Artificial Intelligence Based Learning Approach for Leaf Disease Identification and Detection

G. Karuna, K. Sahithi, B. Rupa, R. Amani, K. Swaraja, and K. Meenakshi

Abstract Plants and Crops get diseased due to many reasons. It might be because of diseases of stems, leaves, roots etc. This Paper mainly congregates on leaves. Leaf Disease identification and Detection has many applications for cultivators and farmers to know whether the plant is diseased or not. So that they can retort in dwarf time and decreasing the loss and then can obtain immense profits. This paper mainly focused at learning the disease of plant through leaves. Here, we scrutinize the leaf through Image Processing and extract features of particular leaf and then utilizing those features as a dataset and done preprocessing and then administering them in Artificial Intelligence based learning algorithms like Convolutional Neural Networks to find disease.

Keywords Leaf disease · Neural networks · Features

1 Introduction

In the Earth, India is the second powerful Country in terms of population and also Food is one of the essential one to a person in order to survive. We are acquiring food from plants and so we have to safeguard our plants from diseases. Diseases of plants are normally due to insects, pests, pathogens and reduce the fertility to an extreme extent if not managed with in time.

Now-A-Days technology plays important role in most of the fields yet till this day we are applying some past approaches in farming. Knowing the plant's condition plays a vital role for productive farming [1]. Detection is happened normally by the scientists but due to the so many changes in the environment, the detection is becoming tough. So we can practice processing of image techniques for recognition of disease of a plant. Normally we can get to know the symptoms of diseases on stems, leafs, flowers etc. so here we mainly utilize leaves for recognition of diseased plants.

G. Karuna (✉) · K. Sahithi · B. Rupa · R. Amani
Computer Science and Engineering, GRIET, Hyderabad, India

K. Swaraja · K. Meenakshi
Electronics and Communications Engineering, GRIET, Hyderabad, India

A. Kumar et al. (eds.), *ICDSMLA 2020*, Lecture Notes in Electrical Engineering 783,
https://doi.org/10.1007/978-981-16-3690-5_77

Leaf Disease Identification is affiliate to domain Data Science. Here, Convolutional Neural Networks is taken in the Data science [2].

If we check a picture of a man, there can be a lot many features varying two persons. We could not give each and every feature as input to some sort of ML Algorithm. Even though, we give most of the features there would be some of other features which we may be missing. So, Here Deep Learning arrives into existence [3]. If we consider an image of 20*20 pixel, No of inputs to the Normal Neural Network is 20 * 20 = 400. 400 is a minute value and it is satisfactory. If we examine a picture of 300 * 300 pixel. No of inputs is 300 * 300 = 90,000 which is large number and it is very complex kind of thing to a Normal Neural Network and so that's the reason we are administering Convolutional NNs.

The learning ability of the human brain, which consists of neurons linked by synapses, motivates artificial neural networks. For images, ANNs are not appropriate since these networks lead to over-fitting due to image size. The major discrimination between the Artificial Neural Networks (ANN) and Convolution Neural Networks (CNN) is that only the last layer of a CNN is completely connected, while in ANN, each neuron is connected to every other neuron. Convolution neural networks directly use images as an input. Convolutional neural networks are used instead of handcrafted features to automatically learn a hierarchy of features that can then be used for classification purposes. This is achieved by successively converging the input image to create a hierarchy of feature maps with learned philtres. CNNs are different layered oversaw networks which can know features directly from datasets. For the recent days, CNNs have obtained superb production in almost all vital determination tasks. It can do one classification and the other feature extraction below the same network architecture [4].

2 Related Work

There are many strategies of management like disease particular chemical applications, vector control through pesticide applications and fungicide applications which gives early data on health of plant and disease detection. This keeps us diseases in control and improve productivity.

For plant disease identification, Number of methods are presently in usage applying computer vision. One among them is detection of disease by taking color features [5] or by extracting texture features. Features of textures such as Homogenity, Inertia, Correlation obtained from co–occurence on image by calculating gray level [6]. Many of the works like using image processing disease recognition approach, microscopy, double-stranded RNA(ribonucleic acid) analysis and nucleic acid probes [7–10] and also others like Particle Swarrn Optimisation (PSO) [11, 12]. We can also apply Support Vector Machines [13] and also K-Means Algorithm as a Clustering Method Algorithm Proposed by authors of [14].

3 Convolution Neural Networks

In the most of neural networks, Convolutional Neural Network (ConvNets or CNNs) is that the one among the most categories to strive to images recognition, image classification. A picture as an input as set of pixels is watched by Computers and it relays on the picture resolution. On the basis of picture resolution, it'll see h * w * d (h = height w = width d = dimension). Figure 1 refers to RGB values.

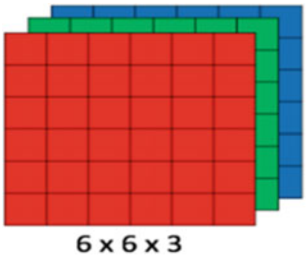
Technically Deep Learning Convolution Neural Network models check each and every input picture will permit it through various steps of layers of convolution with Kernels (filters) such as pooling, fully connected layers (FC) and utilize Softmax or any other methods to determine an object with probabilistic values between 0 and 1. Figure 2 is an overall flow of CNN to undertake a picture as an input and determines the objects supported prices.

Convolution Layer

Convolution is that the top most layer to take features from an image as an input. Convolution retains the interrelation among pixels by knowing picture features utilizing tiny squares of input file. It is a math kind of process that considers two inputs like image matrix and a filter or kernel.

Picture Convolution with different kernels can undergo operations like blur, edge detection and sharpen by using kernels. The following example gives different convolution pictures after using various kinds of kernels.

Fig. 1 RGB Layers



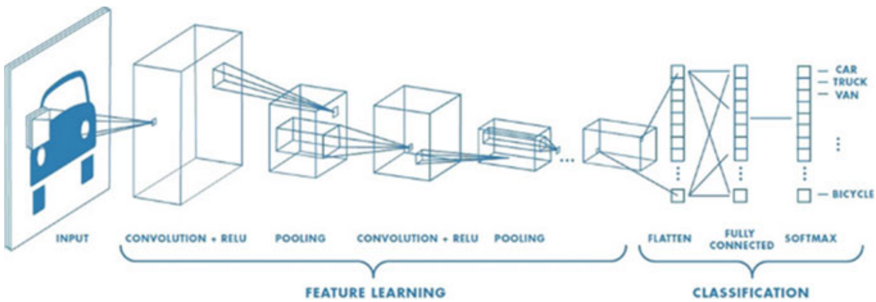


Fig. 2 Neural network with many convolutional layers

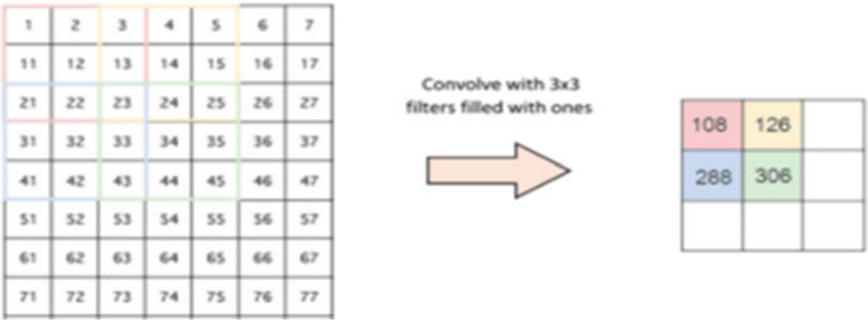


Fig. 3 Stride of two

4 Strides

Stride is that the no. of pixels moves through the input matrix. When the stride is 1 then we proceed the kernels to 1 pixel at a time. When the stride is 2 then we proceed the kernels to 2 pixels at a time then on. Figure 3 displays convolution can be done with a stride of two.

Padding
Kernel doesn't exactly fit the input picture sometimes. We have two options:

- Add the image with zeros (zero-padding) in order that it fits
- Drop the part of the picture where the kernel didn't fit. This is known as valid padding which provides only valid a fraction of the picture.

Non Linearity (ReLU)
ReLU full form is Rectified linear measure for a non-linear operation. The end result is $f(x) = \max(0,x)$. ReLU's purpose is to provide non-linearity in our ConvNet. Because, the important global info would want our ConvNet to find out would be positive linear values. Some kind of other non linear functions like sigmoid or tanh which will even be applied rather than ReLU. Most of the info researchers use ReLU because performance wise ReLU is best than the opposite two.

5 Pooling Layer

Pooling layers section would scale back the amount of parameters when the pictures are overlarge. Spatial pooling also known as to be downsampling or subsampling which trims the dimensionality of each and every map but retains essential information. Spatial pooling could be of variety of kinds:

- Max Pooling

- Sum Pooling
- Average Pooling.

Max pooling takes the most important item from the rectified feature map. Considering the most important item could also provide the typical pooling. Addition of all items within the feature map known as sum pooling (Fig. 4).

The layer which we say as FC layer, we flattened our matrix into vector and give it into a totally layer of connected sort of a neural network (Fig. 5).

In the above figure, matrix of the feature map are going to be converted into vector (× 1, × 2, × 3). With the fully connected layers, we combined all of these features together to produce a model. Ultimately, we have an activation method as sigmoid or softmax to determine the outputs as cat, dog, car, truck etc. (Fig. 6).

Related Background Work

There are many network architectures proposed like LeNet, AlexNet, GoogleLeNet etc. used for image recognition. The LeNet network architecture is the initial Convolutional Neural Network introduced by LeCun et al. to identify digits written by

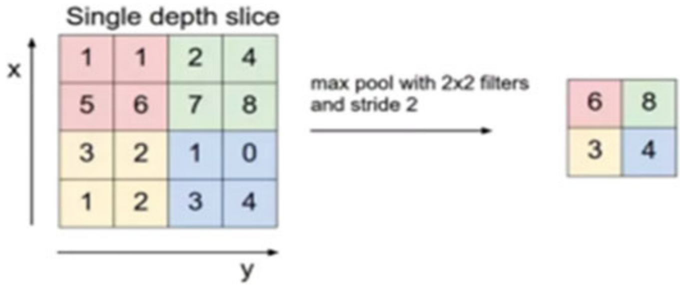


Fig. 4 Max Pooling with stride of two

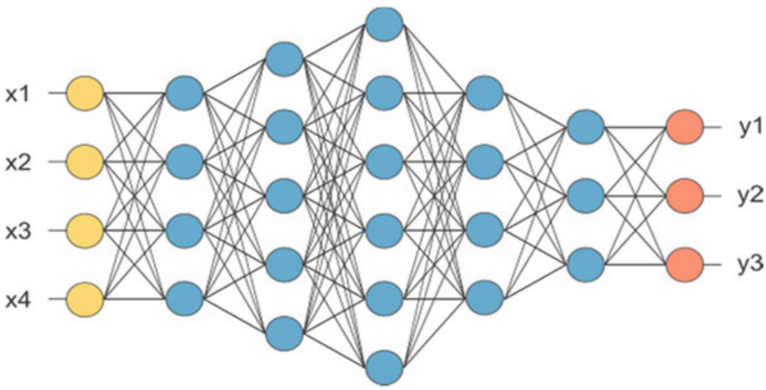


Fig. 5 Fully Connected Layers (FC)

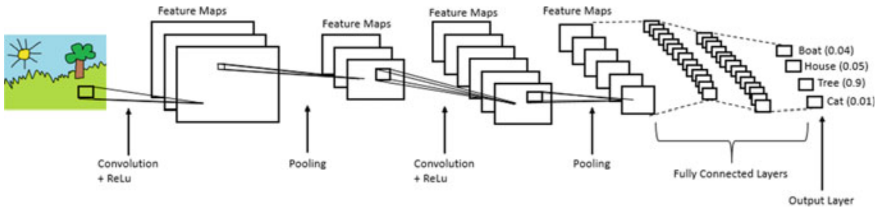


Fig. 6 CNN with multiple layers

hand [15]. It mainly contains two convolutional layers followed by two subsampling layers and a fully connected Multi-Layer Perceptron.

Few experimenters presented the use of CNN for recognition of leaf and disease classification of plant. Atabay [4] created a CNN architecture to detect plants based on images of a leaf. The one which is proposed importantly contains of five layers. After each convolutional layer a Rectified Linear Unit (ReLU) or Exponential Linear Unit (ELU) activation function is used and for each pooling layer, MaxPooling procedure is applied. The proposed one can be done on Flavia (Wu et al. 2007) and Swedish [16] datasets related to leaf consisting 32 species of a plants ¨ with 1907 samples and 15 kinds with 1125 samples respectively. The images of a single leaf in the dataset are pictures taken at uniform background. All the input images are 160 × 160 pixel grayscale images. The model achieved a 97.24% and 99.11% of classification accuracy for each dataset. The outcomes given that the architecture which is proposed for CNN-based leaf detection is almost competing with the new substantial approaches on devising leaf classifiers and features.

Reyes et al. and Camargo [17], suggested deep learning technique of hand-engineering. The designed system contains of 2 fully connected layers which follows 2 convolution layers. The CNN is trained using 1.8 million images from ILSVRC 2012 dataset and used a fine tuning policy to transfer known detection capabilities from normal areas to the particular challenge of Plant Recognition task. The dataset is combination of part of a plant or images of a plant taken both of them under a natural environment as well as in the controlled environment. They procured with an average accuracy of 0.486.

Mohanty et al. [18], applied the existing deep CNN architectures, i.e. AlexNet [19] and GoogLeNet [20] to systemize diseases of a plant. The public data set with 54,306 pictures of diseased and healthy plant leaves considered under controlled environments, the CNN was coached to detect 14 crop types and 26 diseases. The models grant accuracy of 99.35%. When tested on an images set taken at a different kinds of environment than the images used for the coaching, however, the accuracy of model dropped to 31.4%. Overall the results indicates the viability of deep CNN for plant disease classification.

The Proposed CNN Architecture Model

CNN architecture differs with the kind of the difficulty it has. The model which is proposed is a sequential model which contains of four layers of convolution,

Fig. 7 CNN with multiple layers with filter sizes

Layer type	**Filter size**	**Output size**
Conv2d	3*3	256 *256*32
Conv2d	3*3	254*254*32
Pool	8*8	31*31*32
Conv2d	3*3	31*31*32
Conv2d	3*3	29*29*32
Pool	8*8	3*3*32

maxpooling follows two of those layers and flatten function which is used to convert into a single column passed to fully connected layer from pooled feature map. Layer of Fully connected is appended by the dense layers of nearly two to the neural network.

The first convolutional layer filtrates the input picture with 32 kernels of size 3 $\times$ 3. After maxpooling is used, the end result is provided as an input for the second convolutional layer with 64 filters of length 4 $\times$ 4. Fully connected layer of 512 neurons follows the last layer of convolution has 128 filterss of length 1 $\times$ 1. The end result of this layer is provided to softmax function that makes a probability function of the four end result classes (Fig. 7).

6 Datasets

Here we considered grape disease dataset as the input. Grape disease dataset contains nearly 3000 training images and nearly 880 testing images. These images are colored. Images if they are taken from uncontrolled background and different lighting condition, then Background of training images may bias the neural network. So, these sort of pictures must be preprocessed and here the sort of dataset is preprocessed. So, There is uniform background to these images that is, only leaf figure is appeared (Tables 1 and 2). Classification of dataset is shown in Table 3.

Table 1 Training dataset

Disease name	No. of training images
Black-rot	966
Esca (Black Measles)	1154
Healthy	213
Leaf-blight	876

Table 2 Test dataset

Disease name	No. of testing images
Black-rot	210
Esca (black measles)	240
Healthy	220
Leaf-blight	210

Table 3 Disease description in grape leaves

Leaf disease name	Leaf image	Description	No. of training images	No. of testing images
Black Rot		Mainly due to fungus Guignardia bidwelli Leaves are reddish brown Circular to angular spots on upper surface of leaves	966	210
Esca (Black Measles)		Mainly due to fungi Phaeoacremonium aleophilum, Phaeomoniella chlamydospora, Fomitipora mediterranea It's like grapevine trunk disease	1154	240
Healthy		No disease. Healthy leaf	213	220
Leaf_blight (Isariopsis_Leaf_Spot)		It is a Bacterial Leaf spot	876	210

7 Proposed Work

The proposed work comprises of 6 steps given below to identify disease of grapes leaf.

- Resizing data and Split the dataset into inputs and targets
- Building the model
- Compiling the model
- Training the model
- Making predictions on new sample data.

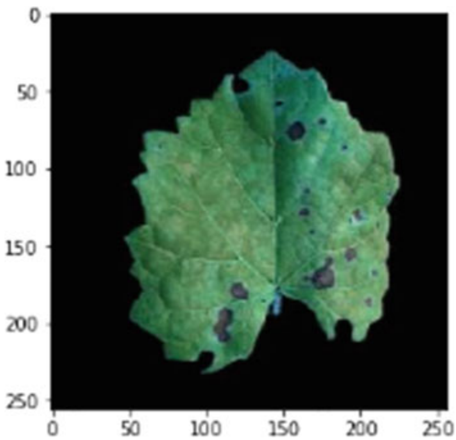
Resizing data and splitting data inputs and targets

Resizing data means converting the image size into which we need whether it may be 256 * 256 or 128 * 128 as shown in Fig. 8.

Building Model
Sequential is pretty easy way in Keras to make a model. It drives you to construct layer by layer a model. Each and every layer has weights that correlate to the layer the follows it. 'add()' method is applied by us to append layers to our model.

Compiling Model
At this stage, we applied categorical cross entropy. As the loss function, we can also utilize MSE. Either Adam Function or rmsprop we can use as learning rate. Adam function is taking long time than rmsprop. The learning rate indicates how quick the best weights for the model are being calculated. A tinier learning rate may cause to high in precise weights (up to a particular point), but the time it considers to evaluate the weights will be more. If learning rate is high, learning done is quick. Whence, there might be prune in precision. If learning rate is less, Even though there may be increase in accuracy, takes lot of time. So, Learning rate must be accordingly.

Fig. 8 Grape leaf image of 256*256 pixels



Training the Model

Right Now, Our model are going to be coached by us. To train, we'll use the 'fit()' function on our model with the next five parameters: training data (train_X), target data (train_y), validation split, the number of epochs and callbacks. The validation split will randomly split the information into use for training and testing. At the Duration of teaching, we'll be able to watch the validation loss, which offers the mean squared error of our model on the validation set. we put the validation split at 0.15, which suggests that 15% of the training data we provide within the model are visiting be forgot for testing model performance.

The number of cycles is that the quantity of times the model will cycle through the information. The high cycles we run, the high the model will get better, up to a specific point. Then point, the model will stop improving during each cycle. Additionally, the more cycles, the more the time the model will consider to execute. To watch this, we are going to use 'early stopping'.

Making Predictions on New Sample Data

In order to predict new data, we used predict() function.

predict_class = model.prdedict(img).

8 Results

Here we considered grape disease dataset as the input. Grape disease dataset contains nearly 3000 training images and nearly 880 testing images. These images are colored. Images if they are taken from uncontrolled background and different lighting condition, then background of training images may bias the neural network. These sort of pictures must be preprocessed. So, There is uniform background to these images that is only leaf figure is appeared and remaining part is black as shown in Fig. 9.

The dataset used in this paper is PlantVillage Dataset [21] and is available on Kaggle which is open source. It has approximately 55,000 well-labelled images of healthy leaves and infected leaves. This dataset contains leaf in broad level and here

Fig. 9 Grape leaf from dataset

taking two types of dataset where first one is segmented that comprises a leaf without background and other set comprises a leaf with a background.

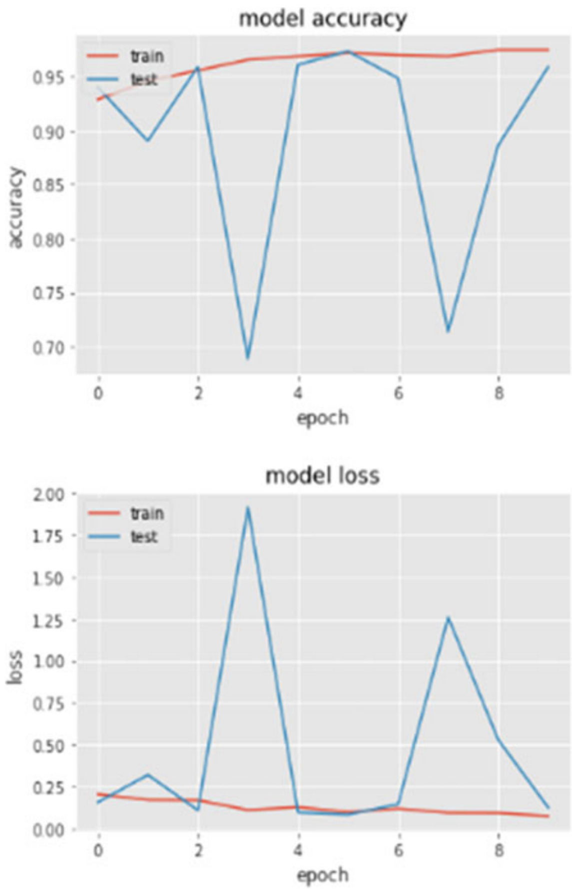
The total images in a given specific class is not same, and it varies from 423 to 1383 images. For our problem statement, we have used only Grape images which comprises of four classes i.e. black rot, Esca (Black Measles), Leaf Blight and healthy leaf images where the train-test-split data is shown in Tables 1 and 2.

Output is given as any one of four categories with the testing accuracy of 96.5.

If we consider grayscale images instead of colored images, Then There would be less classification accuracy compared to classification accuracy from colored images and from this we could understand that feature of color is one of the vital factor for classification. For various network architectures as shown in Table 2, different accuracies and different losses have been occurred which are shown in Figs. 10, 11, 12.

It can be watched from the above graphs that it is overfitting model. Overfitting happens when the model fits extremely well to the training collection. It is becoming

Fig. 10 [3 * 3, 3 * 3, 3 * 3, 3 * 3]



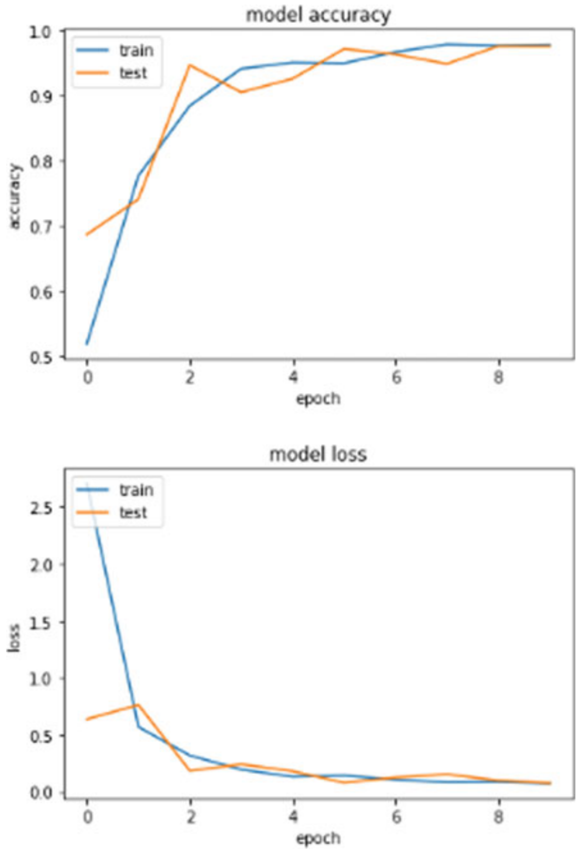


Fig. 11 [3*3, 3*3, 4*4, 4 * 4]

tough for the model to normalize to new examples which were not in the coaching set (Table 4).

9 Conclusions

The major objective of this paper is to identify and predict the Disease type of given grape leaf image. To identify and detect disease type, artificial intelligence based learning method convolutional neural network model is used. This learning method is unsupervised where target label is unknown. There are many kinds of methods in computer or automated vision plant disease identification and classification process but still this research field is lacking. Additionally, there are still no other commercial answers on the industry except those dealing with species identification of plants supported on the leaves images. So, this paper mainly identify the type of disease by using different kinds of CNN architectures, tested their accuracy and help cultivators

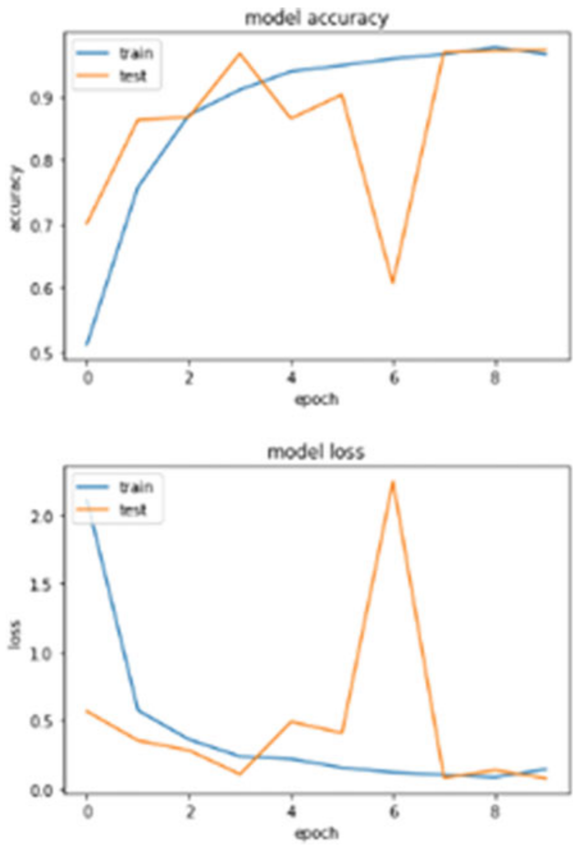


Fig. 12 [4 * 4, 4 * 4, 3 * 3, 3 * 3]

Table 4 Validation and Test accuracy of grape leaves using different CNN architecture models

Model	Network architecture	Validation accuracy	Testing accuracy
Model-I	[3 * 3, 3 * 3, 3 * 3, 3 * 3]	97.43	96.48
Model-II	[3 * 3, 3 * 3, 4 * 4, 4 * 4]	97.73	96.48
Model-III	[4 * 4, 4 * 4, 3 * 3, 3 * 3]	96.59	96.25

to detect the disease at an initial stage and may also provide solution to them what to use there to particular one so as to resolve the matter.

References

1. Amara J, Bouaziz B, Algergawy A et al (2017) A deep learning-based approach for banana leaf diseases classification. In: BTW (Workshops), pp 79–88
2. Jiang P, Chen Y, Liu B, He D, Liang C (2019) Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access 7:59069–59080
3. Geetharamani G, Pandian A (2019) Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Comput Electr Eng 76:323–338
4. Atabay HA (2016) A convolutional neural network with a new architecture applied on leaf classification. IIOAB J 7(5):226–331
5. Chaudhary P, Chaudhari AK, Cheeran AN, Godara S (2012) Color transform based approach for disease spot detection
6. Patil JK, Kumar R (2012) Feature extraction of diseased leaf images. J Signal Image Process 3(1):60
7. Reddy PR, Divya SN, Vijayalakshmi R (2015) Plant disease detection techniquetool—a theoretical approach. Int J Innov Technol Res 91–93
8. Mahlein A-K, Rumpf T, Welke P et al (2013) Development of spectral indices for detecting and identifying plant diseases. Remote Sens Environ 128:21–30
9. Xiuqing W, Haiyan W, Shifeng Y (2014) Plant disease detection based on near-field acoustic holography. Trans Chin Soc Agric Mach 2, article 43
10. Mahlein A-K, Oerke E-C, Steiner U, Dehne H-W (2012) Recent advances in sensing plant diseases for precision crop protection. Eur J Plant Pathol 133(1):197–209
11. Revathi P, Hemalatha M (2014) Identification of cotton diseases based on cross information gain deep forward neural network classifier with PSO feature selection. Int J Eng Technol 5(6):4637–4642
12. Zhou C, Gao HB, Gao L, Zhang WG (2003) Particle swarm optimization (PSO) algorithm. Appl Res Computers 12:7–11
13. Rumpf T, Mahlein A-K, Steiner U, Oerke E-C, Dehne H-W, Plümer L (2010) Early detection and classification of plant diseases with support vector machines based on hyperspectral reflectance. Computers Electron Agric 74(1):91–99
14. Al-Hiary H, Bani-Ahmad S, Reyalat M, Braik M, ALRahamneh Z (2011) Fast and accurate detection and classification of plant diseases. Mach Learn 14:5
15. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
16. Soderkvist O (2001) Computer vision classification of leaves from Swedish trees
17. Camargo A, Smith J (2009a) Image pattern classification for the identification of disease causing agents in plants. Computers Electron Agric 66(2):121–125. Camargo A, Smith J (2009b) An image-processing based algorithm to automatically identify plant disease visual symptoms. Biosyst Eng 102(1):9–21
18. Mohanty SP, Hughes DP, Salathe M (2016) Using deep learning for image-based plant disease detection. Front Plant Sci 7
19. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
20. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition
21. Ali A (2019) PlantVillage dataset, Version 1. Retrieved 22 Feb 2020. https://www.kaggle.com/xabdallahali/plantvillagedataset.