Liu, Q. Wu, A novel K-nearest neighbor classification algorithm based on maximum entropy. *Int. J. Adv. Comput. Technol.* **5**(5), 966–973 (2013)
- Y. Lecun, Y. Bengio, G. Hinton, Deep learning. *Nature* **521**, 436–444 (2015)
- L. Zhang, F. Yang, Y.D. Zhang et al., Road crack detection using deep convolutional neural network, in *ICIP 2016 Conference Proceeding* (2016), pp. 3708–3712
- Z. Zhou, J. Shin, L. Zhang et al., Fine-tuning convolutional neural networks for biomedical image analysis: actively and incrementally, in *CVPR 2017 Conference Proceeding* (2017), pp. 4761–4772
- D. Xie, L. Zhang, L. Bai, Deep learning in visual computing and signal processing. *Appl. Comput. Intell. Soft Comput.* **10**, 1–13 (2017)
- W. Ouyang, X. Wang, Joint deep learning for pedestrian detection, in *ICCV2013 Conference Proceeding* (2013), pp. 2056–2063
- Z. Liu, P. Luo, X. Wang et al., Deep learning face attributes in the wild, in *ICCV2015 Conference Proceeding* (2015), pp. 3730–3738
- X. Cheng, Y. Zhang, Y. Chen, Y. Wu, Y. Yue, Pest identification via deep residual learning in complex background. *Comput. Electron. Agric.* **141**, 351–356 (2017)
- S.F. Alessandro, M.F. Daniela, G.S. Gercina, P. Hemerson, T.F. Marcelo, Weed detection in soybean crops using ConvNets. *Comput. Electron. Agric.* **143**, 314–324 (2017)

19. X. Deng, L. Qi, X. Ma et al., Recognition of weeds at seedling stage in paddy fields using multi-feature fusion and deep belief networks. *Trans. Chin. Soc. Agric. Eng.* **34**(14), 165–172 (2018)
20. I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, C. McCool, DeepFruits: a fruit detection system using deep neural networks. *Sensors* **16**, 1222 (2016)
21. K. Zhang, Q. Wu, A. Liu, X. Meng, Can deep learning identify tomato leaf disease? *Adv. Multimed.* (2018). <https://doi.org/10.1155/2018/6710865>
22. J. Ma, K. Du, F. Zheng et al., Disease recognition system for greenhouse cucumbers based on deep convolutional neural network. *Trans. Chin. Soc. Agric. Eng.* **34**(12), 186–192 (2018)
23. S.P. Mohanty, D.P. Hughes, M. Salathé, Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **7**, 1419 (2016)
24. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in *NIPS 2012 Conference Proceeding* (2012), pp. 1097–1105
25. C. Szegedy, W. Liu, Y. Jia et al., Going deeper with convolutions, in *CVPR 2015 Conference Proceeding* (2015). <http://dx.doi.org/10.1109/cvpr.2015.7298594>
26. K. He, X. Zhang, S. Ren et al., Deep residual learning for image recognition, in *CVPR 2016 Conference Proceeding* (2016), pp. 770–778
27. V. Nair, G.E. Hinton, Rectified linear units improve restricted boltzmann machines, in *ICML 2010 Conference Proceeding* (2010), pp. 807–814
28. M. Lin, Q. Chen, S. Yan, *Network in Network* (2014). [arXiv:1312.4400v3](https://arxiv.org/abs/1312.4400v3)
29. K. He, X. Zhang, S. Ren et al., *Identity Mappings in Deep Residual Networks* (2016). [arXiv:1603.05027v3](https://arxiv.org/abs/1603.05027v3)
30. K.N. Shirish, M. Dheevatsa, N. Jorge et al., On large-batch training for deep learning: Generalization gap and sharp minima, in *International Conference on Learning Representations* (2016). [arXiv:1609.04836](https://arxiv.org/abs/1609.04836)
31. L. Maaten, G. Hinton, Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.