




Identifying plant diseases using deep transfer learning and enhanced lightweight network

Junde Chen¹  • Defu Zhang¹ • Y. A. Nanekaran¹

Received: 7 April 2020 / Revised: 14 July 2020 / Accepted: 18 August 2020

Published online: 21 August 2020

© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Plant diseases can cause significant reductions in both the quality and quantity of agricultural products, and they have a disastrous impact on the safety of food production. In severe cases, plant diseases may even lead to no grain harvest completely. Therefore, seeking fast, automatic, less expensive and accurate methods to detect plant diseases is of great realistic significance. In this paper, we studied the transfer learning for the deep CNNs and modified the network structure to enhance the learning ability of the tiny lesion symptoms. The pre-trained MobileNet-V2 was extended with the classification activation map (CAM), which was used for visualization as well as plant lesion positioning, and both were selected in our approach. Particularly, the transfer learning was performed twice in model training: the first phase only inferred the weights from scratch for new extended layers while the bottom convolution layers were frozen with the parameters trained from ImageNet; the second phase retrained the weights using the target dataset by loading the model trained in the first phase. Then, the yielded optimum model was used for identifying plant diseases. Experimental results demonstrate the validity of the proposed approach. It achieves an average recognition accuracy of 99.85% on the public dataset. Even under multiple classes and complex background conditions, the average accuracy reaches 99.11% on the collected plant disease images. Thus, the proposed approach efficiently accomplished plant disease identification and presented a superior performance relative to other state-of-the-art methods.

Keywords Plant disease identification · Transfer learning · Convolutional neural networks · Image classification

✉ Defu Zhang
dfzhang@xmu.edu.cn

¹ School of Informatics, Xiamen University, Xiamen 361005, China

1 Introduction

Plant diseases have a devastating effect on agricultural products, and if they are not detected in time, there will be an increase in food insecurity [11]. Especially, the main food crops, including rice and maize, are very important for guaranteeing the food supply and agricultural production. Apart from being a source of carbohydrates for humans, the main crops are also planted for animal food, cooking oil, industrial raw materials and so on [27]. Whereas, crops are quite susceptible by diverse diseases as well. Serious plant diseases have a disastrous impact on the plants and can cause grain failure entirely. The early warning or forecast is the basis of effective prevention and control for the plant diseases; it can reduce the cost of plant diseases and avoid unnecessary pesticide use. Therefore, obtaining information about real-time plant diseases is highlighted in modern agriculture. Using the information about plant diseases obtained from different sources, the diseases can be forecasted before its large-scale outbreak [44]. However, so far, plant disease monitoring in many regions, particularly developing countries, still relies on the traditional approach. When a plant disease suddenly happens somewhere, specialists or researchers from different institutions will visit the place and give advice to the farmers. Nevertheless, compared to the number of farmers, there are not adequate plant disease specialists in many areas. Also, affected by the social and economic environment, few people are engaged in plant protection originally. Thus, there are great needs and important realistic significances for seeking fast, easily available, and less expensive tools to automatically detect plant diseases.

With the improvement of digital cameras and the increasing computational capacity, computer vision technology is becoming an attractive approach for the automatic monitoring of plant diseases since it is a low cost, visualized, and non-contact manner [16]. Plenty of previous work has considered the image recognition, and a particular classifier is used which classifies the images into healthy or different disease types [1, 7, 23, 24, 35, 38]. For example, Sharif et al. [38] proposed a Multi-Class Support Vector Machine (M-SVM) to classify the citrus diseases. Including anthracnose, black spot, canker, scab, greening, and melanose, the six types of citrus diseases were categorized and they achieved 97% classification accuracy on citrus disease image dataset. Aurangzeb et al. [7] introduced an automated system for recognition of potato and corn leaf diseases. They used the handcrafted features as the input of various classifiers such as the SVM, linear discriminant, and ensemble tree (ESDA); their recognition accuracies are obtained in the range of 92.8% to 98.7% on chosen crops diseases of PlantVillage dataset. Anthonys et al. [5] used the method of nearest neighbor to classify the paddy disease types, and the concepts of the membership function were used in their work. They realized over 70% classification accuracy around 50 sample images. Kahar et al. [20] performed the rice disease detection using the artificial neural network (ANN) method, where 3 output nodes were employed for the detection of three diseases: leaf blight, leaf blast, and sheath blight. They achieved an accuracy of 74.21%. Chen and Wang [28] proposed a method for the identification of maize leaf diseases based on image processing technology and a probabilistic neural network (PNN); they achieved the best recognition accuracy as 90.4%, etc. More recently, deep learning techniques, particularly convolutional neural networks (CNNs), are becoming the preferred method in addressing most of the technical challenges associated with image classification. CNN is the most popular classifier for image recognition in both large scale and small scale problems, which has shown outstanding ability in image processing and classification [2, 3, 12, 26, 31]. Using the transfer learning techniques, Adeel et al. [2] fused two pre-trained CNN architectures, including AlexNet and ResNet101, to detect grape

leaf diseases on the PlantVillage dataset, and they achieved an accuracy of 99%. Ferentinos et al. [12] trained a deep learning model for recognizing 14 crop species and 26 crop diseases. Their trained model realized an accuracy of 99.35% on a held-out test set. X. Zhang et al. [45] introduced an improved deep convolutional neural network for identifying maize leaf diseases; it reached an average accuracy of 98.8%. Kamal et al. [21] used depth-wise separable convolution architectures to classify plant diseases on the PlantVillage dataset, and they realized the classification accuracy of 98.34%, etc. Although very good results have been reported for the CNN method in the literature, most of the existing studies focused on disease recognition from a public dataset rather than real field wild scenarios [8]. In reality, images are captured under a wide diversity of conditions and consist of heterogeneous backgrounds such as the surroundings of the field, soils of different colors, clutter leaves, etc. Moreover, most works were performed to detect the plant diseases based on plant leaf images instead of various parts of the plants. Indeed, plant diseases can occur in any part of the plant, whether it is leaf, stem or grain. Despite the limitations, the previous investigations have successfully demonstrated the potential of deep learning algorithms in plant disease identification. In this paper, we studied the transfer learning for the deep CNNs and modified the network with the aim of enhancing the learning ability of the tiny lesion symptoms. The enhanced MobileNet-V2 with CAM algorithm were selected in our approach. The top layers of MobileNet-V2 were truncated and a global average pooling (GAP) layer was added behind the pre-trained MobileNet-V2, which was followed by an additional convolutional layer of $5 \times 5 \times 1024$ for high-dimensional feature extraction. Then, it was followed by two fully-connected ReLU layers, which were composed of 1024 neurons and 512 neurons separately. After that, a fully-connected Softmax layer with the practical number of categories was added as the top layer of the modified networks and the CAM algorithm was employed for activated region presentation. In this way, the newly formed networks, which we termed the MobileNet-Beta, was used for the classification of plant diseases. Particularly, transfer learning was performed twice in model training: the first phase only inferred the parameters from scratch for new extended layers while the convolutional layers were frozen with the parameters trained on ImageNet [34]; the second phase retrained the parameters using the target dataset by loading the pre-trained model in the first phase. Thus, the yielded optimum model was used for plant disease identification.

The remainder of this paper is organized in the following manner. Section 2 presents the collection of the image dataset followed by an overall flow introduction, and this section primarily discusses the methodology to accomplish the task of plant disease identification as well as related concepts and the proposed approach. Section 3 dedicates to the algorithm experiments; multiple experiments are conducted and the experimental results are evaluated along with the comparative analysis. The paper is ultimately summarized in Section 4.

2 Materials and methods

2.1 Data acquisition

About 1000 plant disease images, which are composed of 500 rice disease images and 466 maize disease images, are provided by the Fujian Institute of Subtropical Botany, Xiamen, China. These images were captured with uneven illumination intensities and heterogeneous field wild scenarios. All the crop images collected in this paper are defined the disease

categories and have been saved as the JPG format. Among them, the rice plant diseases primarily include the rice stackburn, rice leaf scald, rice leaf smut, rice white tip, and rice bacterial leaf streak, etc. The maize plant diseases consist of Phaeosphaeria Spot, Maize Eyespot, Gray Leaf Spot, Goss's Bacterial wilt, and others. For the subsequent calculations, these plant disease images are uniformly processed into the RGB model by Photoshop tools firstly, and then the sizes of images are adjusted to 224×224 pixels. Some of the sample images are displayed in Fig. 1 and the symptom characteristics of various plant diseases are given in Table 1.

2.2 Overview

The overall process of our method for plant disease identification is presented as follows. Firstly, the samples of plant disease images are collected and labelled based on the experts' knowledge in the field. Then, the image processing techniques such as image resizing, image sharpening and image edge filling, etc., are performed on the acquired images. The sizes of the images are uniformly adjusted to 224×224 pixels to fit the model, the individual blurred images are conducted the image sharpening to make the image clearer, and the image edge filling is performed for the resized images to prevent the information loss. Next, using the data augmentation scheme, new sample images are generated to enrich the dataset. For example, color jittering, rotation, translation and scale transformation, etc. are utilized to generate the augmented dataset, and these operations of data augmentation are implemented using random bounded values. Here color jittering is changing the saturation, brightness, and contrast of color with a random adjustment factor in the interval of (0, 3.1), the rotation operation is conducted with an arbitrary angle in the range of $[0, 360^\circ]$, the image translation is in the range of $\pm 20\%$, and the scale is changed from 0.8 to 1.2. After that, the sample images are input to the proposed method for the model training, and then the trained model is employed to identify plant disease types. Thus, the final results of plant disease identification are obtained and the identified images are used to update the sample library again, as depicted in Fig. 2. The detailed descriptions of these phases are illustrated in subsequent sections.

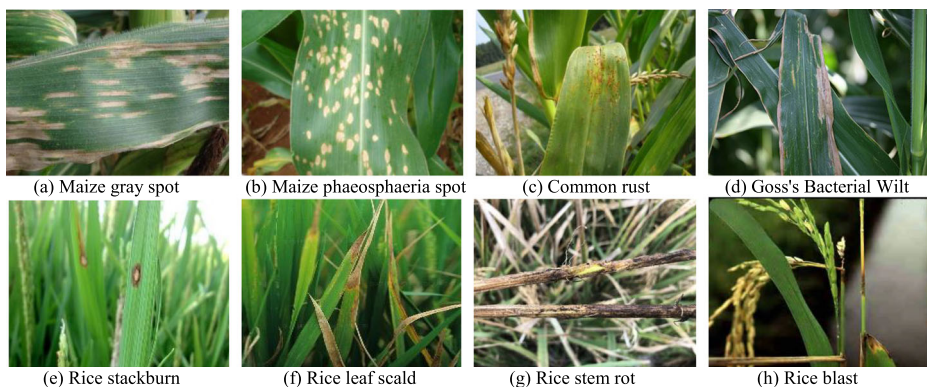


Fig. 1 Sample images of partial plant diseases, **a** Maize gray spot, **b** Maize phaeosphaeria spot, **c** Common rust, **d** Goss's Bacterial Wilt, **e** Rice stackburn, **f** Rice leaf scald, **g** Rice stem rot, **h** Rice blast

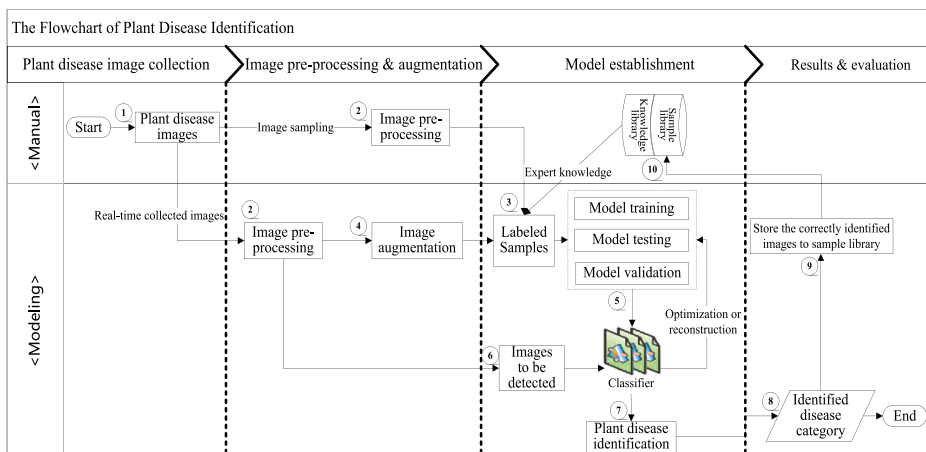
Table 1 The symptom characteristics of plant disease images

Species	Disease Name	The leaf disease characteristics		
		Shape	Color	Junction
Maize	Gray Leaf Spot	Irregular strip	Brown or straw-colored	Leaf
	Phaeosphaeria Spot	Circular or Ellipse	Offwhite or yellowish	Leaf
	Common Smut	Irregular nodules	Grey white	Corn on the cob
	Gibberella Ear Rot	Mycelium layer on seeds	Red mycelium	Corn on the cob
	Crazy Top	Tassel deformity	Dark green to grey	Corn cob
	Common Rust	Watery stain form or punctate	Rust or bronzing	Leaf
	Goss's Bacterial wilt	Long strip or block	Gray mildew to black	Leaf
	Maize Eyespot	Ellipse of sesame seed size	Centre: brown; Edge: dim pale yellow	Leaf
Rice	Rice stackburn	Circular or Ellipse	Yellowish-white macular	Leaf
	Leaf smut	Short strip	Black	Leaf
	Leaf scald	Watery stain form or strip	Offwhite or yellow	Leaf
	White tip	Strip	Centre: Ashen; Edge: Brown	Leaf tip
	Rice stem rot	Long strip (stem)	Black	Stem
	Rice straighthead	Crescent-like shape	Darker green	Grain
	Brown spot	Ellipse of sesame seed size	Centre: Brown to ashen; Periphery: Dim	Leaf
	Rice blast	Circular punctate	Brown	Leaf, Stem

2.3 Deep neural networks architecture

2.3.1 Depthwise separable convolution

Depthwise separable convolution is a form of factorable convolutions that factorizes a standard convolution into two steps: a depth-wise convolution and a 1×1 convolution called point-wise

**Fig. 2** The overall flow of plant disease identification

convolution [39]. In the first step, each channel conducts the convolutional operation with one filter for the input map, as depicted in Fig. 3a. The second is the point-wise convolution, which uses the results of depth-wise convolution to perform a 1×1 convolution kernel operation, and then obtains the output results, as shown in Fig. 3b. Mathematically, the operations can be separately written in Eqs. (1,2).

$$G_{i,j,d}^{l+1} = \sum_{i=0}^H \sum_{j=0}^W \sum_{d=0}^D K_{i,j,d} \times F_{i,j,d}^l \quad (1)$$

$$\hat{G}_{i,j,d}^{l+1} = \sum_{d=0}^D K_d \times \sum_{i=0}^H \sum_{j=0}^W K_{i,j} \times F_{i,j}^l \quad (2)$$

where F represents an input tensor, K is the convolutional kernel, and G is the convolution output results; H , W , and D are the height, width, and depth of the input for layer l in the networks. In particular, K_d denotes a 1×1 convolution kernel. Depthwise separable convolution has been explicitly incorporated in MobileNets [17], which are a family of mobile-first models meeting the resource liabilities for devices. Set the size of the input feature map is $D_f \times D_f$, the number of channels is M , the size of the convolutional kernel is $D_k \times D_k$, and the number of output channels is N . Then, the computation cost of traditional convolution (C_1) and the depth-wise convolution (C_2) can be expressed in Eqs. (3,4) respectively.

$$C_1 = D_f^2 M N D_k^2 \quad (3)$$

$$C_2 = D_f^2 M (D_k^2 + N) \quad (4)$$

Thus, the computational cost ratio of the depth-wise separable convolution to the traditional convolution is

$$R_{C_2/C_1} = \frac{1}{N} + \frac{1}{D_k \times D_k} \quad (5)$$

In practice, the 3×3 convolution kernel is frequently utilized in the depth-wise separable convolution networks, whereas the number of output channels N is big. Therefore, the computational cost of depth-wise separable convolution is around 1/9 of that of the standard convolution network.

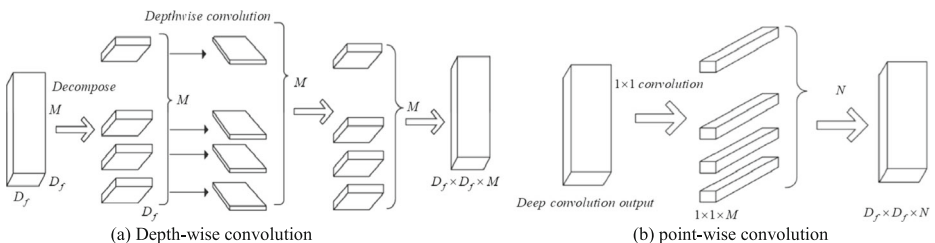


Fig. 3 The operation process of depthwise separable convolution, **a** Depth-wise convolution, **b** point-wise convolution

2.3.2 Classification activation map

Generated from the interaction between the final convolutional layer, GAP layer and classification layer, classification activation map (CAM) is reported by Zhou et al. [46] to visualize the regions that contain distinctive object parts. A CAM indicates the highest activated regions used by the CNN for classification. For example, let $f_k(x, y)$ represent the activation of unit k in the last convolution layer at spatial location (x, y) of a given image. Then, for unit k , the result of performing GAP is calculated in Eq. (6).

$$F^k = \sum_{x,y} f_k(x, y) \quad (6)$$

Thus, for a given class c , the input to the Softmax is given by

$$S_c = \sum_k \omega_k^c F^k \quad (7)$$

where ω_k^c is the weight corresponding to class c for unit k . Essentially ω_k^c represents the significance of F^k for class c , and the output of the Softmax for class c can be finally calculated in Eq. (8).

$$P_c = \exp(S_c) / \sum_c \exp(S_c) \quad (8)$$

Here the bias term is ignored by explicitly setting the input bias of the Softmax to 0 as it has little to no impact on the classification performance.

By plugging Eq. (6) into the class score S_c , Eq. (9) can be obtained.

$$S_c = \sum_k \left(\omega_k^c \cdot \sum_{x,y} f_k(x, y) \right) = \sum_{x,y} \sum_k \omega_k^c \cdot f_k(x, y) \quad (9)$$

The class activation map for class c is defined as M_c , where each spatial element can be written in

$$M_c(x, y) = \sum_k \omega_k^c f_k(x, y) \quad (10)$$

Hence, $M_c(x, y)$ directly indicates the significance of the activation at spatial grid (x, y) causing the identification of an image to class c . By simply up-sampling the CAM to the size of the input image, the image regions most relevant to the particular category can be identified.

2.3.3 Transfer learning

Transfer learning is the technique by which the knowledge gained during training in one type of problem is used to train in other related tasks or domains [32, 42]. Giving a source domain D_s with a corresponding source task T_s and a target domain D_t with a learning task T_t , the objective of transfer learning is to improve the performance of predictive function $f_T(\cdot)$ for learning task T_t by discovering and transferring latent knowledge from D_s and T_s , where $D_s \neq D_t$ or $T_s \neq T_t$ [32]. Transfer learning is particularly important in deep convolutional neural networks because the algorithms of deep learning require a large number of labeled data to train the models, while collecting of a large labeled dataset in a domain is undoubtedly a challenging task.

Thus, the scheme of transfer learning has increasingly become the standard approach in artificial intelligence (AI) and is naturally employed in the practical application, e.g. the solutions consisting in using a pre-trained network on the large ImageNet dataset where only

the parameters of new extended layers need to be inferred from scratch using the target dataset. A typical schematic diagram of the transfer learning for deep convolutional neural networks is depicted in Fig. 4.

2.3.4 Enhanced MobileNet-V2 architecture

As mentioned in Section 2.3.1, MobileNets are a family of lightweight convolutional neural networks based on depthwise separable convolution and have shown promising capability in addressing both large-scale and small-scale problems. Moreover, CAM indicates the highest activated regions used by CNNs for image classification and can ascertain the significance of the regions by projecting back the weights of output layers on to the convolution feature maps. Thus, inspired by the performances, the enhanced MobileNet-V2 [36] with CAM algorithm was selected in our approach. We modified the conventional MobileNet-V2 with the aim of enhancing the learning ability of the tiny lesion symptoms. The top layers of MobileNet-V2 were truncated and a global average pooling layer was added behind the pre-trained MobileNet-V2 networks, which was followed by an additional convolutional layer of $5 \times 5 \times 1024$ for high-dimensional feature extraction. Then, it was followed by two fully-connected ReLU layers, which were separately composed of 1024 neurons and 512 neurons for classification. After that, a fully-connected Softmax layer with the practical number of categories was added as the top layer of the modified networks and the CAM algorithm was employed for activated region presentation. In this way, the newly formed networks, namely MobileNet-Beta, which was used for the identification of plant disease types. More importantly, transfer learning was performed twice for the model training. In the first phase, only the parameters of new extended layers were inferred from scratch while the weights of convolutional layers pre-trained from ImageNet were frozen. The second phase retrained the weight parameters on the target dataset by loading the pre-trained model in the first phase. Specifically, the optimum model is obtained by applying the following processes.

1. An adaptation is performed for the networks. To solve the new classification problem, an adaptation is performed by freezing all the weights of the pre-trained layers, and the

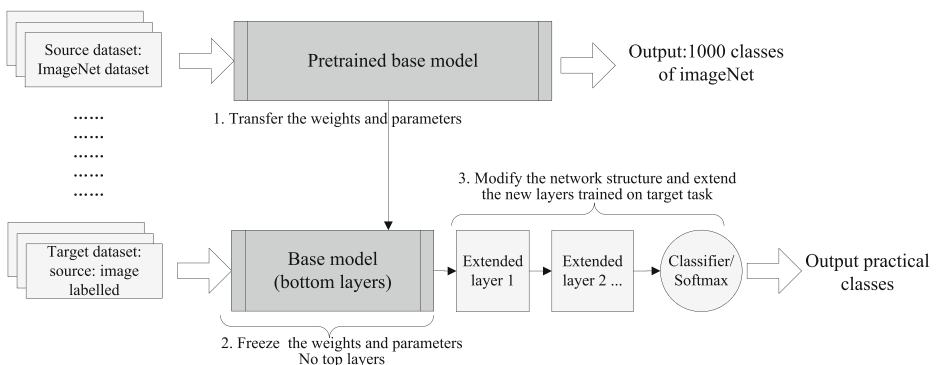


Fig. 4 The schematic diagram of transfer learning

extended layers are trained using the target dataset. In this stage, the Adam [25] which is a stochastic optimization algorithm is utilized to update the weights

$$w_k = w_{k-1} - \alpha * \hat{m}_k / \left(\sqrt{\hat{v}_k} + \varepsilon \right) \quad (11)$$

Where w represents the weight parameters, α is the learning rate, k is the index of classes, \hat{m}_k is the bias-corrected first moment and \hat{v}_k denotes the bias-corrected second moment.

2. The second stage trains the model on the target dataset. By loading the model trained in the first stage, the networks are retrained using the target dataset and all the layers are trained on the new images (target dataset). Likewise, SGD (stochastic gradient descent) [14] optimizer is used to update the weights, as defined in Eq. (12)

$$w_k = w_{k-1} - a(\partial L(w)/\partial w) \quad (12)$$

where w denotes the weight parameters, a is the learning rate, k is the index of classes, and $L(\cdot)$ indicates the average loss function.

3. Considering the tiny lesion symptoms and multi-classification tasks, the Focal function [29] is enhanced and used as the loss function of MobileNet-Beta network, as calculated in Eqs. (13,14).

$$L_{mult} = - \sum_{i=1}^C \alpha_i (1-p_i)^{\gamma} y_i \log(p_i) \quad (13)$$

$$y_i = \begin{cases} 1 & i = actual_class \\ 0 & i \neq actual_class \end{cases} \quad (14)$$

where C denotes the total number of classes, p_i is the predicted probability distribution, and α_i is the weighting factor. Figure 5 depicts the network structure and relevant parameters are presented in Table 2.

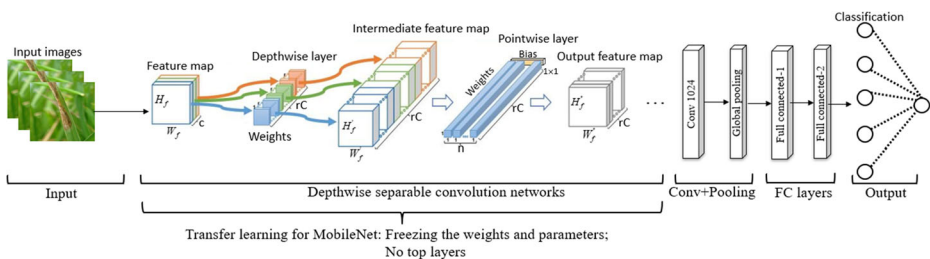


Fig. 5 The proposed network architecture

Table 2 The main parameters of networks

Module (type)	Input shape	Expansion factor	output channel no.	Repeated times	Stride
Input	224×224×3	—	3	1	—
Conv2d	224×224×3	—	32	1	2
Boottleneck	112×112×32	1	16	1	1
Boottleneck	112×112×16	6	24	2	2
Boottleneck	56×56×24	6	32	3	2
Boottleneck	28×28×32	6	64	4	2
Boottleneck	14×14×64	6	96	3	1
Boottleneck	14×14×96	6	160	3	2
Boottleneck	7×7×160	6	320	1	1
Conv2d 1×1	7×7×320	—	1280	1	1
Avgpool 7×7	7×7×1280	—	—	1	1
Conv2d 1×1	1×1×1280	—	1024	1	1
Conv2d 7×7	7×7×1024	—	1024	1	1
Globalpool	1×1×1024	—	—	1	1
FC1	1×1×1024	—	—	1	1
FC2	1×1×512	—	—	1	1
Softmax	1×1×k	—	k	1	1

3 Experimental results and analysis

In the experiments, except for some image pre-processing algorithms conducted by Photoshop or Matlab tools, the data augmentation and CNN were implemented using Anaconda3 (Python 3.6), Keras-GPU library [22], and OpenCV-python3 library. The deep CNN training and test were accelerated by GPU, and the experimental hardware environment includes: Intel® Xeon(R) E5–2620 v4 central processing unit (CPU) at 2.10 GHz with 64-GB memory and NVIDIA GeForce RTX 2080 (CUDA 10.2) graphics card [13], which was used for the program running.

3.1 Experiments on a public dataset

Plantvillage database (www.plantvillage.org) is an international general database for the algorithm test of plant disease detection using machine learning [19]. To evaluate the performance of the proposed approach, we conduct multiple experiments on this general database, which consists of different types of plants and their diseases. The PlantVillage database has 54,306 plant leaf images, with 26 diseases and 12 healthy plants for 14 crop species. Including Apple, Potato, Grape, Tomato, and Maize, etc., the 38 image datasets are downloaded from the Plantvillage database. It is noted that some images of the same leaf are captured from different orientations and the sizes of sample datasets are not uniform. The sample images are taken in simple background conditions and the dimensions of all the images are fixed as 256×256 pixels. Figure 6 displays the partial sample images.

Based on the method introduced in Section 2.3.4, we perform the model training and validation on the plant disease dataset. The dataset is split into two parts with the percentage ratio of 80% and 20% respectively. One is the modeling data and the other is the validation data, which is used for prediction and evaluation of the models. The choice of the split ratio is based on Mohanty et al. (2016) work [30]. Further, the modeling data is split into two parts: training and test sets with the ration of 80% and 20% respectively, to train the model and determine if the model is overfitting. To compare the proposed approach, we considered five

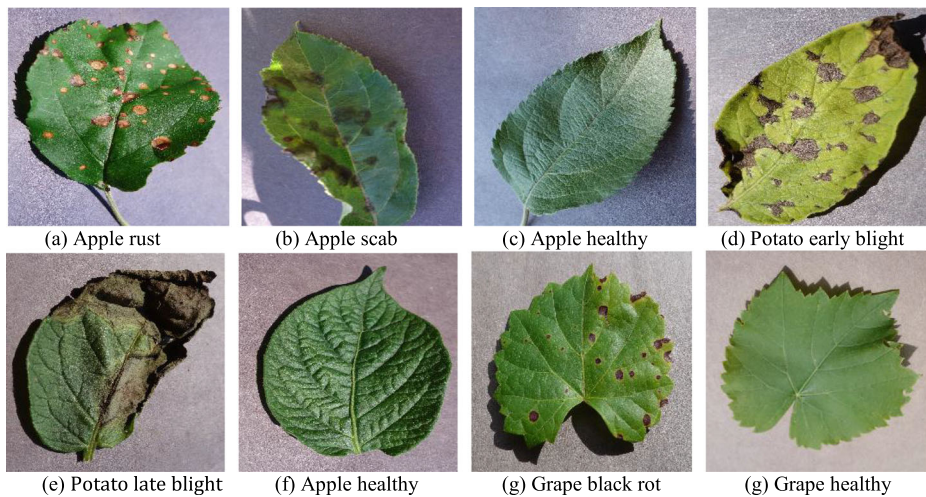


Fig. 6 Examples of the plant image dataset, **a** Apple, **b** Apple scab, **c** Apple healthy, **d** Potato early blight, **e** Potato late blight, **f** Apple healthy, **g** Grape black rot, **h** Grape healthy

influential CNNs, including VGGNet [40], Inception V3 [41], ResNet [15], DenseNet [18], and MobileNet-V2 [36], for the comparative experiments. Using the transfer learning, these models are created and loaded with pre-trained weights from the ImageNet dataset, and the top layers are truncated by defining a new fully-connected Softmax layer with the practical number of classifications. Additionally, referring to the approach proposed by Sethy et al. [37], the CNNs were used to extract the features while SVM was used for classification purpose. Also, as their statements of experimental results, the resnet50 plus SVM was the superior classification model with the F1-score among deep feature approaches, and the accuracy of mobilenetv2 plus SVM is comparative enough with resnet50 plus SVM. So, these two methods are further selected in our experiments for the comparative analysis. Thus, the models of diverse CNNs are trained and multiple experiments are conducted on the dataset. The training and test accuracies of different approaches are obtained in Table 3.

From Table 3, it can be concluded that the proposed approach outperforms the other well-known CNNs on the experimental dataset even if the optimal classifiers are adopted. The main

Table 3 The training accuracy and loss of different methods on the public dataset

Pre-trained model	10 epochs			30 epochs			
	Training accuracy %	Test accuracy %	Training loss	Training accuracy %	Test accuracy %	Training loss	Test loss
Inception V3	99.64	98.08	0.1227	99.92	98.41	0.0268	1.0182
VGGNet-19	96.16	93.94	0.2282	97.52	94.10	0.1866	0.3224
ResNet-50	99.01	97.95	0.0344	99.83	98.13	0.0062	0.0630
DenseNet-121	99.81	98.74	0.0806	99.95	98.89	0.0154	0.6133
MobileNet-V2	98.57	98.74	1.3626	99.40	98.97	0.7897	1.0179
ResNet50 + SVM	99.37	98.12	0.6779	99.90	98.25	0.0247	0.1965
MobileNetV2 + SVM	99.22	94.37	0.0567	99.77	97.78	0.0197	0.1820
Proposed approach	99.25	98.96	0.8586	99.83	99.12	0.4756	0.7288

explanation for this is that the proposed approach not only makes use of the generic characteristics of images extracted from the ImageNet by pre-trained networks but also takes advantage of specific features of the experimental dataset through the newly extended layers. Moreover, the transfer learning is performed twice for model training, which learns the weight parameters entirely and makes the optimum model be obtained, as depicted in Fig. 7a. By contrast, the other approaches are individual networks. Although the weights are initialized with pre-trained models instead of random initial weights and the fine-tuning is applied, the optimal results of these models are not achieved because the specific features of the target dataset are not completely learned. Thus, the proposed approach, which fully trains the network parameters and avoids the overfitting problem, achieves the top performance in the experiments of the public dataset.

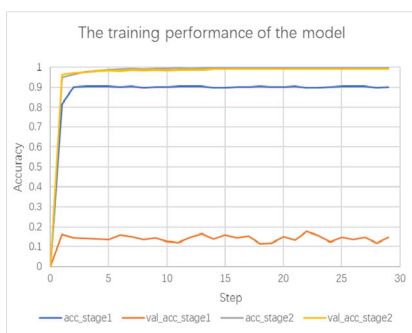
Furthermore, the optimum model obtained by the proposed approach is employed to identify the unseen images, and the partial plant species, including the Apple, Grape, and Potato, are selected to verify the validity. Figure 7b depicts the corresponding confusion matrices of the identification results. Considering the statistics of correct detections (also known as true positives), misdetections (also known as false negatives), true negatives and false positives, we can evaluate the performance of the models with the indicators, including the *Accuracy*, *Sensitivity*, and *Specificity*, as defined in Eqs. (15–17).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (15)$$

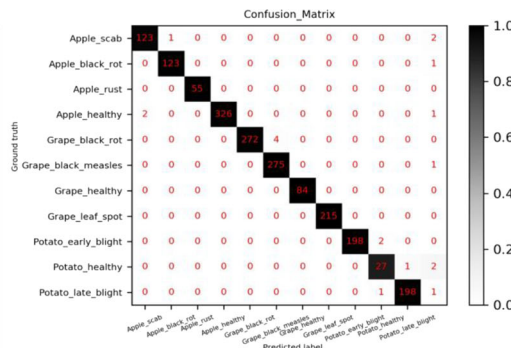
$$Sensitivity = \frac{TP}{TP + FN} \quad (16)$$

$$Specificity = \frac{TN}{FP + TN} \quad (17)$$

where true positive (TP) denotes the number of instances that the classifier accurately identifies; false negative (FN) is the reverse, which is the number of instances that are incorrectly identified; false positive (FP) represents the number of cases which are mistakenly marked as such; true negative (TN) is the number of samples that they are not in such disease



(a) Model training performance.



(b) Confusion matrix of identified results

Fig. 7 ROC curve and confusion matrix of plant disease identification, **a** Model training performance, **b** Confusion matrix of identified results

and accurately identified by the classifier. Table 4 gives the actual analysis of the results which are assessed in the form of different metrics.

As seen in Table 4, most of the samples in each class are correctly identified by the proposed approach, and the average *Accuracy* of the experimental results achieves 99.85% on the public dataset, which presents that the proposed approach has an impressive capability to recognize the plant diseases in the simple background condition. Particularly, in addition to identifying whether the plant is healthy or diseased, the proposed approach also discriminates the specific disease types of plant disease images, and the average *Sensitivity* and *Specificity* reach 99.01% and 99.94% separately.

Additionally, Table 5 shows the experimental results of some recent research performed on the PlantVillage dataset since 2016. The highest accuracy rate and corresponding method are listed in the table. Most of the literature do not use all the data of the PlantVillage dataset, so the number of classifications is much smaller than the 38 classes of this dataset. In this work, all the plant types are used for the experiments and the total of 38 categories are involved in the model training. Thus, even in the case of multiple categories, a relatively high accuracy rate is achieved for the proposed approach, which shows an impressive performance compared with the other state-of-the-art methods.

3.2 Experiments on the collected dataset

Similar to the experiments conducted in Section 3.1, the proposed approach was further tested on the collected plant dataset, which was captured in real-life agricultural scenarios with clutter field background conditions and non-uniform illumination intensities, such as the surroundings of the field, soils of different colors, taking in overcast or sunny weather, etc. The original images are divided into a training set and a test set with the ratio of 8:2 except for a certain number of raw images retaining for the evaluation of the model. That is, the dataset is split into the training and test sets separately to train the model and determine if the model is overfitting, while the new images outside modeling are applied to validate the effectiveness of the model. In particular, to ensure the diversity of sample images and avoid the overfitting problem, the data augmentation scheme is employed to enrich the dataset and at least 200 images are guaranteed for each category. Specifically, the detailed processes are described below.

Table 4 The identified results of different plant disease types

No.	Category names	Identified samples	Correct samples	Accuracy (%)	Sensitivity (%)	Specificity (%)
1	Apple scab	126	123	99.73	97.61	99.88
2	Apple black rot	124	123	99.89	99.19	99.94
3	Apple rust	55	55	100.00	100.00	100.00
4	Apple healthy	329	326	99.84	99.08	100.00
5	Grape black rot	236	272	99.79	98.55	100.00
6	Grape black measles	276	275	99.73	99.63	99.76
7	Grape healthy	84	84	100.00	100.00	100.00
8	Grape leaf spot	215	215	100.00	100.00	100.00
9	Potato early blight	200	198	99.89	99.00	100.00
10	Potato healthy	30	27	99.68	90.00	99.84
11	Potato late blight	200	198	99.84	99.00	99.94
—	Average	—	—	99.85	99.01	99.94

Table 5 Compared with the latest literature results on the public PlantVillage dataset

Researches	Year	Model	No. of classes	Accuracy
(Mohanty et al.) [30]	2016	GoogLeNet	26	99.35%
(Amara et al.) [4]	2017	LeNet	3	98.61%
(Brahimi et al.) [9]	2017	GoogLeNet	9	99.18%
(Durmuş et al.) [10]	2017	AlexNet	10	95.65%
(Voulodimos et al.) [43]	2018	VGG16	4	90.4%
(Rangarajan et al.) [33]	2018	VGG16	6	96.19%
(Too et al.) [42]	2018	DenseNets	38	99.75%
(Arsenovic et al.) [6]	2019	PlantDiseaseNet	18	93.67%
Proposed approach	2020	MobileNet-Beta	38	99.85%

1. Change the size of the images. All the images are resized to the fixed dimension of 224×224 pixels to fit the model, and the plant disease images are enlarged using the data augmentation techniques, such as color jittering, random rotation, flipping, scale transform, etc.
2. Image pre-processing. After resizing the images, the image pre-processing is performed to blacken the shorter sides of the images so that they become the same proportion, preserving the information of original images and avoiding image distortion.
3. Model training and test. Using the method proposed in Section 2.3.4, the MobileNet-Beta is trained and tested. To obtain the optimum model, multiple experiments are conducted with the shuffling of images.
4. Model validation. The new images outside modeling are applied to validate the effectiveness of the model, and the output results are compared with the actual category of images. The related evaluation indicators are calculated as well.

Thus, referring to the above processes, we performed the model training and the validation of the new unseen images. Figure 8 depicts the ROC curve and confusion matrix of identification results and the corresponding evaluation indicators are calculated in Table 6.

From Fig. 8a, it can be easily observed that the proposed approach shows ideal operating points in the upper, left corner of the ROC curve. The curve of each class is basically close to

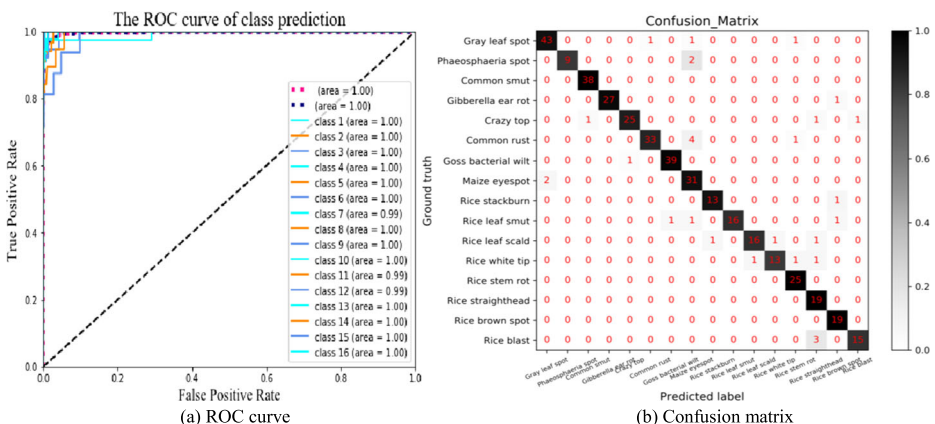
**Fig. 8** ROC curve and confusion matrix of the identification results, **a** ROC curve **(b)** Confusion matrix

Table 6 The evaluation indicators of plant disease identification

Species	No.	Maize disease classes	Predicted samples	Corrected samples	Accuracy (%)	Sensitivity (%)	Specificity (%)
Maize	1	Gray Leaf Spot	46	43	98.77	93.47	99.44
	2	Phaeosphaeria Spot	11	9	99.50	81.81	100.00
	3	Common Smut	38	38	99.75	100.00	99.72
	4	Gibberella Ear Rot	28	27	99.75	96.42	100.00
	5	Crazy Top	28	25	99.01	89.28	99.73
	6	Common Rust	38	33	98.52	86.84	99.72
	7	Goss's Bacterial wilt	40	39	99.50	97.50	99.72
	8	Maize Eyespot	33	31	97.54	93.93	97.86
	9	Rice stackburn	14	13	99.50	92.85	99.74
	10	Leaf smut	19	16	99.50	88.88	100.00
	11	Leaf scald	19	16	99.01	84.21	99.74
	12	White tip	16	13	99.01	81.25	99.75
	13	Rice stem rot	25	25	99.50	100.00	99.47
	14	Rice straighthead	19	19	98.52	100.00	98.45
	15	Brown spot	19	19	99.26	100.00	99.22
	16	Rice blast	18	15	99.01	83.33	99.74
	–	Average	–	–	99.11	92.92	99.52
Rice							

the upper left corner, and the TPR (true positive rate) is higher when its FPR (false positive rate) is lower, indicating the effectiveness of the proposed approach. In addition, as seen in Fig. 8b, most of the samples in each class are correctly identified by the MobileNet-Beta model. For example, 43 sample images are correctly identified in the class of “Gray Leaf Spot” except for 3 misclassified samples. The correct number is 9 for the classification of “Phaeosphaeria Spot” in 11 samples, and all the 38 samples are correctly identified for the class of the “Common Smut”, etc. Thus, there are total 381 plant disease images that are correctly classified by the proposed approach in 411 samples. The average identification *Accuracy* achieved 99.11% on the validation dataset, and the average *sensitivity* realized no less than 92.92% too, as shown in Table 6. Conversely, there are also individual plant disease types misclassified as other categories. Such as the “Gray Leaf Spot” disease, the samples are misclassified to the category of “Maize Eyespot”, which is due to the scenario that some diseases including “Gray Leaf Spot” and “Maize Eyespot” may occur in the same plant. Besides, the heterogeneous background conditions and irregular lighting strengths can also affect the identification results. The partial identified sample images are displayed in Fig. 9.

As seen in Fig. 9, the top images are the original sample images, the middle images are the positioning images conducted by the CAM algorithm, and the bottom images are the results of identification. Such as Fig. 9a~d), the identified categories are consistent with the actual categories of these samples, and most plant diseases are correctly classified by the proposed approach with a high probability except for individual images. As mentioned above, the individual samples are misclassified due to some different diseases occurring in the same plant. In addition, the extreme clutter field backdrops and uneven illumination intensities, which affect the feature extraction of lesion images, can also cause the individual incorrect classification, as shown in Fig. 9e. Although the

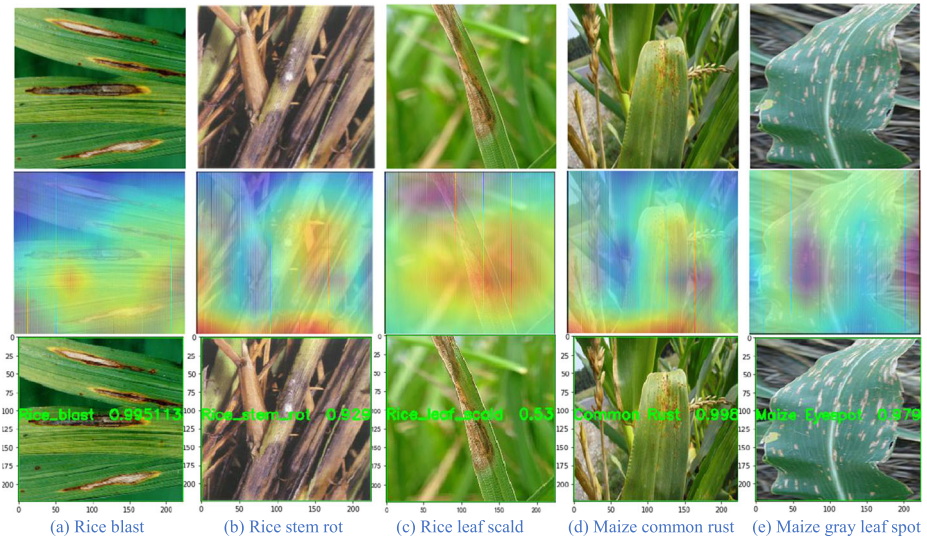


Fig. 9 The partial detected results of plant disease types, **a** Rice blast, **b** Rice stem rot, **c** Rice leaf scald, **d** Maize common rust, **e** Maize gray leaf spot

individual images are misclassified, most of the plant disease types are correctly identified by the proposed approach, and the high accuracy is achieved on the unseen images for multiple experiments, which indicates that the proposed MobileNet-Beta approach has a significant capability to recognize the plant diseases. Based on the experimental results, it can be assumed that the proposed approach is effective for the identification of plant disease types and can also be extended to the application of other fields, such as online fault diagnosis, target recognition, etc.

4 Conclusions

The identification and classification of plant diseases using digital images is of great significance to improve the quality of plant production. Deep learning techniques, and CNNs, in particular, have shown promising performance in addressing most of the technical challenges associated with image recognition. Therefore, in this paper, a novel deep learning architecture called MobileNet-Beta is proposed for the identification of plant disease types. The transfer learning for the deep convolutional neural networks was studied and the conventional MobileNet-V2 was modified with the aim of enhancing the learning capability of the tiny lesion features. The top layers of MobileNet-V2 were truncated and a global average pooling layer was added behind the pre-trained MobileNet-V2 networks, which was followed by an additional convolutional layer of $5 \times 5 \times 1024$ along with two fully-connected ReLU layers composed of 1024 neurons and 512 neurons separately. After that, a fully-connected Softmax layer with the practical number of categories was added as the top layer of the modified networks, and the CAM algorithm was used for visualization as well as plant lesion positioning for assisted manual re-inspection. In particular, transfer learning was performed twice for the

model training: the first phase only inferred the parameters from scratch for new extended layers while the convolutional layers were frozen with the parameters trained on ImageNet; the second phase retrained all the parameters using the target dataset by loading the pre-trained model in the first phase. Thus, the yielded optimum model was used to detect crop diseases. Experimental findings demonstrated the promising performance of the proposed approach on both the public dataset and the collected local dataset. In future development, we intend to deploy it on mobile devices to monitor and identify the wide range of plant disease information automatically. Meanwhile, we plan to apply it on more real-world applications.

Acknowledgments This work is partly supported by the grants from the National Natural Science Foundation of China (Project no. 61672439) and the Fundamental Research Funds for the Central Universities (#20720181004). The authors would like to thank all the editors and anonymous reviewers for their constructive advice.

Compliance with ethical standards

Conflict of interest The authors declare no conflicts of interest.

References

1. Adeel A et al (2019) Diagnosis and recognition of grape leaf diseases: An automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. *Sustain Comput: Inform Syst* 24:100349
2. Adeel A et al. (2020) Entropy-controlled deep features selection framework for grape leaf diseases recognition. *Expert Syst*
3. Alghamdi A et al. (2020) Detection of myocardial infarction based on novel deep transfer learning methods for urban healthcare in smart cities. *Multimed Tools Appl*: 1–22
4. Amara J, Bouaziz B, Algergawy A (2017) A deep learning-based approach for banana leaf diseases classification. *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband*
5. Anthony G, Wickramarachchi N (2009) An image recognition system for crop disease identification of paddy fields in Sri Lanka. 2009 International Conference on Industrial and Information Systems (ICIIS). IEEE
6. Arsenovic M et al (2019) Solving current limitations of deep learning based approaches for plant disease detection. *Symmetry* 11(7):939
7. Aurangzeb K et al. (2020) Advanced Machine Learning Algorithm Based System for Crops Leaf Diseases Recognition. 2020 6th Conference on Data Science and Machine Learning Applications (CDMA). IEEE
8. Barbedo JGA (2018) Factors influencing the use of deep learning for plant disease recognition. *Biosyst Eng* 172:84–91
9. Brahimi M, Boukhalfa K, Moussaoui A (2017) Deep learning for tomato diseases: classification and symptoms visualization. *Appl Artif Intell* 31(4):299–315
10. Durmuş H, Güneş EO, Kırıcı M (2017) Disease detection on the leaves of the tomato plants by using deep learning. 2017 6th International Conference on Agro-Geoinformatics. IEEE
11. Faithpraise F et al (2013) Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters. *Int J Adv Biotechnol Res* 4(2):189–199
12. Ferentinos KP (2018) Deep learning models for plant disease detection and diagnosis. *Comput Electron Agric* 145:311–318
13. GeForce GTX 1060. Available online: <https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1060/specifications> (Accessed on 17 Jun 2019).
14. Ghazi MM, Yanikoglu B, Aptoula E (2017) Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing* 235:228–235
15. He K et al. (2016) Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*

16. Hemming J, Rath T (2001) PA—precision agriculture: computer-vision-based weed identification under field conditions using controlled lighting. *J Agric Eng Res* 78(3):233–243
17. Howard AG et al. (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*
18. Huang G et al. (2017) Densely connected convolutional networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*
19. Hughes D, Salathé M (2015) An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*
20. Kahar MA, Motalib S, Abdul-Rahman S (2015) Early detection and classification of paddy diseases with neural networks and fuzzy logic. *Proceedings of the 17th International Conference on Mathematical and Computational Methods in Science and Engineering, MACMESE*
21. Kamal KC et al (2019) Depthwise separable convolution architectures for plant disease classification. *Comput Electron Agric* 165:104948
22. Keras-GPU. Available online: <https://anaconda.org/anaconda/keras-gpu> (Accessed on 17 Jun 2019)
23. Khan MA, Lali MIU, Sharif M, Javed K, Aurangzeb K, Haider SI, Altamrah AS, Akram T (2019) An optimized method for segmentation and classification of apple diseases based on strong correlation and genetic algorithm based feature selection. *IEEE Access* 7:46261–46277
24. Khan MA et al. (2020) An automated system for cucumber leaf diseased spot detection and classification using improved saliency method and deep features selection. *Multimed Tools Appl*: 1–30
25. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*
26. Kussul N, Lavreniuk M, Skakun S, Shelestov A (2017) Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geosci Remote Sens Lett* 14(5):778–782
27. Kusumo BS et al. (2018) Machine Learning-based for Automatic Detection of Corn-Plant Diseases Using Image Processing. 2018 International Conference on Computer, Control, Informatics and its Applications (IC3INA). IEEE
28. Li C, Wang L (2011) Research on Application of Probability Neural Network in Maize Leaf Disease Identification [J]. *J Agric Mechan Res* 6
29. Lin T-Y et al. (2017) Focal loss for dense object detection. *Proceedings of the IEEE international conference on computer vision*
30. Mohanty SP, Hughes DP, Salathé M (2016) Using deep learning for image-based plant disease detection. *Front Plant Sci* 7:1419
31. Nestor T et al (2020) A multidimensional hyperjerk oscillator: Dynamics analysis, analogue and embedded systems implementation, and its application as a cryptosystem. *Sensors* 20(1):83
32. Pan SJ, Yang Q (2009) A survey on transfer learning. *IEEE Trans Knowl Data Eng* 22(10):1345–1359
33. Rangarajan AK, Purushothaman R, Ramesh A (2018) Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Comp Sci* 133:1040–1047
34. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252
35. Safdar A, Khan MA, Shah JH, Sharif M, Saba T, Rehman A, Javed K, Khan JA (2019) Intelligent microscopic approach for identification and recognition of citrus deformities. *Microsc Res Tech* 82(9): 1542–1556
36. Sandler M et al. (2018) Mobilenetv2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE conference on computer vision and pattern recognition*
37. Sethy PK et al (2020) Deep feature based rice leaf disease identification using support vector machine. *Comput Electron Agric* 175:105527
38. Sharif M, Khan MA, Iqbal Z, Azam MF, Lali MIU, Javed MY (2018) Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. *Comput Electron Agric* 150:220–234
39. Sifre L, Mallat S (2014) Rigid-motion scattering for image classification. Ph. D. thesis
40. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
41. Szegedy C et al. (2016) Rethinking the inception architecture for computer vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*
42. Too EC, Yujian L, Njuki S, Yingchun L (2019) A comparative study of fine-tuning deep learning models for plant disease identification. *Comput Electron Agric* 161:272–279
43. Voulodimos A et al (2018) Recent developments in deep learning for engineering applications. *Comput Intell Neurosci* 2018:1–2
44. Wang X, Zhang X, Zhou G (2017) Automatic detection of rice disease using near infrared spectra technologies. *J Ind Soc Remote Sens* 45(5):785–794

45. Zhang X, Qiao Y, Meng F, Fan C, Zhang M (2018) Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access* 6:30370–30377
46. Zhou B et al. (2016) Learning deep features for discriminative localization. *Proceedings of the IEEE conference on computer vision and pattern recognition*

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