IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

MDFC–ResNet: An Agricultural IoT System to Accurately Recognize Crop Diseases

WEI-JIAN HU[1], JIE FAN[1], YONG-XING DU[1], BAO-SHAN LI[1], NEAL N. XIONG[2], ERNST BEKKERING[3]

[1]School of Information Engineering, Inner Mongolia University of Science and Technology, Inner Mongolia, Baotou 014010, China
[2]College of Intelligence and Computing, Tianjin University, Tianjin 300072, China
[3]Department of Mathematics and Computer Science, Northeastern State University, Tahlequah OK 74464, USA

Corresponding author: YONG-XING DU and BAO-SHAN LI (e-mail: dyxql@imust.edu.cn, libaoshan@imust.edu.cn).

This work was supported in part by the Inner Mongolia Autonomous Region's major science and technology: artificial intelligence application technology and product research and development; application research and demonstration in modern pastures, Fund Number 2019ZD025. It was also supported by the Inner Mongolia Natural Science Foundation on livestock big data, livestock grazing trajectory mining, and optimized production decision-making research, Fund Number 2019MS06021. It was supported by Inner Mongolia's special project on the transformation of scientific and technological achievements; Xiaoweiyang's entire industry chain quality traceability big data service platform. It also received support from the Inner Mongolia Autonomous Region Graduate Education Teaching Reform Research and Practice Project; Innovation and Development of Information Discipline Graduate Education Teaching under the Background of Artificial Intelligence Technology, Fund Number YJG20191012710.

ABSTRACT Crop disease diagnosis is an essential step in crop disease treatment and is a hot issue in agricultural research. However, in agricultural production, identifying only coarse-grained diseases of crops is insufficient because treatment methods are different in different grades of even the same disease. Inappropriate treatments are not only ineffective in treating diseases but also affect crop yield and food safety. We combine IoT technology with deep learning to build an IoT system for crop fine-grained disease identification. This system can automatically detect crop diseases and send diagnostic results to farmers. We propose a multidimensional feature compensation residual neural network (MDFC–ResNet) model for fine-grained disease identification in the system. MDFC–ResNet identifies from three dimensions, namely, species, coarse-grained disease, and fine-grained disease and sets up a compensation layer that uses a compensation algorithm to fuse multidimensional recognition results. Experiments show that the MDFC–ResNet neural network has better recognition effect and is more instructive in actual agricultural production activities than other popular deep learning models.

INDEX TERMS IoT; Multiple crops; Fine-grained disease recognition; ResNet; Singular value decomposition.

I. INTRODUCTION

For a long time, crop disease has been one of the most urgent problems in the field of agriculture. It directly affects crop yield, food safety, and sustainable development. South American rubber blight prevents exploitation of native rubber trees, and wheat stem rust strain Ug*99* spread across Africa, Asia, and the Middle East. Historically, the worst pest is the Irish potato blight, which killed *1.2* million people from *1845–1849* [1]. Oerke [2] calculated total global potential loss from *2001–2003* at *26%–29%* for soybean, wheat, and cotton; *31%* for maize; *37%* for rice; *40%* for potatoes. Savary [3] estimated global losses for wheat at *21.5%*, rice at *30%*, maize at *22.5%*, potatoes at *17.2%*, and soybeans at *21.4%*. Generally, crop losses due to pathogens, animals, and weeds are approximately between *20%* and *40%* of global production [2], [4]–[6]. In the near future, global warming may increase crop losses through more active fungi [7] and insects [8].

Rapid and accurate identification of crop diseases is the first step prevention and control. Early identification limits the damage and allows for less intensive countermeasures. If crop diseases are inaccurately recognized, then treatments may be ineffective or even harmful to crops. Globally, and especially in developing nations, methods for identifying crop diseases

are mostly manual. Farmers recognize diseases on the basis of tradition and limited training with potentially high error rates. Similarly, they may not have access to latest information about crop disease treatments. Even when visual inspection is performed by experts in accordance with detailed guidelines and standards, significant interrater variability and low intrarater repeatability are still found [9]–[14]. Therefore, misdiagnosis of crop diseases and inappropriate treatment methods are common and can severely affect agricultural production. Inaccurate chemical treatments can be ineffective, and unnecessarily high dosages increase cost and cause pollution. We must use scientific control technology.

Some scientific detection methods are direct. Samples of plants are analyzed in laboratories with techniques, such as polymerase chain reaction, immune fluorescence, fluorescence in-situ hybridization, enzyme-linked immunosorbent assay, flow cytometry, and gas chromatography–mass spectrometry [15]. Indirect methods include thermography, fluorescence imaging, and hyperspectral techniques [16].

With the development of artificial intelligence technology, deep learning has been used in many fields. In agriculture, deep learning technologies have been widely used in obstacle detection, fruit counting, crop yield estimation, field soil moisture prediction, weather prediction, crop disease identification, and other production activities [17]–[25]. Specifically, crop disease recognition has been a frequent subject of research in recent years [16], [21], [26]–[28]. In crop disease recognition, diseases in fruits, vegetables, and cash crops are mainly used for identification [17].

Analysis of current methods for identifying crop diseases based on deep learning found that the existing methods still have certain limitations. The success of these models is dependent on the quality of the data set [17], [29], color spectrum and vegetation indices [28], different stages of the disease [30], and the limited availability of data sets [29]. The neural networks often have the following limitations:

1.) Existing methods often use a single type of crop (such as tomatoes [21] and cucumbers [31]), and few methods support multiple crops and diseases.

2.) Identification process focuses on recognizing diseases without clear indication of the severity of the disease. In agricultural production, the degree of crop disease is also essential. It directly determines the type and level of treatment. Inaccurate type or dosage not only affects the efficacy but also the safety of consumers [32] and causes unnecessary environmental damage [33], [34].

3.) Existing methods mostly focus on shallow neural network models, such as AlexNet and VGG. Although they have achieved good recognition accuracy for diseases, they need more sophisticated recognition for the disease level.

To solve these limitations and contribute to agricultural production, we have combined deep learning with IoT technology to build an agrarian IoT system for crop disease identification. In the deep learning module of the IoT system, we constructed the multidimensional feature compensation residual neural network (MDFC–ResNet) with feature compensation. Compared with the existing methods of crop disease identification, our model can identify the severity of crop diseases; it is instructive in actual agricultural production activities. Our contributions are as follows:

1.) On the basis of deep learning and IoT technology, we build an end-to-end IoT system for crop disease identification. This system can obtain crop disease information in time and feed it back to farmers.

2.) We improve the model training by using singular value decomposition (SVD) technology to prepare the data. We use SVD to process images of crop leaves, extract relevant information, eliminate noise, compress data size, and reduce image size to a certain extent.

3.) We optimize the network model by adjusting the initialization and optimization procedures of the residual network.

4.) We construct MDFC–ResNet for fine-grained identification of crop diseases.

The remainder of the paper is arranged as follows: Chapter *II* describes the related literature; Chapter *III* introduces our IoT system and the residual network of the multidimensional feature compensation mechanism proposed in this article; Chapter *IV* analyzes the experimental results, and Chapter *V* describes the conclusions and recommendation for future research.

II. RELATED WORK

Early crop disease identification methods are mostly manual, mainly by farmers or related experts to diagnose and identify crop diseases in the field [13], [35]. These methods are very dependent on the farmers or relevant experts' own experience in identifying crop diseases. The problems of these methods are as follows: keen personal subjective awareness, low recognition efficiency, and high recognition error rate.

With the development of image processing technology and its application in the agricultural field, crop disease identification has improved [18], [19], [36]. Computer image processing refers to converting an image signal into a digital signal and then processing it by using a computer. Its advantages are good reproducibility, high processing accuracy, rich processing content, complex nonlinear processing, and flexibility. However, in solving complex problems of crop disease identification, image processing technology seems to be inadequate, and the recognition accuracy rate cannot reach the expected effect.

To solve the problem of crop disease identification in agricultural production activities, a large number of scholars introduced deep learning [20]–[25] into the agricultural field and achieved excellent results. In deep learning technology, photos of crop disease parts are captured, and then a neural network model is sent via the computer. The photos of crop disease parts are sent to the neural network model for learning (feature extraction), and the learned model is finally used to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001237, IEEE Access

identify crop diseases. Deep learning technology is faster, more convenient, and has higher recognition accuracy than the first two methods.

Recent works use the combination of IoT technology and deep learning technology to identify crop diseases [37]–[39]. This method uses IoT technology to combine various information sensing devices with the Internet to collect information in real-time and feed back to the deep learning model. The deep learning model processes the collected information in time and then displays the results in smart terminal devices. In this way, we can grasp relevant information in real-time. An artificial intelligence brain is also available to analyze the information. We combine the IoT technology with deep learning to build an IoT system for crop disease identification in real-time. In the next section, we will focus on the structure of the experimental system.

III. PROPOSED SYSTEM AND MDFC–ResNet SCHEME

A. SYSTEM STRUCTURE

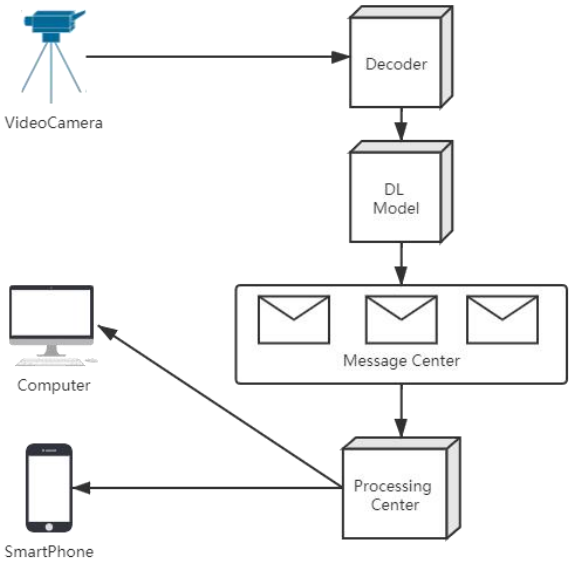

Fig. 1. System structure

The proposed IoT system combines video cameras, deep learning models, and intelligent terminal devices. The system uses cameras to collect crop videos. It feeds back the health status of crops analyzed by deep learning models to farmers through smart terminal applications (web applications or smartphone applications). Fig. *1* shows the system structure. The system consists of the following six parts: one or more video cameras, decoder, deep learning model, message center, processing center, and terminal (computer or smartphone). The main functions of each par and the system operation process are as follows:

The video camera is set up in crop-fields or greenhouse to collect crop information. Usually, we set up multiple video cameras.

The decoder can receive data from multiple video cameras, decode the video data, and extract the crop image from the decoded information stream.

The deep learning model receives the crop image from the decoder, judges the health status of the crop through the trained model, and sends the result to the message center.

The message center receives the discrimination results and organizes and manages the discrimination results in the form of the message queue, which the processor uses.

The processor obtains information from the message center, processes the information, and sends it to the web application and smartphone application in the form of notification.

Deep learning models are an essential part of the system, directly determining the performance of the IoT system. At present, most crop disease identification systems only identify the types of crop diseases but do not perform more fine-grained identification of these diseases (the degree of crop disease identification). However, in actual agricultural production activities, the degrees of crop disease are different. Thus, the treatment plan adopted and the amount of medicine used are also different. Fine-grained identification of crop diseases is instructive in terms of disease treatment, reducing the number of pesticides used, and protecting the crop and the natural environment. Therefore, we propose the MDFC–ResNet model, which can identify the general and serious diseases of crops and is more instructive in actual agricultural production activities, for the system's deep learning model. The MDFC–ResNet model is described in detail in the following section.

B. MODEL FLOW

Fig. *2* shows the overall process of our experiment. It consists of two parts. In the data processing phase, we perform data enhancement, normalization, and SVD operations on pictures in the data set. The purpose is to reduce the negative impact of the data set on model training. In the model training phase, we divide the data set into a training set, a validation set, and a test set. The training set trains the model. The validation set verifies that the expectations are satisfied; if yes, then the model is saved, and if no, then the parameters in the model training are adjusted until the expectations are satisfied. The test set is used to test the accuracy of the model.

C. DATA PROCESSING

The data set used in this study is obtained from AI Challenger [40]. It was divided according to "species–disease–degree," with a total of 59 categories, including 10 species, 49 detailed disease categories, and 10 health categories with a total of 36258 pictures. Each image was obtained from a crop in a natural environment and treated with only one leaf. Fig. 3 shows the various class images.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001237, IEEE Access

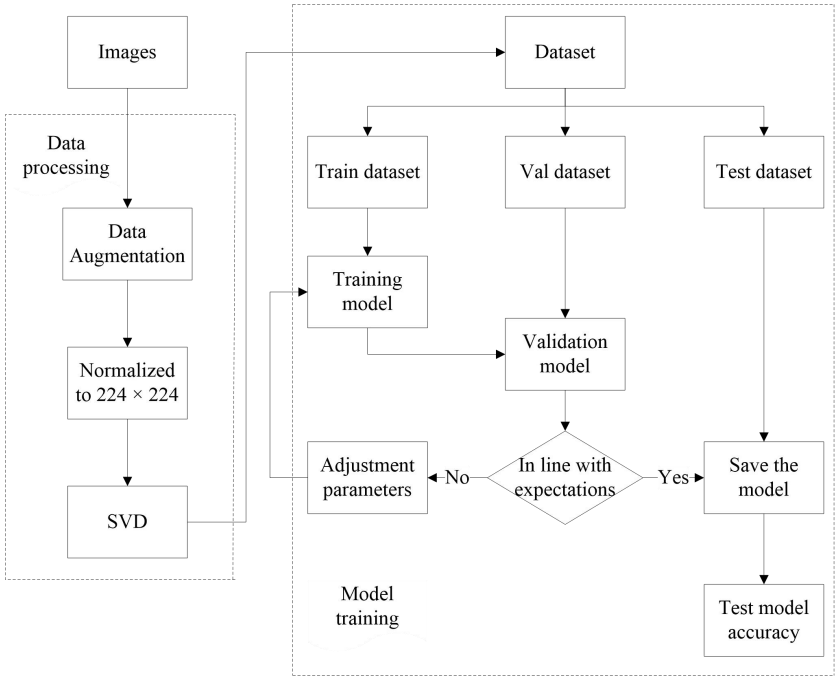

Fig. 2. **Overall experimental process**

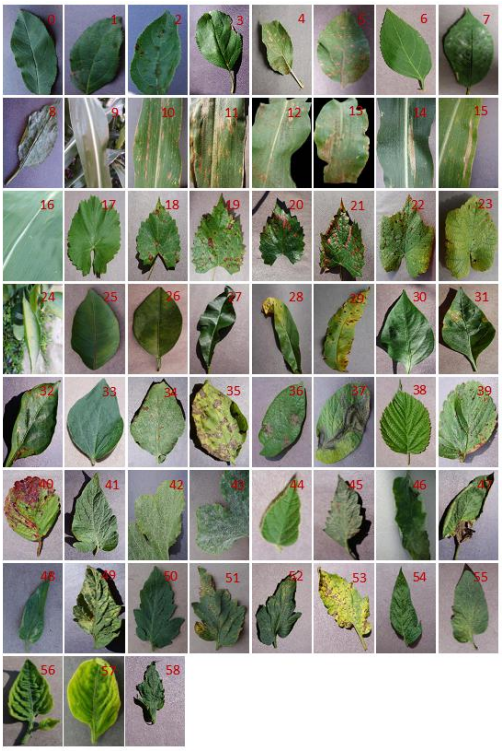

Fig. 3. **Sample dataset diagram**

The dataset was not ready for use in image classification due to different sources of pictures in the original dataset, the shooting environment, equipment, and differences between the crop species. Substantial differences in the number of picture types, uneven picture quality, and inconsistent picture sizes cause image recognition problems. To solve these problems, we processed the dataset before model training. The process included three critical steps, namely, data enhancement, data normalization, and SVD.

The first step of data enhancement solves the problem of significant differences in the number of pictures between the various categories. The largest category has *2473* images, and the smallest category has only *22* images. The number of images in categories affects the model training, resulting in a decrease in test accuracy. Data augmentation technology expands categories with a small amount of samples in the original data set. Random cropping might remove the diseased part of the picture, resulting in loss of the characteristic information. Therefore, we used rotation and horizontal flip to enhance the dataset. We expanded categories with less than *1,000* pictures to approximately *1,000* and reduced larger categories to approximately *1,000*. Balancing the number of images in various categories removes the impact of the number of pictures on the final classification accuracy. Fig. *4* shows an example of data enhancement process. The original image is flipped horizontally and rotated by *90°*, *180°*, and *270°*. At the end of this phase, the dataset increased from *36258* to *63265* images.

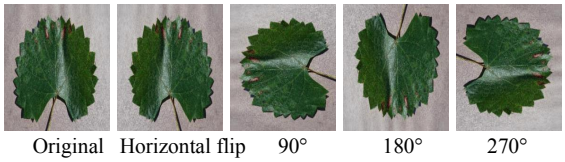

Fig. 4. **Image flip and rotation horizontally**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001237, IEEE Access

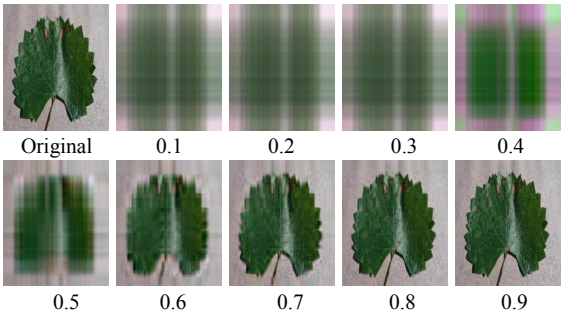

Fig. 5. Original and SVD-processed images

The second step of data normalization converts all images to a uniform size to facilitate model training. We normalized the pictures in the dataset to *224* × *224* pixels prior to the experiment. Many deep learning models use this image size.

The third step of singular value decomposition solves the problem of picture quality. It extracts important information from the original picture and removes noise. In many images, a small section of data carries most of the information, and the remainder is irrelevant. The quality of the pictures in the original data is uneven. We adjust the singular value and observe the effect on the image, an example of which is shown in Fig. *5*. We select the singular value *0.9* to process the images in the data set.

D. MODEL TUNING

During model training, the initializer and optimizer play an essential role and have a significant impact on the final test results.

The residual network is a deep network, and deep networks require appropriate weight initialization to reduce the risk of gradient explosion and gradient disappearance. If the cost gradients are extremely large, then the cost oscillates around the minimum value. If the cost gradient is extremely small, then the cost converges before it can reach the minimum value. Weight initialization refers to the process of setting values in advance before the neural network starts training. Without initialization, the weight values are *0*, and all neural nodes are the same. During backpropagation, each weight gradient is the product of the node's input value x and the gradient of the previous layer. If the weights are equal, then each neural node of the neural network updates equally and no difference between nodes is found. The neural network cannot learn useful information during the training process. An appropriate initializer sets initial weights that allow the network to train in an efficient and timely manner.

The goal of deep learning is to continuously adjust network parameters to allow them to perform various nonlinear transformations on the input to fit the output effectively. Essentially, it is the process of solving the optimal solution of the loss function. Research on deep learning focus on algorithms to update the parameters. We refer to these algorithms as optimizers. Selecting an optimizer in the field of deep learning is one of the top priorities of a model. Even when the dataset and the model architecture are the same, using different optimizers may lead to different training results. Therefore, the combination of network and dataset is tested using different optimizers to select the most effective approach.

With the development of deep learning in recent years, some new optimizers and initializers have been proposed and applied [41]. Therefore, before the experiment, the model is tuned to select the appropriate initializer and optimizer. This experiment is performed in a GPU environment, using Keras framework based on TensorFlow, which mainly adjusts the model's learning rate, epoch, and batch parameters. Table *1* shows the experimental environment and parameter configuration.

TABLE 1
EXPERIMENTAL ENVIRONMENT AND PARAMETER CONFIGURATION

Parameter	Numerical value
Development environment	Using the Keras framework based on Tensorflow.
GPU	CUDA *9.0* and Tensorflow-GPU *9.0*.
Learning rate	The learning rate of the model varies with different optimizers.
Batch	*8*.
Dropout	Dropout is used to prevent model overfitting with a parameter of *0.5*.

We use 58725 pictures in the dataset during the experiment. We use training samples and 4,540 pictures as test samples. The training samples are divided into the training set and the verification set with a ratio of *8:2*, as shown in Table *2*.

TABLE 2
DATA SET PARTITION

Data set	Number of pictures	Effect
Train	35182	Train the model
Val	8795	Adjust the parameters in the model
Test	4540	Test the accuracy of the model

1) INITIALIZER SELECTION.

This experiment selects the optimal initializer for the traditional residual neural network. In the experiment, three common initializers, namely, lecun, glorot, and he [42], [43], [44], [45] are selected as the research objects. Each initializer is used with normal and uniform distribution, resulting in six types, namely, lecun_normal, lecun_uniform, glorot_normal, glorrot_uniform, he_normal, and he_uniform. Technical specifications can be found in the Keras initializer documentation [46]. On the basis of a learning rate of *0.0001*, the optimizer selects *six* sets of comparative experiments. Table *3* shows the experimental results. The glorot_normal initializer proved most suitable for the residual network given

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001237, IEEE Access

the highest scores for training accuracy, validation accuracy, and test accuracy. Therefore, we selected the glorot_normal initializer as the model's initializer for the remaining experiments.

TABLE 3
EXPERIMENTAL RESULTS OF EACH INITIALIZER

Initializer	Training accuracy (%)	Validation accuracy (%)	Test accuracy (%)
lecun_uniform	88.61	85.73	83.92
lecun_normal	85.53	83.77	78.83
he_uniform	85.85	83.80	77.73
he_normal	86.31	84.06	78.59
glorot_uniform	88.58	85.85	83.72
glorot_normal	**90.92**	**87.31**	**84.07**

2) OPTIMIZER SELECTION.

TABLE 4
LEARNING RATES OF FIVE OPTIMIZERS AND EXPERIMENTAL RESULTS

Optimizer	Learning Rates	Training accuracy (%)	Validation accuracy (%)	Test accuracy (%)
SGD	0.0010	90.92	87.31	84.07
RMSProp	0.0001	94.65	94.36	82.75
Adadelta	0.1000	93.16	89.33	82.44
Adamax	0.0001	93.77	89.03	80.72
Adam	**0.0001**	**96.44**	**90.72**	**84.16**

This experiment aims to select the best optimization algorithm for the traditional residual neural network. We compared five optimizers by using the initializer, glorot_normalas. The five optimizers are SGD [47], RMSProp [48], Adadelta [49], Adam [50], and Adamax [51]. Technical specifications can be found in the Keras optimizer documentation [48]. The results of the comparative tests are shown in Table *4*. The experimental results show that the Adam optimizer scores best in training accuracy and test accuracy and second best in validation accuracy. Therefore, we select Adam optimizer for this experiment.

E. MDFC–ResNet

Our MDFC–ResNet is based on the deep residual network optimized in the previous section. The MDFC–ResNet model consists of three dimensions, namely, species, disease, and disease level. A compensation layer connects the three levels. The recognition results of species and disease dimensions are fed back as a compensation and error correction mechanism and to improve the accuracy of crop disease level recognition. Fig. *6* shows the structure of our deep residual neural network with the multidimensional compensation mechanism. The uppermost layer shows the first species dimension. The optimized ResNet-*34* network is used to identify the species to which the picture belongs. The second dimension is the disease dimension in the middle layer of Fig. *6*. It uses the optimized ResNet-*50* network to identify diseases. The third dimension is the disease level dimension in the bottom layer of Fig. *6*. The ResNet-*50* models of disease dimension and disease level dimension use parameter sharing to speed up the recognition process and improve recognition accuracy.

Algorithm *1* describes the process of MDFC–ResNet. For any single crop disease picture, the output of the three dimensions is the probability distribution matrix of the crop species to which the picture belongs, the disease, and the level of disease. A compensation layer is used after three dimensions. The compensation layer receives the probability distribution matrices from the three dimensions. It uses the probability distribution matrix of the identified species and t of the disease as feedback data to compensate for the probability of the obtained level of disease.

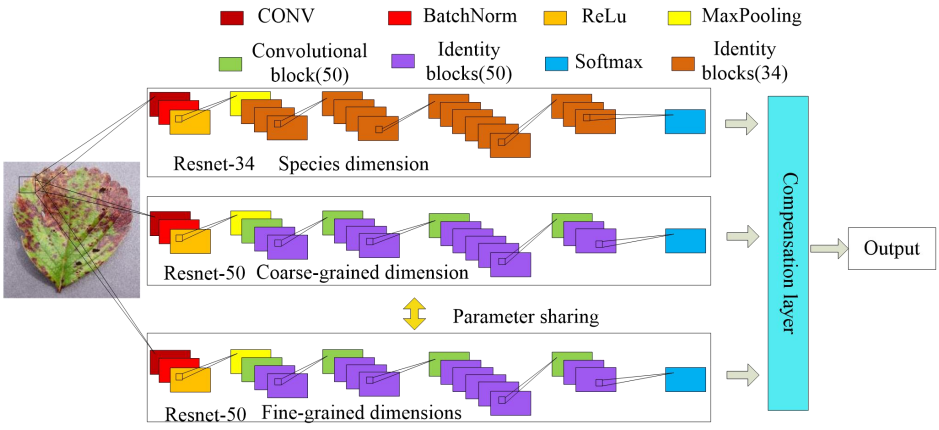

Fig. 6. MDFC–ResNet

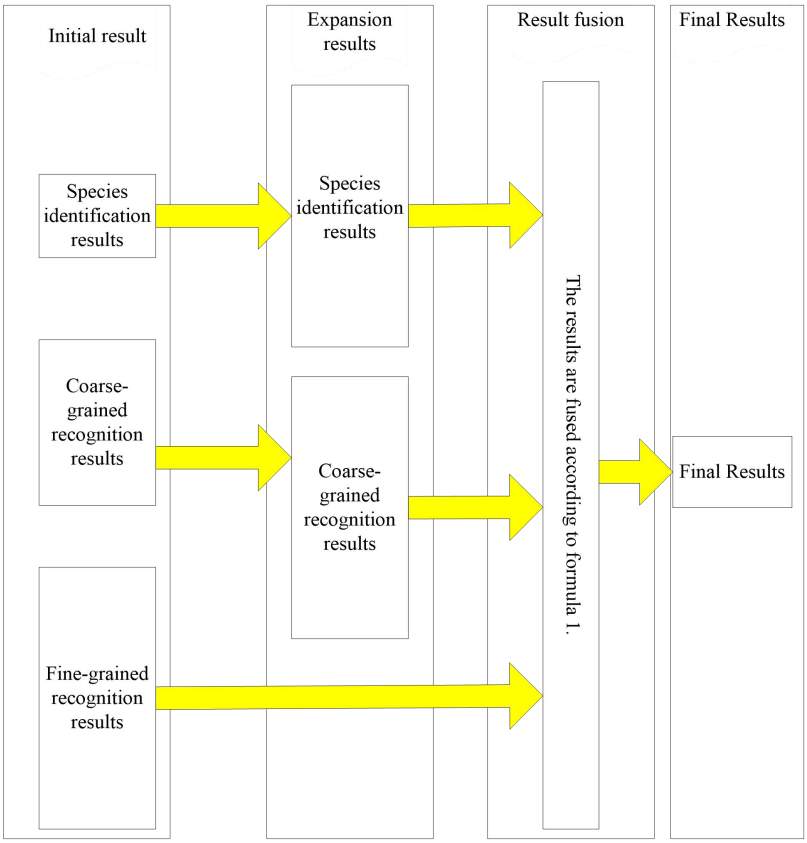

Fig. 7. Operation process of compensation layer

Algorithm *1* introduces the specific design scheme of MDFC–ResNet.

Algorithm 1: MDFC–ResNet

Input: Training dataset and validation dataset
Output: Picture recognition results

For image in train dataset
Picture feed into the first (species) dimension
ResNet-34
Result of species identification
Picture feed into the second (disease) dimension
ResNet-50
Obtain disease results
Picture feed into the third (disease level) dimension
ResNet-50
Get disease level results
Use compensation layer to integrate species results, disease results, and disease level results
End For
Output Final results
End Algorithm 1

Fig. *7* shows the operation flow of the distribution matrix and compensation layer.

The results of the species identification, disease identification, and disease level identification are in different dimensions. Thus, the species dimension and the disease dimension results are "expanded" in accordance with the disease level result before all results are fused by using Eq. *1*, as follows:

$$\begin{pmatrix} P_{Z_0}' \\ P_{Z_1}' \\ P_{Z_2}' \\ P_{Z_3}' \\ \cdot \\ \cdot \\ \cdot \\ P_{Z_{59}}' \end{pmatrix} = \alpha \begin{pmatrix} P_{X_0} \\ P_{X_0} \\ P_{X_1} \\ P_{X_1} \\ \cdot \\ \cdot \\ \cdot \\ P_{X_9} \end{pmatrix} + \beta \begin{pmatrix} P_{Y_0} \\ P_{Y_0} \\ P_{Y_1} \\ P_{Y_1} \\ \cdot \\ \cdot \\ \cdot \\ P_{Y_{35}} \end{pmatrix} + \begin{pmatrix} P_{Z_0} \\ P_{Z_1} \\ P_{Z_2} \\ P_{Z_3} \\ \cdot \\ \cdot \\ \cdot \\ P_{Z_{59}} \end{pmatrix} \quad (Y_j \in X_i \ \& \ Z \in Y_j). \tag{1}$$

In the model, P_{X_i} is the probability of the *i*-th species, P_{Y_j} is the probability of the *i*-th disease, P_{Z_i} is the probability of the *i*-th disease level, and P_{Z_i}' is the final detailed feature recognition result. In the process of selecting α and β values, we set $\alpha = 1$, determine the value of β, and finally, determine the best value of α. After several trials, we achieve the highest accuracy of the test set when $\alpha = 10$ and $\beta = 1.5$.

Algorithm *2* shows the calculation algorithm of the compensation layer.

Algorithm 2: Compensation layer design

Input: Species result matrix, disease result matrix, and disease level result matrix
Output: Picture recognition results

For disease level result *i* in disease level result matrix

2: Find the species *s* to which *i* belongs to
3: From species result matrix, find species *s* result
4: Set position *i* of the extended species result matrix as species *s* result
5: Find the disease *c* to which *i* belongs
6: From disease result matrix, find disease level *c* result
7: Set position *i* of the extended disease result matrix as disease *c* result
8: **End** For
9: Combine the extended species result matrix, the extended disease result matrix, and the disease level result matrix by using Formula 1 to calculate the detailed result matrix of compensation
10: Find the maximum score from the detailed result matrix of compensation
11: **Output** The category corresponding to the maximum score
12: **End Algorithm 2**

The specific process of "expanding" is shown in Fig. *8*, with apples and tomatoes as examples. The recognition result of the species dimension is a matrix of *2×1*, which represents the probability of apple and tomato. The recognition result of the disease dimension is a *4×1* matrix. The matrix contains apple scab, apple gray spot, tomato powdery mildew, and tomato scab. The disease level dimension result is an *8×1* matrix, which generally contains apple scab, severe apple scab, general apple gray spot, severe apple gray spot, general tomato powdery mildew, severe tomato powdery mildew, general tomato scab, and severe tomato scab. At this point, the dimensions of the matrices are different. In the expansion process, the size of the disease level dimension is used as a reference, and the results of the species dimension and the disease dimension are expanded into *8×1* matrix. The species dimension extends from *2×1* to *8×1*; thus, each element repeats four times before moving to the next element. Similarly, because the disease dimension expands from *4×1* to *8×1*, each element of the disease dimension repeats twice before moving to the next element. At the end of this expansion process, the three matrices have identical sizes.

Ⅳ. Performance analysis

We preprocess the data set in the model tuning process by normalizing the size of the pictures. We use SVD technology to remove the noise in the picture and determine

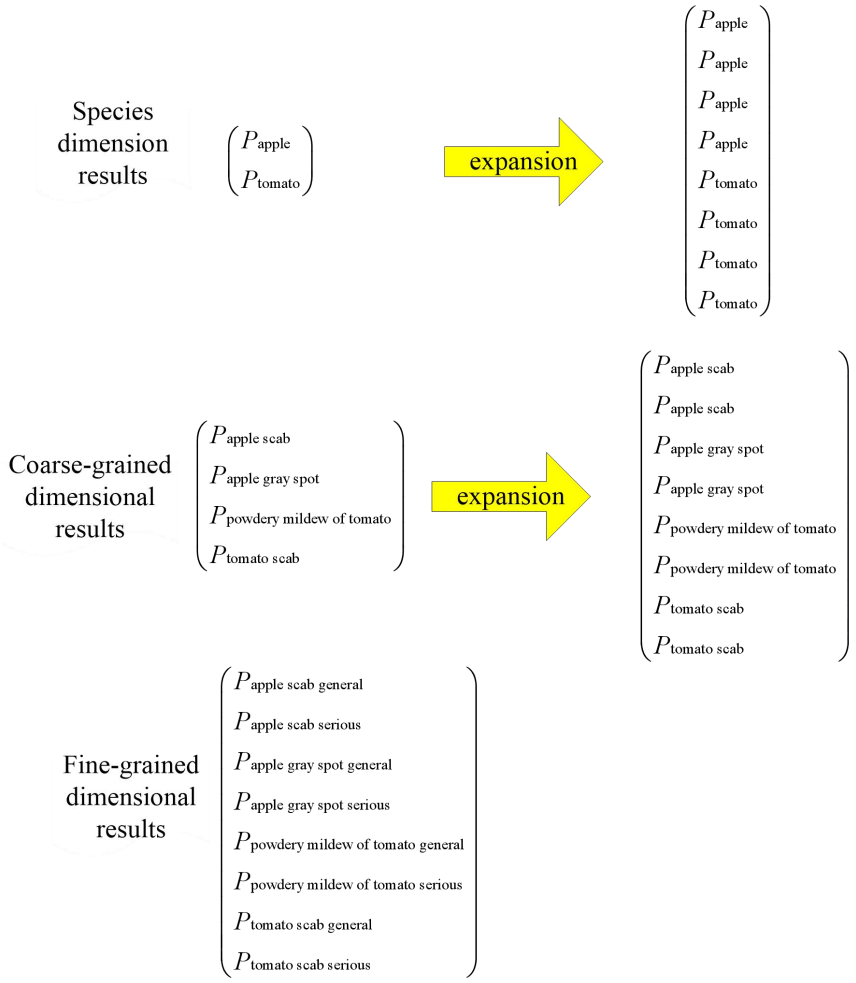

Fig. 8. Apple and tomato as examples to illustrate the operation process of the compensation layer in detail

the model's initializer and optimizer. This section focuses on the performance of our proposed model.

In the experiment, we compare MDFC–ResNet with commonly used methods in crop disease identification, namely, AlexNet, VGG, and ResNet-50. The results of the experiment confirm that our proposed residual network with multidimensional feature comparison performs better than traditional models. The overall results are shown in Table *5*. The table shows that our MDFC–ResNet has the best performance in terms of training accuracy, validation accuracy, and test accuracy. Specifically, the accuracy of the training set is increased by *1.10%*, *8.40%*, and *5.31%*; the accuracy of the validation set is increased by *0.08%*, *4.12%,* and *3.04%*; the accuracy of the test set is improved by *5.03%*, *2.12%,* and *3.18%*. Our model contains two ResNet-*50* models and one ResNet-*34* model (arguably less accurate than ResNet-*50*); thus, we conclude that the compensation layer adds accuracy to our model is reasonable.

TABLE 5
EXPERIMENTAL RESULTS OF EACH MODEL

Model	VGG-19	AlexNet	ResNet-50	MDFC–ResNet
Training accuracy (%)	92.86	85.52	88.65	**93.96**
Validation accuracy (%)	89.74	85.70	86.79	**89.82**
Test accuracy (%)	80.19	83.10	82.04	**85.22**

Network performance is not only measured by high accuracy. Accuracy is simply the ratio of accurately predicted values to the total number of observations. In any recognition model, we have true positive and true negative values, as well as false positive and false negative values. Precision is the ratio of true positive over true positive and true negative. In our study, it represents the accurately identified diseased leaves out of the total diseased leaves. Recall or sensitivity is the ratio of true positive, overall labeled as positive, whether correct or not. In our study, it represents the accurately identified diseased leaves out of all leaves labeled as diseased. Finally, F1 ratio is the weighted average of precision and recall. This measure is more useful than accuracy if the class distribution is uneven. Our model holds 59 crop disease categories, but 49 of which are levels of specific diseases; only 10 categories show healthy leaves. This finding skew the disease identification, but the F1 statistic corrects the disparity. For a more detailed analysis, we review the precision, recall, and F1 values of all four models in the 59 classifications. Table 6 shows the precision, recall rate, and F1 measure of each model.

We compare the performance of MDFC–ResNet with the best performance of the other three models, in terms of ranges of precision, recall, and F1. For each network and each statistic, we record the average, minimum, and maximum scores. We then determine the highest average for the three other networks and compare with the average of MDFC–ResNet. The same procedure is performed for the minimum and maximum scores. All networks had perfect 100% scores; thus, we considered them as well. Table 7 shows the results. MDFC–ResNet performs better than the other models on all measures. It has the highest average accuracy, the highest range (minimum to maximum), and perfect scores for precision, recall, and F1 values. The only exception is the minimum score for F1 percentage, and class 22 obtain the worst case; the score of the two other models is only 44.45%. Finally, MDFC–ResNet scores lower than 50% for F1 in only two of 59 cases, namely, 22 and 50.

TABLE 6.
RESULTS FOR EACH CLASS IN MDFC-RESNET

Class	VGG-19			AlexNet			ResNet-50			MDFC-ResNet		
	precision (%)	Recall (%)	F1(%)	precision (%)	Recall (%)	F1(%)	precision (%)	Recall (%)	F1(%)	precision (%)	Recall (%)	F1(%)
0	72.78	90.44	80.65	88.17	100.00	93.71	97.63	93.75	95.65	**98.22**	**95.40**	**96.79**
1	60.00	56.25	58.06	56.67	53.13	54.84	70.00	72.41	71.18	**63.33**	**70.37**	**66.66**
2	63.64	58.33	60.87	59.09	54.17	56.52	72.73	76.19	74.42	**68.18**	**78.95**	**73.17**
3	96.72	90.77	93.65	100.00	96.83	98.39	100.00	91.04	95.31	**100.00**	**96.83**	**98.39**
4	90.00	90.00	90.00	90.00	85.71	87.80	100.00	83.33	90.91	**95.00**	**86.36**	**90.47**
5	83.33	83.33	83.33	83.33	71.43	76.92	83.33	100.00	90.91	**66.67**	**66.67**	**66.67**
6	75.29	100.00	85.90	100.00	92.39	96.04	98.82	96.55	97.67	**100.00**	**98.84**	**99.42**
7	100.00	85.71	92.31	91.67	73.33	81.48	100.00	92.31	96.00	**83.33**	**100.00**	**90.91**
8	88.89	88.89	88.89	83.33	93.75	88.23	100.00	85.71	92.31	**100.00**	**100.00**	**100.00**
9	96.30	100.00	98.12	94.44	98.08	96.23	98.15	89.83	93.81	**100.00**	**100.00**	**100.00**
10	70.37	55.88	62.29	70.37	70.37	70.37	92.59	59.52	72.46	**85.19**	**57.50**	**68.66**
11	66.67	40.00	50.00	70.83	60.71	65.38	54.17	65.00	59.09	**58.33**	**66.67**	**62.22**
12	62.32	91.49	74.14	73.91	83.61	78.46	75.36	85.25	80.00	**68.12**	**85.45**	**75.81**
13	86.27	63.77	73.33	82.35	70.00	75.67	82.35	71.19	76.36	**84.31**	**66.15**	**74.13**
14	48.28	77.78	59.58	72.41	65.63	68.85	79.31	69.70	74.20	**79.31**	**63.89**	**70.77**
15	74.65	82.81	78.52	78.87	83.58	81.16	77.46	96.49	85.93	**71.83**	**94.44**	**81.60**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3001237, IEEE Access

16	100.00	99.15	99.57	100.00	100.00	100.00	93.10	10.00	18.06	**100.00**	**100.00**	**100.00**
17	97.62	97.62	97.62	100.00	100.00	100.00	100.00	95.45	97.67	**100.00**	**97.67**	**98.82**
18	62.96	65.38	64.15	53.70	72.50	61.70	77.78	67.74	72.41	**72.22**	**78.00**	**75.00**
19	78.79	73.24	75.91	87.88	69.88	77.85	75.76	80.65	78.13	**83.33**	**79.71**	**81.48**
20	77.03	80.28	78.62	78.38	85.29	81.69	89.19	82.50	85.71	**86.49**	**84.21**	**85.33**
21	89.83	76.81	82.81	89.83	80.30	84.80	77.97	86.79	82.14	**81.36**	**81.36**	**81.36**
22	22.22	66.67	33.33	11.11	25.00	15.38	22.22	66.67	33.33	**22.22**	**40.00**	**28.57**
23	97.78	90.72	94.12	96.67	89.90	93.16	98.89	92.71	95.70	**96.67**	**92.55**	**94.57**
24	100.00	98.11	99.05	98.08	96.23	97.15	100.00	94.55	97.20	**100.00**	**100.00**	**100.00**
25	73.23	72.69	72.96	79.55	70.39	74.69	72.86	73.41	73.13	**66.17**	**73.86**	**69.80**
26	67.18	74.89	70.83	64.89	76.58	70.25	71.76	71.48	71.62	**75.57**	**68.04**	**71.61**
27	88.89	86.49	87.67	91.67	78.57	84.62	91.67	89.19	90.41	**86.11**	**93.94**	**89.85**
28	72.13	91.67	80.73	80.33	90.74	85.22	79.51	94.17	86.22	**83.61**	**94.44**	**88.70**
29	90.00	87.61	88.79	90.91	84.03	87.33	94.55	85.25	89.66	**90.00**	**83.90**	**86.84**
30	78.23	92.74	84.87	95.92	97.24	96.58	91.84	97.83	94.74	**98.64**	**98.64**	**98.64**
31	67.50	81.81	73.97	77.50	83.78	80.52	75.00	78.95	76.92	**85.00**	**79.07**	**81.93**
32	87.04	74.60	80.34	90.74	84.48	87.50	81.48	84.62	83.02	**81.48**	**83.02**	**82.24**
33	94.12	86.49	90.14	98.04	99.01	98.52	95.10	97.98	96.52	**98.53**	**98.53**	**98.53**
34	68.97	76.92	72.73	79.31	74.19	76.66	82.76	77.42	80.00	**89.66**	**72.22**	**80.00**
35	91.78	80.72	85.90	89.04	84.42	86.67	91.78	89.33	90.54	**87.67**	**90.14**	**88.89**
36	80.56	63.04	70.73	83.33	63.83	72.29	88.89	65.31	75.30	**88.89**	**80.00**	**84.21**
37	79.69	68.92	73.91	82.81	76.81	79.70	75.00	73.85	74.42	**84.38**	**81.82**	**83.08**
38	100.00	53.85	70.00	97.14	94.44	95.77	100.00	85.37	92.11	**100.00**	**92.11**	**95.89**
39	66.67	90.00	76.60	62.96	85.00	72.34	74.07	90.91	81.63	**85.19**	**79.31**	**82.14**
40	98.80	86.32	92.14	96.39	86.96	91.43	96.39	90.91	93.57	**91.57**	**93.83**	**92.69**
41	96.53	71.37	82.06	95.95	95.40	95.67	97.11	94.38	95.73	**98.27**	**92.39**	**95.24**
42	71.74	73.33	72.53	76.09	97.22	85.37	78.26	62.07	69.23	**71.74**	**68.75**	**70.21**
43	90.58	87.41	88.97	86.23	90.84	88.47	84.06	92.06	87.88	**89.13**	**90.44**	**89.78**
44	63.89	56.10	59.74	75.00	72.97	73.97	75.00	84.38	79.41	**80.56**	**90.63**	**85.30**
45	85.71	73.97	79.41	79.37	69.44	74.07	84.13	75.71	79.70	**85.71**	**90.00**	**87.80**
46	55.26	80.77	65.62	63.16	70.59	66.67	65.79	67.57	66.67	**71.05**	**75.00**	**72.97**
47	77.22	91.73	83.85	74.68	93.65	83.10	72.78	92.00	81.27	**87.34**	**92.00**	**89.61**
48	71.74	73.33	72.53	73.91	72.34	73.12	73.91	73.91	73.91	**78.26**	**83.72**	**80.90**
49	72.92	71.43	72.17	77.08	75.51	76.29	81.25	68.42	74.29	**87.50**	**82.35**	**84.85**
50	25.00	50.00	33.33	50.00	33.33	40.00	50.00	40.00	44.44	**50.00**	**40.00**	**44.44**
51	60.00	60.00	60.00	20.00	100.0	33.33	60.00	42.86	50.00	**60.00**	**42.86**	**50.00**
52	81.67	74.24	77.78	76.67	79.31	77.97	76.67	80.70	78.63	**81.67**	**77.78**	**79.68**
53	81.74	85.45	83.55	86.09	86.84	86.46	89.57	88.79	89.18	**89.57**	**90.35**	**89.96**
54	64.94	64.94	64.94	77.92	70.59	74.07	83.12	73.56	78.05	**77.92**	**78.95**	**78.43**
55	41.03	76.19	53.34	69.23	65.85	67.50	61.54	68.57	64.87	**66.67**	**65.00**	**65.82**
56	79.70	67.93	73.35	85.15	65.65	74.14	87.62	63.44	73.59	**85.15**	**67.98**	**75.60**
57	76.49	89.11	82.32	72.80	89.86	80.44	69.41	93.16	79.55	**75.35**	**90.78**	**82.35**
58	91.89	69.39	79.07	91.89	91.89	91.89	91.89	94.44	93.15	**91.89**	**94.44**	**93.15**

TABLE 7
SUMMARY OF ACCURACY

	VGG-19			AlexNet			ResNet-50			MDFC-ResNet		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
average	77.20	77.78	76.47	79.71	80.06	79.06	82.40	79.79	79.93	82.79	82.33	82.24
									others	82.40	80.06	79.93
										highest	highest	highest
min	22.22	40	33.33	11.11	25	15.38	22.22	10	18.06	22.22	40	28.57
									others	22.22	40	33.33
										highest	highest	not
max	100	100	99.57	100	100	100	100	100	97.67	100	100	100
									others	100	100	100
										highest	highest	highest
perfect	4	2	0	4	4	2	7	1	0	8	5	4
									others	7	4	2
										highest	highest	highest

V. CONCLUSION AND FUTURE RESEARCH

We have shown that our IoT system is effective in crop disease recognition systems of the agricultural industry. Through the combination of deep learning and IoT technology, the proposed method can be automatically used with multiple crop types. It differentiates between levels of disease, apart from recognizing the disease. Treatment protocols may differ between mild and severe cases of crop disease, especially in types and amounts of chemicals used to combat the disease. Most importantly, accuracy is high, even compared with human recognition. We attribute much of this accuracy to using three stages of recognition and the compensation layer.

Future research can focus on two aspects; on the one hand, identifying reasons why recognition fails in some cases. Improving accuracy at the low end enhances the efficiency of the system. Picture quality may be a factor in some cases, and we can establish guidelines and requirements for pictures in the dataset. One possible solution may be the use of color calibration charts to be included in the images. Preprocessing may then include correcting the image colors to standard. Visual attention mechanisms and target detection can then improve picture data. On the other hand, we improve the IoT system by increasing sensor types. Through sensors, we can collect weather, soil, and air quality data to improve the accuracy of crop disease recognition.

ResNet is suitable for multiple purposes in agriculture. Farmers can use them for fruit counting, crop yield estimation, field soil moisture prediction, weather prediction, crop disease identification, and other production activities. Surprisingly, little is known about the actual level of use. We may learn from the failure of medical diagnostic systems in healthcare. One reason is the difficulty of data entry. This case is not the same as in agriculture. Farmers can take pictures with their cell phones and send them for analysis. Another factor may be more relevant. Medical diagnostic systems suffer from lack of follow-up in the form of recommendations for treatment. Future research could focus on the impact of providing treatment recommendations according to best practices. Combined with the excellent performance of IoT technology and neural convolutional networks, these steps may improve penetration in practice.

REFERENCES

[1]. D. P. Hughes and M. Salathé, “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” p. 13, 2015.

[2]. E.-C. Oerke, “Crop losses to pests,” J. Agric. Sci., vol. 144, no. 1, pp. 31–43, Feb. 2006, doi: 10.1017/S0021859605005708.

[3]. S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, “The global burden of pathogens and pests on major food crops,” Nat. Ecol. Evol., vol. 3, no. 3, pp. 430–439, Mar. 2019, doi: 10.1038/s41559-018-0793-y.

[4]. P. S. Teng and S. V. Krupa, “Assessment of losses which constrain production and crop improvement in agriculture and forestry. Proceedings of the EC Stakman Commemorative Symposium. University of Minnesota, Minneapolis, Minnesota.,” in Assessment of losses which constrain production and crop improvement in agriculture and forestry. Proceedings of the EC Stakman Commemorative Symposium. University of Minnesota, Minneapolis, Minnesota., 1980.

[5]. P. S. Teng, Crop Loss Assessment and Pest Management. St. Paul: APS Press.

[6]. S. Lee, Y. Jeong, S. Son, B. Lee,. "A self-predictable crop yield platform (SCYP) based on crop diseases using deep learning." Sustainability 11.13 (2019): 3637.

[7]. Á. Medina, J. M. González-Jartín, and M. J. Sainz, “Impact of global warming on mycotoxins,” Curr. Opin. Food Sci., vol. 18, pp. 76–81, Dec. 2017, doi: 10.1016/j.cofs.2017.11.009.

[8]. C. A. Deutsch et al., “Increase in crop losses to insect pests in a warming climate,” Science, vol. 361, no. 6405, pp. 916–919, Aug. 2018, doi: 10.1126/science. aat3466.

[9]. J. Boulent, S. Foucher, J. Théau, P. L. St-Charles, "Convolutional Neural Networks for the Automatic Identification of Plant Diseases: a Review." Frontiers in Plant Science 10 (2019): 941.

[10]. S. Verma, A. Chug, A. P. Singh. "Exploring capsule networks for disease classification in plants." Journal of Statistics and Management Systems 23.2 (2020): 307-315.

[11]. T. Fan, J. Xu. "Image Classification of Crop Diseases and Pests Based on Deep Learning and Fuzzy System." International Journal of Data Warehousing and Mining (IJDWM) 16.2 (2020): 34-47.

[12]. M. Loey, A. ElSawy, M. Afify. "Deep Learning in Plant Diseases Detection for Agricultural Crops: A Survey." International Journal of Service Science, Management, Engineering, and Technology (IJSSMET) 11.2 (2020): 41-58.

[13]. S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, D. Traore. "Deep neural networks with transfer learning in millet crop images." Computers in Industry 108 (2019): 115-120.

[14]. S. H. Lee, H. Goëau, P. Bonnet, A. Joly. "New perspectives on plant disease characterization based on deep learning." Computers and Electronics in Agriculture 170 (2020): 105220.

[15]. Y. Fang and R. P. Ramasamy, “Current and Prospective Methods for Plant Disease Detection,” Biosensors, vol. 5, no. 3, pp. 537–561, Aug. 2015, doi: 10.3390/bios5030537.

[16]. A.-K. Mahlein, E.-C. Oerke, U. Steiner, and H.-W. Dehne, “Recent advances in sensing plant diseases for precision crop protection,” Eur. J. Plant Pathol., vol. 133, no. 1, pp. 197–209, May 2012, doi: 10.1007/s10658-011-9878-z.

[17]. A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” Comput. Electron. Agric., vol. 147, pp. 70–90, 2018.

[18]. P. Pandey, Y. Ge, V. Stoerger, and J. C. Schnable, “High throughput in vivo analysis of plant leaf chemical properties using hyperspectral imaging,” Front. Plant Sci., vol. 8, p. 1348, 2017.

[19]. M. Nouri et al., “Near infrared hyperspectral dataset of healthy and infected apple tree leaves images for the early detection of apple scab disease,” Data Brief, vol. 16, pp. 967–971, 2018.

[20]. J. Ma, K. Du, F. Zheng, L. Zhang, Z. Gong, and Z. Sun, “A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network,” Comput. Electron. Agric., vol. 154, pp. 18–24, Nov. 2018, doi: 10.1016/j.compag.2018.08.048.

[21]. A. K. Rangarajan, R. Purushothaman, and A. Ramesh, “Tomato crop disease classification using pre-trained deep learning algorithm,” Procedia Comput. Sci., vol. 133, pp. 1040–1047, 2018.

[22]. D. Oppenheim, G. Shani, O. Erlich, and L. Tsror, “Using deep learning for image-based potato tuber disease detection,” Phytopathology, vol. 109, no. 6, pp. 1083–1087, 2019.

[23]. S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” Front. Plant Sci., vol. 7, p. 1419, 2016.

[24]. J. G. ArnalBarbedo, “Plant disease identification from individual lesions and spots using deep learning,” Biosyst. Eng., vol. 180, pp. 96–107, Apr. 2019, doi: 10.1016/j.biosystemseng.2019.02.002.

[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, 2017.
[26]. A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, "A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition," Sensors, vol. 17, no. 9, p. 2022, Sep. 2017, doi: 10.3390/s17092022.
[27]. B. Chilwal and P. K. Mishra, "A Survey of Fuzzy Logic Inference System and Other Computing Techniques for Agricultural Diseases," in International Conference on Intelligent Computing and Smart Communication 2019, Singapore, 2020, pp. 1–6, doi: 10.1007/978-981-15-0633-8_1.
[28]. M. Kerkech, A. Hafiane, and R. Canals, "Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in UAV images," Comput. Electron. Agric., vol. 155, pp. 237–243, Dec. 2018, doi: 10.1016/j.compag.2018.10.006.
[29]. J. G. A. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," Comput. Electron. Agric., vol. 153, pp. 46–53, 2018.
[30]. M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning-based approaches for plant disease detection," Symmetry, vol. 11, no. 7, p. 939, 2019.
[31]. S. Zhang, X. Wu, Z. You, and L. Zhang, "Leaf image-based cucumber disease recognition using sparse representation classification," Comput. Electron. Agric., vol. 134, pp. 135–141, Mar. 2017, doi: 10.1016/j.compag.2017.01.014.
[32]. M. C. Fisher, N. J. Hawkins, D. Sanglard, and S. J. Gurr, "Worldwide emergence of resistance to antifungal drugs challenges human health and food security," Science, vol. 360, no. 6390, pp. 739–742, May 2018, doi: 10.1126/science. aap7999.
[33]. J. R. Lamichhane, M. P. You, V. Laudinot, M. J. Barbetti, and J.-N. Aubertot, "Revisiting Sustainability of Fungicide Seed Treatments for Field Crops," Plant Dis., p. PDIS-06-19-1157-FE, Sep. 2019, doi: 10.1094/PDIS-06-19-1157-FE.
[34]. D. A. Martinez, U. E. Loening, and M. C. Graham, "Impacts of glyphosate-based herbicides on disease resistance and health of crops: a review," Environ. Sci. Eur., vol. 30, no. 1, p. 2, Jan. 2018, doi: 10.1186/s12302-018-0131-7.
[35]. J. W. Bentley and G. Thiele, "Bibliography: Farmer knowledge and management of crop disease," p. 8.
[36]. J. Xiong, R. Lin, R. Bu, Z. Liu, Z. Yang, and L. Yu, "A Micro-Damage Detection Method of Litchi Fruit Using Hyperspectral Imaging Technology," Sensors, vol. 18, no. 3, 2018.
[37]. F. Dankhara, K. Patel, N. Doshi. Analysis of robust weed detection techniques based on the Internet of Things (IoT)[J]. Procedia Computer Science, 2019,160.
[38]. I. Mohanraj, K. Ashokumar, J. Naren. Field Monitoring and Automation Using IOT in Agriculture Domain[J]. Procedia Computer Science, 2016,93.
[39]. G. Laura, P. Lorena, M J J, et al. IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture.[J]. Sensors (Basel, Switzerland), 2020,20(4).
[40]. "2018 AI Challenger Crop Disease Identification." [Online]. Available:https://pan.baidu.com/s/1TH9qL7Wded2Qiz03wHTDLw#list/path=%2F. [Accessed: 12-Feb-2020].
[41]. H. Faris, I. Aljarah, and S. Mirjalili, "Training feedforward neural networks using multi-verse optimizer for binary classification problems," Appl. Intell., vol. 45, no. 2, pp. 322–332, Sep. 2016, doi: 10.1007/s10489-016-0767-1.
[42]. Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in Neural networks: Tricks of the trade, Springer, 2012, pp. 9–48.
[43]. X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010, pp. 249–256.
[44]. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[45]. K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.
[46]. "Initializers - Keras Documentation." [Online]. Available: https://keras.io/initializers/. [Accessed: 12-Feb-2020].
[47]. M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic Variational Inference," J. Mach. Learn. Res., vol. 14, pp. 1303–1347, 2013.
[48]. G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning lecture 6a overview of mini-batch gradient descent," Cited On, vol. 14, no. 8, 2012.
[49]. M. D. Zeiler, "ADADELTA: An Adaptive Learning Rate Method," Dec. 2012.
[50]. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," ArXiv14126980 Cs, Jan. 2017.
[51]. "Optimizers - Keras Documentation." [Online]. Available: https://keras.io/optimizers/. [Accessed: 12-Feb-2020].

WEI-JIAN HU received the B.S. degree in computer software and the M.S. degree in computer science and technology from Southwest JiaoTong University, Chengdu, China in 2013 and 2016, respectively. From 2015 to 2016, he was a Software R & D Engineer in General Electric; 2016-2019, he was a Software R & D Engineer in JD.COM and since 2019 he has been a Lecturer with the school of Information Engineering, Inner Mongolia University of Science and Technology, Baotou, China. Her research interest is computer vision and deep learning.

JIE FAN received the B.S. degree in computer science and technology from the Taiyuan Institute of Technology, Taiyuan, China, in 2018. He is currently pursuing the M.S. degree in computer science and technology with the Inner Mongolia University of Science and Technology, Baotou, China. His research interests include pattern recognition and computer vision.

YONG-XING DU received the B.S. degree in communication engineering from the Chengdu Univer-sity of Information Technology, Chengdu, China, in 2001, and the M.S. degree in electronic circuitand system and the Ph.D. degree in microelectron-ics and solid electronics from the Xi'an Universityof Technology, Xi'an, China, in 2007 and 2014, respectively. From 2001 to 2007, he was a Research Assistantand from 2007 to 2012, he was a Lecturer with theSchool of Information Engineering, Inner Mongolia University of Scienceand Technology, Baotou, respectively, where he has been a Professor since 2018. He is the editor of American Jorunal of Traffic and Transportation Engineering and Journal of Electronic Research and Application and the reviewer of IEEE ACCESS, IEEE MWCL, Optical and Quantum Electronics, International Journal of Electronics and Communications, Asia-Pacific Microwave Conference, Measurement and Control Technology, Journal of Optics, Laser Journal, and Optical Communication Technology. People; Senior member of Chinese Institute of Electronics. His main research fields are numerical calculation of electromagnetic fields, new microwave radiators, attenuation of black barrier effects, beam synthesis, etc.

BAO-SHAN LI was born in Tangshan, China, in 1965. He received the B.S. degree in radio technology from the Northeastern University, Shenyang, China, in 1986, and the M.S. degree in communication engineering from the Inner Mongolia University of Technology, Hohhot, China, in 2007. From 1986 to 1999, he was a Lecturer and from 1999 to 2008, he was an Associate Professor with the School of Information Engineering, Inner

Mongolia University of Science and Technology, Baotou, respectively, where he has been a Professor since 2008. He has authored over 30 articles. His current research interest is the application research of RFID and Internet of things technology.

NEAL N. XIONG received the Ph.D. degrees in sensor system engineering and in dependable sensor networks from Wuhan University and the Japan Advanced Institute of Science and Technology, respectively. Before he attended Tianjin University, he was with Northeastern State University, Georgia State University, the Wentworth Technology Institution, and Colorado Technical University (Full Professor for about five years) for about 10 years. He is currently a Professor with the College of Intelligence and Computing, Tianjin University, China. His research interests include cloud computing, security and dependability, parallel and distributed computing, networks, and optimization theory.

ERNST BEKKERING is current an Associate Professor at Department of Mathematics and Computer Science, Northeastern State University, OK, USA. He got his PhD degree in Mississippi State University, 2004. His research interests include Cloud Computing, Security and Dependability, Parallel and Distributed Computing, Networks, and Optimization Theory.