| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 7, July 2020||

Precision-Agriculture:AnImageNet - BasedMultilayer Convolution Neural Network for Leaf Disease Detection in Coffee Plant in Early-Stage System

Shubhashini pal[1], Ashritha[2], Shruti B P[3]

[1,2] ISE Department, Sri Krishna Institute of Technology, B'lore-560090, India

[3] Faculty ISE Department, Sri Krishna Institute of Technology, B'lore-560090, India

ABSTRACT: Coffee is one of the most important cash crops, known best for its delicious taste. Carefully roasted beans are manufactured to produce popular beverages that are consumed worldwide. Coffee beans are grown along the Equatorial zone called "The Bean Belt", in over 70 countries. World coffee production is estimated to have produced 155 million 60-kilogram bags, with that figure rising annually. Commercial farming of coffee beans is an excellent business and one can obtain desired profits under ideal crop practices. Fungal diseases not only influence the economic importance of the plants and its products but also abate their ecological prominence. Coffee grown in the forests of India, the world's sixth largest producer of coffee, is cultivated under thick canopies in the Western Ghats-a UNESCO World Heritage site and one of the world's most important biodiversity hotspots. In the 2016-17 season, India produced 5.5 million bags of coffee. Coffee plants, specifically the fruits, berry and the leaves are highly affected by the fungal disease named as Cercospora, Miner, Phoma, Rust. The main aim of this paper is to develop an appropriate and effective method for diagnosis of the disease and its symptoms, therefore espousing a suitable system for an early and cost-effective solution of this problem. Over the last few years, due to their higher performance capability in terms of computation and accuracy, computer vision, and deep learning methodologies have gained popularity in assorted fungal diseases classification. Therefore, for this paper, a multilayer convolutional neural network (MCNN) is proposed for the classification of the Coffee leaves infected by the fungal diseases.

KEYWORDS: Convolutional neural network, image classification, plant pathology, precision agriculture, deep learning, image processing.

I. INTRODUCTION

Secured Plants often are affected by various fungal infections and leaf diseases that hinders the growth of healthy crop and eventually reducing yield in plants, lead, buds, flowers and fruits, Plant diseases are the main cause of quantity and quality losses in agricultural production. These losses negatively impact the production in agriculture. Traditionally, farmers and plant pathologists use their eyes to detect diseases and make decisions based on their experiences, which requires a lot of layman and human efforts, which is often not accurate and sometimes biased since in the early stage many types of diseases appear to be the same and proper results might not record and the accuracy might be low sometimes. This approach leads to the unnecessary use of pesticides, which in turn results in higher production cost. Based on these pieces of evidence, the need for an accurate disease detector associated with a reliable database to help farmers is necessary, especially for the case of young and inexperienced ones. Advances in computer vision pave the way for this with the state-of-the-art Deep learning (DL) and machine learning (ML) algorithms. There is also a need for an early disease detection system to protect the crop in time. There are many previous researches conducted for this purpose. We have made use of the ''Plant Village'' dataset, a widely known dataset that is available online, with CNNs most popular model. However, the CNNs require a large amount of data for their training. In this work, we proposed approach: CNN models enhanced with ImageNet algorithm learning that uses Artificial Neural Network (ANN) with Feature Selection (FS) to solve the multi-class classification for four types of diseases, namely Cercospora, Miner, Phoma, and Rust The proposed frame work aim at increasing the models' accuracy when the data is limited.

II. MODULES

The flowchart of our proposed work is shown in Fig. 1. Inspired by Alex Net architecture, a Multilayer Convolutional Neural Network is proposed in this work for the classification of the Coffee leaves infected with the fungal disease named as Cercospora, Miner, Phoma, Rust.

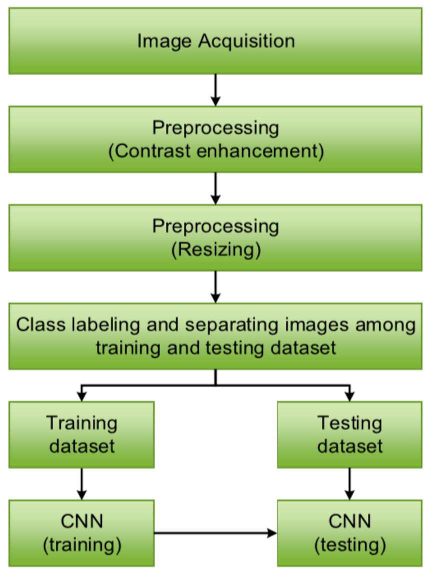

FIG 1: Work Flow of the Proposed MCNN

The procedure of proposed method is revealed by the algorithm given in Table 1.

TABLE 1: Algorithm for the proposed work

Algorithm	
1.	Acquire the real-time images of the coffee plant containing both diseased and non-diseased leaves and also, images from PlantVillage dataset.
2.	Preprocess all the images for contrast enhancement using histogram equalization method and rescaling using central square crop method.
3.	Assign the class labels to the image.
4.	Categorize the images among training and testing dataset selecting from all the class labels.
5.	Train the CNN with the help of training images.
6.	Test the CNN with the help of testing images.
7.	Validate the performance of the proposed model and compare the results with the other state-of-the-art approaches.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 7, July 2020||

Use case diagram of Proposed System.

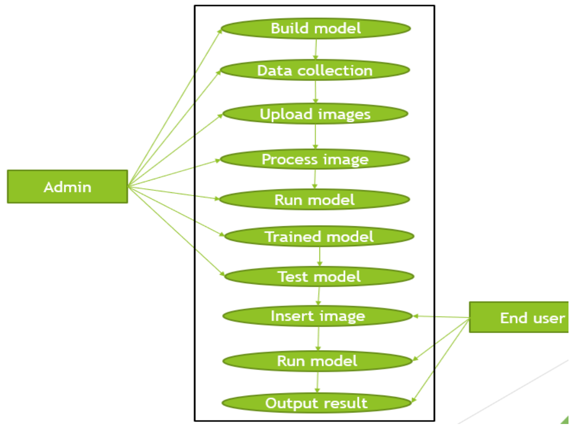

FIG 2. Use case for the proposed work.

The two participants are user and system.

User: Trains the machine and inputs the images of leaves.

System: processes the images and runs it into the model and generates the output.
The data set is sent to image processing.
The image that is processed is next put up for data image.
Generates the data set and will be given to model.
Model compares the data set with the image fed if matched then generates the output otherwise**.**

III. PROPOSED SYSTEM

The data Architecture of Proposed System.

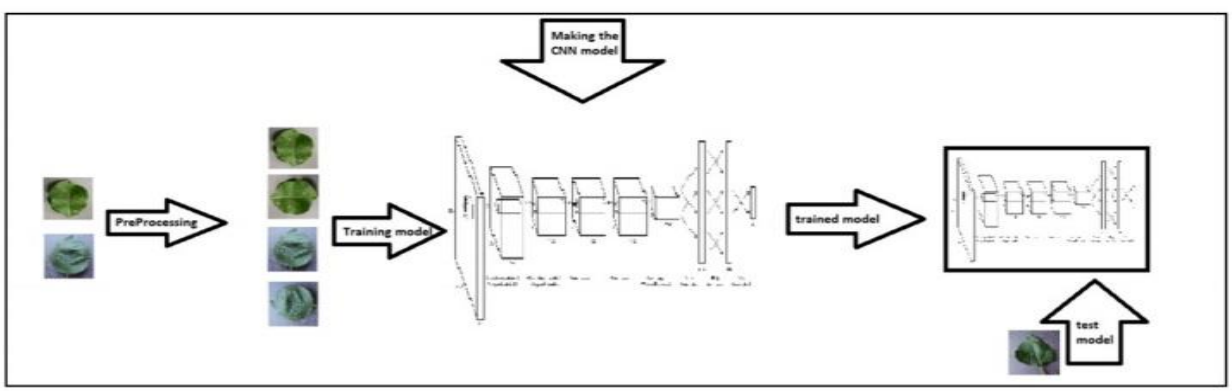

FIG 3: Architecture for the Proposed Work.

When programming a CNN, the input is a tensor with shape (number of images) x (image height) x (image width) x (image depth). Then after passing through a convolutional layer, the image becomes abstracted to a feature map, with shape (number of images) x (feature map height) x (feature map width) x (feature map channels). A convolutional layer within a neural network should have the following attributes:

Convolutional kernels defined by a width and height (hyper-parameters).
The number of input channels and output channels (hyper-parameter).
The depth of the Convolution filter (the input channels) must be equal to the number channels (depth) of the input feature map.

Convolutional layers convolve the input and pass its result to the next layer. This is similar to the response of a neuron in the visual cortex to a specific stimulus. Each convolutional neuron processes data only for its receptive field. Although fully connected feedforward neural networks can be used to learn features as well as classify data, it is not practical to apply this architecture to images. A very high number of neurons would be necessary, even in a shallow (opposite of deep) architecture, due to the very large input sizes associated with images, where each pixel is a relevant variable. For instance, a fully connected layer for a (small) image of size 100 x 100 has 10,000 weights for each neuron in the second layer. The convolution operation brings a solution to this problem as it reduces the number of free parameters, allowing the network to be deeper with fewer parameters.

For instance, regardless of image size, tiling regions of size 5 x 5, each with the same shared weights, requires only 25 learnable parameters. By using regularized weights over fewer parameters, the vanishing gradient and exploding gradient problems seen during backpropagation in traditional neural networks are avoided.

A.1 Activation Function

Activation function decides, whether a neuron should be activated or not by calculating weighted sum and further adding bias with it. The purpose of the activation function is to introduce non-linearity into the output of a neuron.

A.2 ReLU

The rectified linear activation function is a simple calculation that returns the value provided as input directly, or the value 0.0 if the input is 0.0 or less

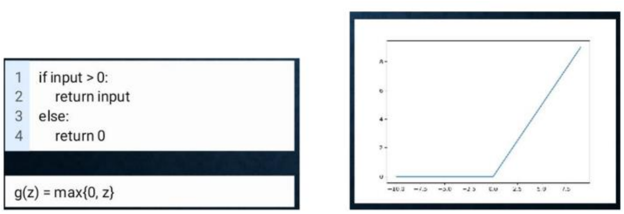

FIG 4. ReLU in MCNN

A.3 Max Pooling

Convolutional networks may include local or global pooling layers to streamline the underlying computation. Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, typically 2 x 2. Global pooling acts on all the neurons of the convolutional layer in addition, pooling may compute a max or an average. Max pooling uses the maximum value from each of a cluster of neurons at the prior layer. Average pooling uses the average value from each of a cluster of neurons at the prior layer.

A.4 Flatten

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 7, July 2020||

After finishing the previous two steps, we're supposed to have a pooled feature map by now. As the name of this step implies, we are literally going to flatten our pooled feature map into a column like in the image below.

FIG 5. Flatten in the Proposed Work.

As you see in the image above, we have multiple pooled feature maps from the previous step.

A.5 Full Connection

Adding a Fully-Connected layer is a (usually) cheap way of learning non-linear combinations of the high-level features as represented by the output of the convolutional layer. The Fully Connected layer is learning a possibly non-linear function in that space.

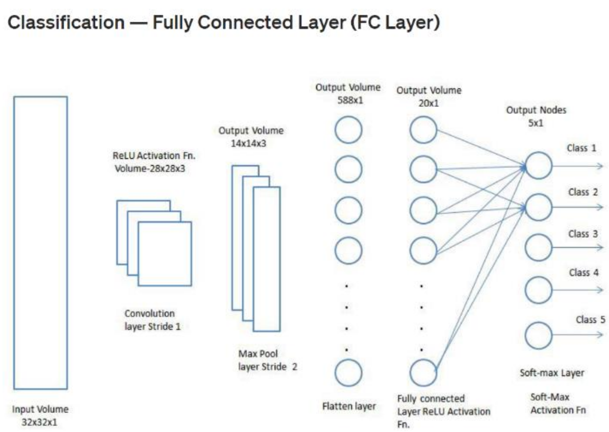

FIG 6. FC in the proposed work.

A.6 LeNet Architecture

The LeNet architecture consists of two sets of convolutional, activation, and pooling layers, followed by a fully-connected layer, activation, another fully-connected, and finally a SoftMax classifier.

The original LeNet architecture used TANH activation functions rather than RELU. The reason we use RELU here is because it tends to give much better classification accuracy due to a number of nice, desirable properties.

The LeNet architecture consists of the following layers:

INPUT => CONV => RELU => POOL => CONV => RELU => POOL => FC => RELU => FC

Figure 2: LeNet Architecture

In the first set of CONV => RELU => POOL layer sets.

Our CONV layer will learn 20 convolution filters, where each filter is of size 5 x 5. The input dimensions of this value are the same width, height, and depth as our input images, so we'll have 28 x 28 inputs with a single channel for depth (grayscale).

We'll then apply the ReLU activation function followed by 2 x 2 max-pooling in both the x and y direction with a stride of 2 (imagine a 2 x 2 sliding window that "slides" across the activation volume, taking the max operation of each region, while taking a step of 2 pixels in both the horizontal and vertical direction).

In our second set of CONV => RELU => POOL layers

we'll be learning 50 convolutional filters rather than the 20 convolutional filters as in the previous layer set. It's common to see the number of CONV filters learned increase in deeper layers of the network.

We'll apply the same ReLU activation function and 2 x 2 max pooling as the first set. Next, we come to the fully-connected layers (often called "dense" layers) of the LeNet architecture:

we take the output of the preceding MaxPooling2D layer and flatten it into a single vector, allowing us

to apply dense/fully connected layers. We know that a dense/fully connected layer is a "standard" type of layer in a network, where every node in the preceding layer connects to every node in the next layer (hence the term, "fully connected"). Our fully-connected layer will contain 500 units whichwe pass through another nonlinear ReLU activation.

Finally, we apply a SoftMax classifier (multinomial logistic regression) that will return a list of probabilities, one for each of the class labels. The class label with the largest probability will be chosen as the final classification from the network.

With this LeNet Model we use Adam optimizer. The Adam can be looked at as a combination of RMSprop and Stochastic Gradient Descent with momentum. It uses the squared gradients to scale the learning rate like RMSprop and it takes advantage of momentum by using moving average of the gradient instead of gradient itself like SGD with momentum.

IV. RESULTS

The execution incorporates the datasets are taken care of the system with 3000 generic data of agriculture includes— these incorporate crop, diseases, quality, etc. To utilize the prescient framework, machine learning algorithms requires two kinds of Data Trained Data, test data. Trained Data is the study data gathered in the time of a year, Whereas test data is the current review data.

Classification Technique

The classification method is the most important of the process as the usage of the algorithm happens here. Random forest algorithm is actualized in the process to give them through consequences of the datasets. The algorithm steps through 20% of the exam data (Random data) as the size given to the framework, and staying 80% of train data is taken.Dataset results that will be as a matrix, for example, real positive, genuine negative and so on) The anticipated Data can be the judge from the matrix itself.progressively the values of the matrix is utilized to predict the land to develop the alluring crop is given highlights of the month.

V. CONCLUSION

It is important to detect whether a leaf is healthy or diseased. Once detected, the disease needs to be identified. Four different coffee disease classes are detected in this study, i.e., Cercospora, Phoma, Miner and Rust. This paper has proposed a leaf disease detection approach that is based on Convolutional Neural Networks. The deep learning-based approach can automatically extract the discriminative features of the diseased leaf images and detect the diseases with high accuracy. Finally, a CNN model, LeNet is developed for classification stage using the extracted features. In future, we want to target multiple plants with multiple disease. this will make them capable of adopting the decision to improve yield by taking necessary precautions, preventions and correct measures to improve the health of citrus orchards. The accuracy of our results can improve using more training example. Our proposed methodology can be utilized in detection of diseases in other plants with the collaboration of respective domain experts. We intend to develop a mobile application that is based on proposed technique for direct benefits of the farmer. There is still a need of standard publicly available datasets to improve the overall performance of such systems and making the computer aided diagnosis systems more wide-ranging, which are capable of detecting and classifying different diseases more accurately.

REFERENCES

[1] X. E. Pantazi, D. Moshou and A. A. Tamouridou, "Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers," Comput. Electron. Agriculture, doi.org/10.1016/j.compag.2018.11.005

[2] Peng Jiang, Yuehan Chen, Bin Liu, Dongjian HE, and Chunquan Liang, "Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks", IEEE ACCESS vol. 7, Digital Object Identifier 10.1109/ACCESS.2019.2914929

[3] Muhammad Attique Khan, M Ikramullah Lali, Muhammad Sharif, Kashif Javed, Khursheed Aurangzeb, Syed Irtaza Haider, Abdulaziz Saud Altamrah, and Talha Akram, "An Optimized Method for Segmentation and Classification of Apple Diseases based on Strong Correlation and Genetic Algorithm based Feature Selection", IEEE ACCESS, Digital Object Identifier 10.1109/ACCESS.2018.DOI VOLUME 1,

[4] Uday Pratap Singh, Siddharth Singh Chouhan, Student Member, IEEE, Sukirty Jain, and Sanjeev Jain, Member, IEEE "Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease Uday" IEEE Access, VOLUME XX, 2017 Digital Object Identifier 10.1109/2017.Doi Number

[5] Sukhvir Kaur, Shreelekha Pandey and Shivani Goel, "Semi-automatic leaf disease detection and classification system for soybean culture," IET Image Processing Vol. 12 Iss. 6, pp. 1038-1048 doi: 10.1049/iet-ipr.2017.0822

[6] Artzai Picona, Aitor Alvarez-Gilaa, Maximiliam Seitzc,d, Amaia Ortiz-Barredob, Jone Echazarraa and Alexander Johannesd, "Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild", Computers and Electronics in Agriculture ,doi.org/10.1016/j.compag.2018.04.002

[7] Xihai Zhang, Yue Qiao, Fanfeng Meng, Chengguo Fan and Mingming Zhang, "Identification of Maize Leaf Diseases Using Improved Deep Convolutional Neural Networks," IEEE Access Digital Object Identifier 10.1109/ACCESS.2017.Doi Number VOLUME XX, 2017 2169-3536.

[8] Juncheng Maa, Keming Dua, Feixiang Zhenga, Lingxian Zhangb, Zhihong Gongc and Zhongfu Suna, "A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network", Computers and Electronics in Agriculture doi.org/10.1016/j.compag. 2018.08.048 0168-1699.

[9] Muhammad Sharif Muhammad Attique Khana, Zahid Iqbal, Muhammad Faisal Azam, M. Ikram Ullah Lali, Muhammad Younus Javed, "Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection," ELESIVER Computers and Electronics in Agriculture.