Proceedings of the Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV 2021).
IEEE Xplore Part Number: CFP21ONG-ART; 978-0-7381-1183-4

Deep Learning model for early prediction of plant disease

Rubini PE
Ph.D. Research Scholar, Department of Computer Science Engineering (VTU RC), CMR Institute of Technology, Bengaluru
Rubini.p@cmrit.ac.in

Dr.Kavitha P
Associate Professor, Department of Computer Science Engineering, CMR Institute of Technology, Bengaluru
Kavitha.p@cmrit.ac.in

2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) | 978-1-6654-1960-4/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICICV50876.2021.9388538

Abstract— **Agriculture is the main activity in many parts of the countries. Agriculture acts as a vital part of the economic system of every country. Agriculture not only provides food and raw material but also acts as a source of livelihood for farmers. Today, farmers are facing many challenges in agricultural land. This research work focuses on one of the main challenges in agricultural land i.e., disease prediction. The disease in crop plants affects agricultural production, so a model is proposed to automate a method for the prediction of disease in the plants and intimating the farmers to take appropriate action beforehand. In this work, a deep learning model is proposed that accurately classifies any leaf imagesis having a disease or not, in addition to providing a type of disease. The images of tomato plants were drawn from the Plant Village dataset and trained using pre-trained models like VGG16 and Dense Net using transfer learning and their accuracy is compared. So, the proposed system in blending with its measured interpretability and accuracy may effectively aid farmers in the early diagnosis of the diseased leaf.**

Keywords—disease prediction, CNN, pre-trained model, deep learning, agriculture

I. Introduction

Agriculture is the main area that needs to be concentrated for the growth of the economy. Farmers are held up with many issues like weather forecasting, disease prediction, phenology identification, and crop harvesting. This research work concentrates on one of the problems i.e., disease prediction. The identification of disease in plants is a very tough task, and if it's not monitored properly it leads to a decrease in the yield. To resolve the issues caused by unhealthy crops which affects agriculture demands an approach for disease identification and prediction. The need and challenge of disease detection in crops are solved by a deep learning approach. In our work, deep learning mechanism is used to classify leaf images as diseased or healthy based on the input patterns of the image. The most important work done in this research is to provide the type of disease in which a plant is affected, it is totally irrespective of binary classification of the neural network where the farmer can understand the type of disease and provide an appropriate fertilizer. The dataset used in the proposed work contains around 17000 images of tomato leaves. They are split into three groups test, train, and validation. Each of these groups contains completely different images. For example, the images contained in the training dataset will not be contained in the validation dataset. Transfer learning emphasizes saving evidence gained while resolving one problem and applying it to a different but interrelated problem. The model architecture used here is VGG-16 and Dense Net.

II. Related Work

In paper [1], Hyeon Park et al., projected a method to identify the strawberry disease. The farmer has to take a picture of a leaf or a fruit through a smartphone and once the picture is uploaded in the analysis engine system, it will find out whether it is a healthy or diseased image. If it's a diseased image it will classify the type of disease. The author used a convolution and fully connected network for identifying the diseased strawberry fruits. They used the dataset with four types of diseased images and healthy images of strawberries. Also, they generated the dataset by intentionally causing diseases to the strawberry plants. In another paper [2], Halil Darmus et al., anticipated a deep learning approach to identify a variety of diseases on the leaves of tomato plants. Two different pre-trained deep learning network architectures namely Alex Net and Squeeze Net were used. Nvidia Jetson TX1 was employed to train and validate the architecture. Plant Village dataset with tomato leaf images has been used for training. It is shown that Squeeze Net is a good architecture for the mobile deep learning classification. [3] Amara J., et al., proposed a deep learning-based technique of CNN. LeNet architecture was used to classify the banana leaf diseases. The effectiveness of this approach is that it works well with many complicated conditions.

[4] Guan Wang et al., had taken the Plant Village dataset with the apple black rot images of four severity stages and used deep convolutional neural networks to identify the complexity of the disease. The performance of both the shallow and deep network model is evaluated. The author mentioned that the best model is the deepVGG16 model which provides an overall accuracy of 90.4%. [5] Aditya Khamparia et al., proposed a hybrid approach called a Convolutional encoder network to identify crop leaf diseases. In this approach, 900 image datasets

(600: training and 300: test set) were used from Plant Village. They have considered three crops (Potato, Tomato, and Maize) and five kinds of crop disease (early blight, late blight, leaf mold, yellow leaf curl, rust disease) for their implementation. The proposed approach works well to identify crop disease given the input as a leaf image. Also, two kinds of Convolutional filters like 2X2 and 3X3 were used in the implementation part which produces an accuracy of 97.50% and 100 % respectively.

III. METHODOLOGY

The implementation is done in three stages: dataset collection, pre-processing, Training and evaluating the model.

A. Dataset Collection

The dataset was taken from the Plant Village database. It was created for use in Plant Disease Detection System, containing 4 different species of plant varieties. The proposed research work concentrated only on tomato leaf images. The dataset has two categories of images corresponding to diseased and healthy leaf images. The diseased images are labeled into 9 different classes as tomato bacterial spot, tomato early blight, tomato leaf mold, tomato Septoria leaf spot, tomato spider mites, tomato target spot, tomato yellow leaf curl virus, and tomato mosaic virus.

B. Data Pre-Processing

In data preprocessing, the raw image taken from the database had gone through preprocessing before fed into the CNN model. The images are reconstructed and normalized to establish a base size for all images and remove noise. Each image is characterized as a three-dimensional vector of P, Q, and R. P and Q represents the width and height of an image, and R represents a number of RGB channel. The images are resized to 224 x 224 pixels to be compatible with the transfer learning models. The images obtained after pre-processing is shown in Fig.1

Fig. 1. Pre-processing of images

C. CNN Model

Convolutional Neural Network (CNN) is a category of feed-forward ANN in which the shape of connection between the neurons or nodes is the same as the working principle of the visual cortex of the human brain. There are three layers in CNN: Convolution layer, Pooling layer, and Fully Connected layer. Training the model from scratch is a tedious and time-consuming process so there are many pre-trained models are like VGG 16, VGG-19, ResNet, Inception, Squeeze Net, Alex Net and DenseNet can be directly used. In this proposed work VGG 16 and DenseNet architecture were implemented and the results are analyzed with the same dataset. The dataset taken for implementation is the Plant Village dataset which contains 17000 training images and 200 testing images

D. VGG16 Architecture

The proposed work focusses on VGG16 architecture. VGG16(Visual Geometry Group) is a CNN model presented by K. Simonyan and A. Zisserman. It is trained on the ImageNet dataset, a dataset containing 1.4 Million pictures of 1000 categories. It has a total of 16 CNN layers, hence the name VGG-16, it consists of 13 CNN layers and 3 fully connected layers. The character extraction layer of VGG-16 will be frozen and the changes are made to the last three fully connected layers. VGG-16 will use the weights of the already trained images from ImageNet as a reference and classify the leaf images as diseased and not diseased. The architecture is depicted in the below diagram Fig.2

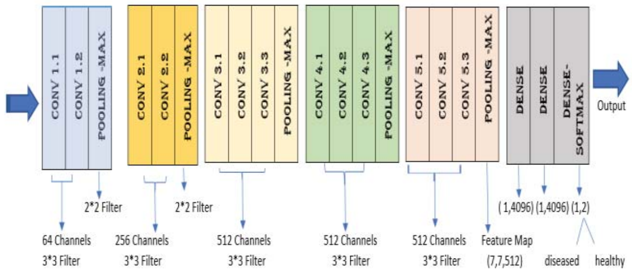

Fig. 2. VGG16 Architecture

The algorithmic step of VGG16 as described below:

1. The fixed size of 224 x 224 RGB image is forwarded as input to Conv 1.1. Then the image proceeds through a pile of convolutional (Conv.) layers, where the filters or the kernel used is of size 3*3 with a stride fixed to 1 pixel.
2. Spatial pooling is proceeded by 5 max-pooling layers over a 2×2-pixel window, with stride 2.
3. Three Fully-Connected (FC) layers trail a pile of convolutional layers:
    a. The first two dense layers have 4096 channels.
    b. The third performs 2-way classification and contains two channels (one for diseased and another for healthy).
4. The fourth and the last layer is the soft-max layer

For model creation, the VGG-16 is able to classify 1000 different labels, but here just needed 2 labels i.e., Healthy or diseased. In order to do that the last fully connected layer of the model is replaced with a new one with 2 output features instead of 1000. The two labels used here are 0 for diseased and 1 for healthy.

The loss function is used to compute the loss of the model so that the weights can be restructured to minimize the loss in the next evaluation. The choice of loss function plays an important role because it must go with the appropriate modeling problem. The equation (1) is used to compute loss function.

$$LE = -\frac{1}{N}\sum_{i=1}^{N} \log\left(\frac{p}{yi}\right) \tag{1}$$

The loss function used is Cross-Entropy which is more suitable for binary classification problems where the target values are in the set {0, 1}. Cross-entropy will estimate a value that precises the average difference between the actual and predicted probability distributions for predicting class 1. The score is minimized and a perfect cross-entropy value is 0. The number of epochs chosen here is 70. The equation (2) is used to calculate Cross-Entropy.

$$\text{Cross Entropy Loss} = -\sum_{i=1}^{2} t_i \log(p_i) = -[\,t \log(p) + (1-t)\log(1-p)] \tag{2}$$

The two phases involved are training the model and testing. For every epoch, all the training batches are iterated, compute the loss, and adjust the network weights. Then the performance over the validation set is evaluated. At the end of every epoch, the network progress (loss and accuracy) is evaluated. The accuracy will convey how many predictions were correct. Training is done only for the classifier part of the model and freeze of the feature extraction layer. For testing, an image which isn't present in any of the training and validation dataset is used as an input. Using an entirely different image would test the efficiency of the model and will check if the model can predict correctly even if a completely different image is given as an input.

E. DenseNet Architecture

In CNN, the neural network is very deep, even though there is a statement saying "deeper network provides greater accuracy", but it is very hard to train the model due to vanishing gradient. This process can be overcome by DenseNet which improves the gradient propagation by connecting all layers directly with each other. If there are L layers in the network then DenseNet will be having L(L+1)/2 connections, where a typical network with L layers will have L connections.

DenseNet consists of dense blocks, and transition blocks. In dense blocks all the layers are densely connected together. The layers in the dense block get the input from previous layer output feature maps. For example, if there are 10 layers and each layer provide the output feature map of different size, then each layer receives more supervision from the previous layer which causes the loss function so the DenseNet model fixes the feature map of each layer.

The dense block contains a batch normalization, ReLu activation and 3x3 convolution. The transition layer acts like a max pooling layer to reduce the dimensionality, it concatenates all the feature map. The architecture of DenseNet is depicted in the below diagram Fig.3

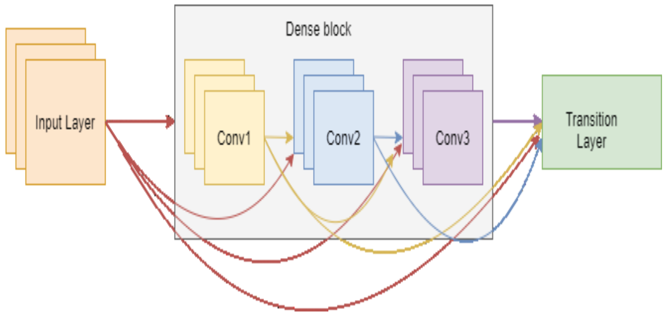

Fig. 3. DenseNet Architecture

In the above diagram Fig.3, the input layer is connected with a dense block. There are three layers in the dense block. The first Conv layer receives the feature map from the input layer. The second Conv layer receives the feature map from Conv1 and the input layer. Similarly, the third conv layer receives the feature map from Conv2, Conv,1 and the input layer. Then the output of dense block is passed to the transition layer which actually performs the task of max pooling. The transition layer receives the feature map from Conv3, Conv2, Conv1 and input layer then concatenates the feature map and reduces the dimensionality. To avoid loss, the output of each layer feature map is fixed. Each conv layer in dense block contains Batch Normalization, ReLu, and Convolution layer. The number of feature maps in each layer is called growth factor. Before the input reaches the dense block there is an initial convolution layer that produces the output size of 112x112, followed by a pooling layer which reduces the output size to 56x56.

IV. RESULT

The main intent of the proposed work is to recognize and identify whether a leaf is diseased or healthy and mention the type of disease to the farmer. Trained models are tested on the validation set using GPU. There are 14000 training images and 3000 validation images. All of them are labeled. The deep learning model was able to classify with an accuracy of 95-97 percent. The accuracy can be increased when trained with a vast number of images and by adopting pre-trained CNN models.

The output of the VGG-16 and DenseNet model is shown below in Fig.4 Upon giving a healthy leaf image as input the model predicts the output as healthy.

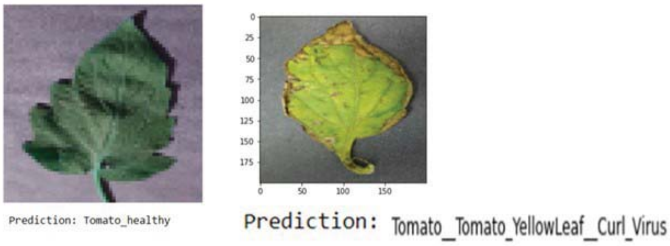

Fig. 4. Prediction result for the test image.

The comparison of accuracy between VGG 16 and DenseNet is tabulated in Table I and also a graph is drawn to find the model accuracy and loss with respect to each epoch count. It is observed that the accuracy of DenseNet is slightly higher than the VGG16 model in terms of crop disease detection.

TABLE I. COMPARISON OF VGG16 AND DENSE NET

S. No	Algorithm	Accuracy
1.	VGG16	92
2.	DenseNet	98.25

The model accuracy of DenseNet and VGG16 for the training and testing data is shown in Fig.5,6 & 7. The model also proves better accuracy for the test image given by the user. The model loss is evaluated and it shows less than 5%

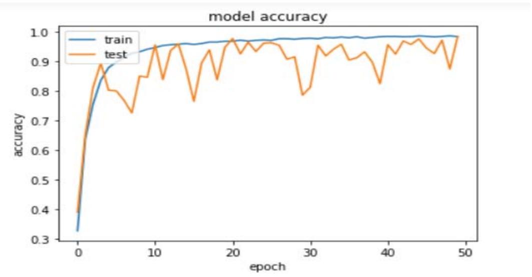

Fig. 5. DenseNet Accuracy

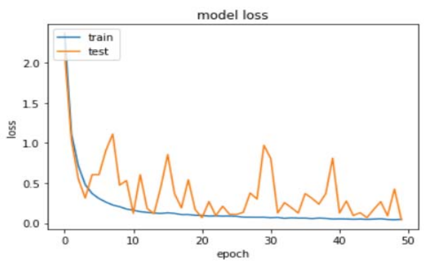

Fig 6. DenseNet Loss

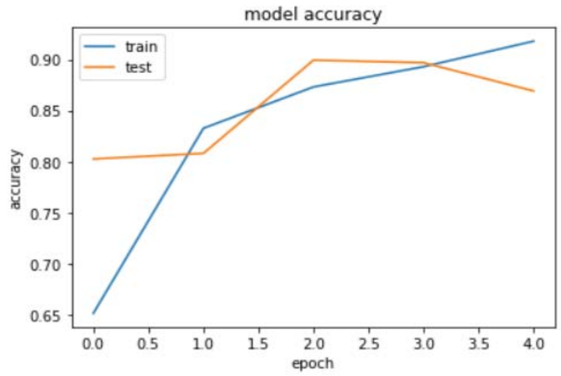

Fig 7. Accuracy of VGG16 in training and testing data

V. CONCLUSION

The research work has discussed the development of a machine learning model that identifies whether a plant is diseased or healthy. The accuracy achieved with this model is satisfactory. To improve this model, we can adopt other machine learning algorithms and try to obtain a more efficient classifier. The main drawback of VGG-16 is its fully connected nodes and is over 533MB. This makes deploying VGG a tiresome task. The DenseNet architecture proves better accuracy than VGG16 because of its more diversified features. The dataset used here includes only one type of plant and can detect the type of disease with which the leaf is affected with. In addition to this, we can include different varieties of plant species and train the model to detect different kinds of plant diseases.

REFERENCES

[1] Hyeon Park, Eun JeeSook, Se-Han Kim, “Crops Disease Diagnosing using Image-based Deep Learning Mechanism,” IEEE International Conference on Computing and Network Communications (CoCoNet), October 2018.

[2] Halil Durmus,Ece Olcay Guneu, Murvet Korco,“Disease Detection on the Leaves of the Tomato Plants by Using Deep Learning”, International Conference on Agro-geoinformatics, 2017.

[3] J.Amara, B.Bouaziz, Algergawy, “A Deep Learning-Based Approach for Banana Leaf Diseases Classification”, BTW workshop, Stuttgart, 2017, pp.79–88.

[4] Guan Wang, Yu Sun, and Jianxin Wang, “Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning”, Computational Intelligence and Neuroscience, 2017.

[5] Aditya Khamparia, Gurinder Saini, Deepak Gupta, Ashish Khanna, Shrasti Tiwari, Victor Hugo C., de Albuquerque, “Seasonal Crops Disease Prediction and Classification Using Deep Convolutional Encoder Network”, Circuits, Systems, and Signal Processing, Springer Publication,2019.