A STUDY OF AGRICULTURAL HEALTH MONITORING SYSTEM USING TOOLS OF DIGITAL IMAGE PROCESSING, CONVOLUTIONAL NEURAL NETWORKS AND DEEP LEARNING

Dr.Devireddy Srinivasa Kumar
Professor in CSE, Malineni Lakshmaiah Women's Engineering College, Guntur, AP, India

ABSTRACT

Crop diseases in agriculture are caused by various factors, including fungi, bacteria, viruses, and nematodes, and can significantly impact crop yields and quality. Understanding the types of diseases, their symptoms, and effective control methods is crucial for maintaining healthy and productive crops. For effective crop production in agriculture, early disease identification is crucial. Tomato crop quality is impacted by septoria leaf spot, late blight, bacterial spot, and yellow-curved plant leaf diseases. Using automatic techniques to classify plant diseases facilitates response after leaf disease signs are identified. This research provides a method for tomato leaf disease detection and classification based on a Convolutional Neural Network (CNN) model and the K-means clustering algorithm. several images of tomato leaves with four different disease symptoms can be found in the dataset. For automatic feature extraction and categorization, we have a CNN model. Research on plant leaf diseases actively uses information about colour. The filters in our model are applied to three channels according to RGB components. The output feature vector from the convolution portion was input into the k-means algorithm in order to train the network. The outcomes of the experiment confirm that the suggested technique accurately identifies four distinct tomato leaf disease types.

INTRODUCTION

Agriculture is the primary source of food for humans and livestock, ensuring food availability and security. Agriculture contributes to the economy through production, trade, and employment. For many, agriculture is a primary source of employment and income, particularly in rural areas. Agriculture provides raw materials for various industries, including food processing, textiles, and pharmaceuticals. Agriculture has played a significant role in shaping human societies and cultures, influencing settlement patterns and social structures. Many small-scale farmers in India practice subsistence farming, growing crops primarily for their own consumption. Larger farms and agribusinesses engage in commercial farming, producing crops for market and export. India produces a wide variety of crops, including rice, wheat, sugarcane, cotton, pulses and vegetables. Managing pests and diseases in crops is essential for maintaining productivity. Promoting sustainable agricultural practices is crucial for long-term food security and environmental sustainability. Adoption of new technologies and innovations can improve productivity and efficiency in agriculture.

Fig 1) images of different crops

The primary occupation in India is agriculture. India ranks second in the agricultural output worldwide. Here in India, farmers cultivate a great diversity of crops. Various factors such as climatic conditions, soil conditions, various disease, etc., affect the production of the crops. The existing method for plants disease detection is simply naked eye observation which requires more man labour, properly equipped laboratories, expensive devices, etc. And improper disease detection may lead to inexperienced pesticide usage that can cause development of long term resistance of the pathogens, reducing the ability of the crop to fight back. The plant disease detection can be done by observing the spot on the leaves of the affected plant. The method we are adopting to detect plant diseases is image processing using Convolution neural network (CNN).

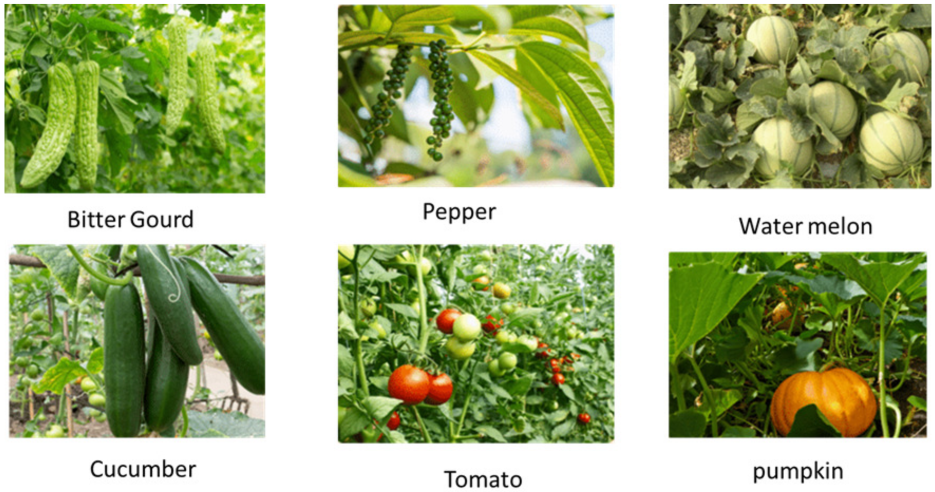

Fig 2) Images showing some of the crops

The user is to select a particular diseased region in a leaf and the cropped image is sent for processing. This paper intends to study about the prediction of the plant diseases, at an untimely phase using k-mean clustering algorithm. Specifically, we concentrate on predicting the disease. It would be useful for identifying different diseases on crops. It provides various methods used to study crop diseases/traits using image processing and data mining. In addition, the infected area and affected percentage is also measured. Back Propagation concept is used for weight adjustment of training database.

Fig 3) pics showing agricultural crop diseases

The aim of the project is to identify and classify the disease accurately from the leaf images and provide the solution for it. It shows the affected part of the leaf in percentage. The steps required in the process are pre- processing, training, identification and solution providing.

LITERATURE STUDY

Agricultural productivity [1] is something on which economy highly depends. This is the one of the reasons that disease detection in plants plays an important role in agriculture field, as having disease in plants are quite natural. If proper care is not taken in this area then it causes serious effects on plants and due to which respective product quality, quantity or productivity is affected. For instance, a disease named little leaf disease is a hazardous disease found in pine trees in United States. Detection of plant disease through some automatic technique is beneficial as it reduces a large work of monitoring in big farms of crops, and at very early stage itself it detects the symptoms of diseases i.e. when they appear on plant leaves. Several visualization techniques [2] to detect and classify the symptoms of plant diseases. Moreover, several performance metrics are used for the evaluation of these architectures/techniques. This review provides a comprehensive explanation of DL models used to visualize various plant diseases. In addition, some research gaps are identified from which to obtain greater transparency for detecting diseases in plants, even before their symptoms appear clearly.

Accurate and timely detection of diseases and pests in rice plants can help farmers in applying timely treatment on the plants [3] and thereby can reduce the economic losses substantially. Recent developments in deep learning-based convolutional neural networks (CNN) have greatly improved image classification accuracy. Being motivated by the success of CNNs in image classification, deep learning-based approaches have been developed in this paper for detecting diseases and pests from rice plant images. The contribution of this paper is twofold: (i) State-of- the-art large scale architectures such as VGG16 and InceptionV3 have been adopted and fine-tuned for detecting and recognising rice diseases and pests.

Plant leaf diseases [4] and destructive insects are a major challenge in the agriculture sector. Faster and an accurate prediction of leaf diseases in crops could help to develop an early treatment technique while considerably reducing economic losses. Modern advanced developments in Deep Learning have allowed researchers to extremely improve the performance and accuracy of object detection and recognition systems. In this paper, we proposed a deep- learning-based approach to detect leaf diseases in many different plants using images of plant leaves.

Rice disease recognition [5] is crucial in automated rice disease diagnosis systems. At present, deep convolutional neural network (CNN) is generally considered the state-of-the- art solution in image recognition. In this paper, we propose a novel rice blast recognition method based on CNN. A dataset of 2906 positive samples and 2902 negative samples is established for training and testing the CNN model. The evaluation results show that the high-level features extracted by CNN are more discriminative and effective than traditional hand-crafted features including local binary patterns histograms (LBPH) and Haar-WT (Wavelet Transform). Moreover, quantitative evaluation results indicate that CNN with Softmax and CNN with support vector machine (SVM) have similar performances, with higher accuracy, larger area under curve (AUC), and better receiver operating characteristic (ROC) curves than both LBPH plus an SVM as the classifier and Haar-WT plus an SVM as the classifier.

PROPOSED WORK

The aim of the project is to identify and classify the disease accurately from the leaf images. A colour based segmentation model is defined to segment the infected region and placing it to its relevant classes. Disease detection involves steps like image acquisition, image pre-processing, image segmentation, feature extraction and classification. It detects the affected part of the leaf by the way of percentage. We propose an enhanced k- mean clustering algorithm to predict the infected area of the leaves. It provides the accurate solution for the plant diseases.

The benefits of our work is

1. More accuracy in classification
2. Easily find out the plant disease and
3. Reduce the man power

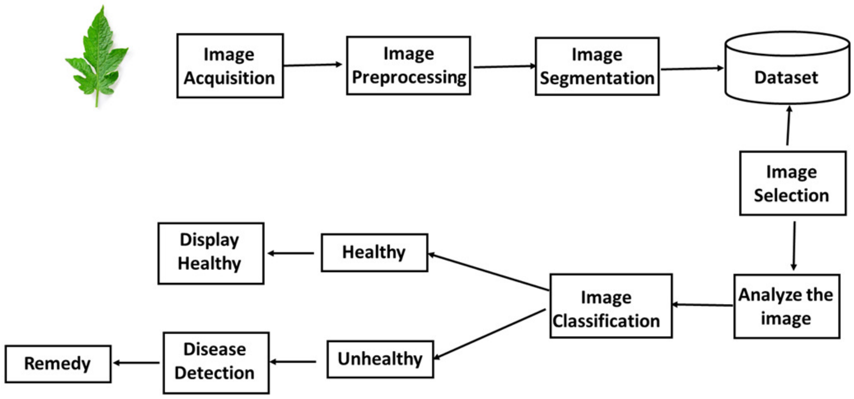

Figure 4: Architecture of Proposed Work

a) First we collect leaf images such as Pepper, Potato, Tomato.
b) The images can be pre-processed, segmented and final dataset will be prepared.
c) The image can be selected from the dataset and image analyzation is done.
d) Then it classifies the image whether it is healthy or unhealthy.
e) If the leaf healthy, shows healthy otherwise shows unhealthy and detects the disease and it provides remedy

SYSTEM MODULES: The system is designed with the following modules.

Module 1: Leaf Data Collection.

Module 2: Leaf Data Pre-processing.

Module 3: Leaf Classification.

Module 4: Leaf Disease Detection.

Module 5: Remedy.

Module 1: Leaf Data Collection

Collecting the data from public repository. In our project, we have collected three different plant leaf images. They are i) Pepper ii) Potato iii) Tomato

Module 2: Leaf Data Pre-processing

Fetching and loading the images

Transform the Image Labels using Label Binarizer

Labels are transformed into separate classes

RGB colour image to Grey scale image conversion

Images are converted numpy

Module 3: Leaf Classification

Classifies the images into two types. Thay are,

i) Healthy ii) Unhealthy

If image is healthy, shows healthy otherwise shows unhealthy

Module 4: Leaf Disease Detection

Train and Test image comparison

Disease Detection

Module 5: Remedy

Analyze the disease

Provides remedy

Let's say that an input data of size (m, n) where m is the number of training examples and n is the number of features in each example and a label vector of size (m,1). First, it initializes the weights of size (n, c) from the first c number of training samples with different labels and should be discarded from all training samples. Here, c is the number of classes. Then iterate over the remaining input data, for each training example, it updates the winning vector (weight vector with the shortest distance (Ex. Euclidean distance) from the training example).

The weight updating rule is given as follows:

if correctly classified: $w_{ij}(new) = w_{ij}(old) + alpha(t) * (x^k - w_{ij}(old))$ else

$wij(new) = wij(old) - alpha(t) * (xik - wij(old))$

where alpha is a learning rate at time t, j denotes the winning vector, i denotes the i^{th} feature of training example and k denotes the k^{th} training example from the input data. After training the CNN network, trained weights are used for classifying new examples. A new example is labelled with the class of the winning vector.

Proposed Algorithm

Step 1: Initialize reference vectors.

From a given set of training vectors, take the first "n" number of clusters training vectors and use them as weight vectors, the remaining vectors can be used for training. Assign initial weights and classifications randomly

Step 2: Calculate Euclidean distance for i=1 to n and j=1 to m,

$$D(j) = \Sigma\Sigma (x_i-W_{ij})^2$$

Find winning unit index j, where D(j) is minimum

Step 3: Update weights on the winning unit w_i using the following

conditions: if T = J then $w_i (new) = w_i (old) + \alpha[x - w_i (old)]$

if $T \neq J$ then $w_i(new) = w_i (old) - \alpha[x - w_i(old)]$

Step 4: Check for the stopping condition if false repeat the above steps.

DATA FLOW DIAGRAM: The data flow diagram shows a visual representation of the flow of data within the system which includes raw data acquisition, Pre-processing, Feature extraction, Feature fusion, Feature selection, Classification and the final output will be classified emotion.

a) First step is capturing the image.
b) Using digital image preprocessing, eliminate the noise from the image and improves the pixel value.
c) Then splitting the data into train and test.
d) In leaf classification, SVM classifier classifies the leaf into healthy and unhealthy.
e) If the leaf is healthy; it shows the status of healthy else, it shows unhealthy.
f) Then disease can be identified and provides the remedy for that.

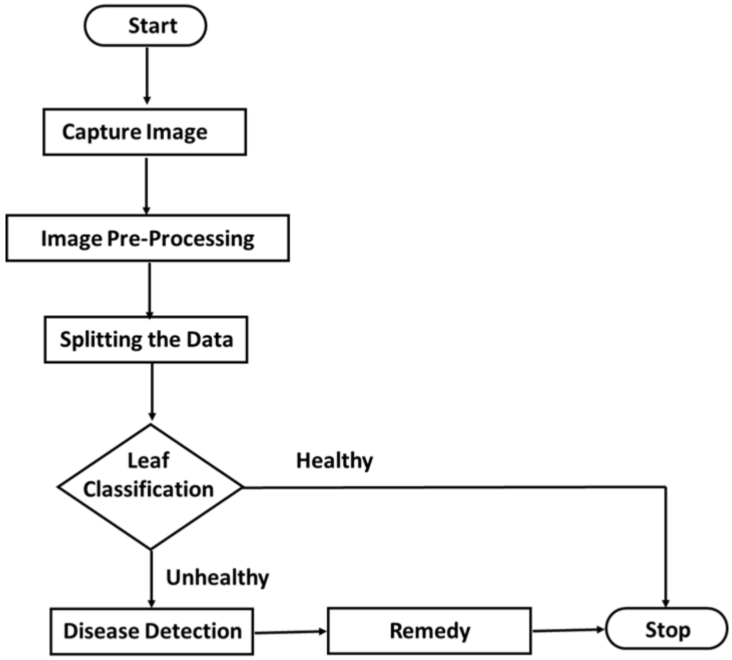

Fig 5) Data Flow Diagram

USE CASE DIAGRAM: A Use Case Diagram in Unified Modelling Language (UML) is a visual representation that illustrates the interactions between users and system. It captures the functional requirements of a system, showing how different users engage with various use cases, or specific functionalities, within the system. Use case diagrams provide a high-level overview of a system's behaviour, making them useful for stakeholders, developers, and analysts to understand how a system is intended to operate from the user's perspective, and how different processes relate to one another. They are crucial for defining system scope and requirements

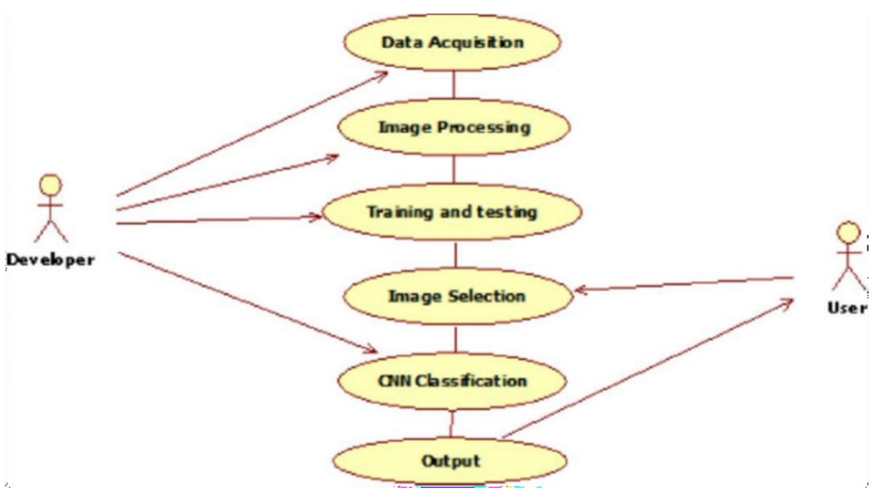

Fig 6) USECASE DIAGRAM

i The developer collects the data and pre-process the image.
ii The images can be splitted into train and test data.
iii Then the user selects the image and classifies the image by using CNN classification.
iv The results will be displayed to the user.

DEEP LEARNING: Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don't need to explicitly program everything. The concept of deep learning is not new. It has been around for a couple of years now. It's on hype nowadays because earlier, we did not have that much processing power and a lot of data. As in the last 20 years, the processing power increases exponentially, deep learning and machine learning came in the picture.

In human brain approximately 100 billion neurons all together this is a picture of an individual neuron and each neuron is connected through thousands of their neighbors. The question here is how do we recreate these neurons in a computer. So, we create an artificial structure called an artificial neural net where we have nodes or neurons.

Architectures:

1. **Deep Neural Network** – It is a neural network with a certain level of complexity (having multiple hidden layers in between input and output layers). They are capable of modeling and processing non-linear relationships.
2. **Deep Belief Network (DBN)** – It is a class of Deep Neural Network. It is multi- layer belief networks.

Steps for performing DBN:

a. Learn a layer of features from visible units using Contrastive Divergence algorithm.
b. Treat activations of previously trained features as visible units and then learn features of features.
c. Finally, the whole DBN is trained when the learning for the final hidden layer is achieved.

3. **Recurrent** (perform same task for every element of a sequence) Neural Network – Allows for parallel and sequential computation. Similar to the human brain (large feedback network of connected neurons). They are able to remember important things about the input they received and hence enables them to be more precise.

CONVOLUTIONAL NEURAL NETWORK: A Convolutional Neural Network (CNN) is a type of artificial neural network that excels at image recognition and processing. Its architecture is built upon distinct layers, typically including convolutional, pooling, activation, and fully connected layers. These layers work together to extract features, down sample data, introduce non-linearity, and ultimately classify or predict based on the input.

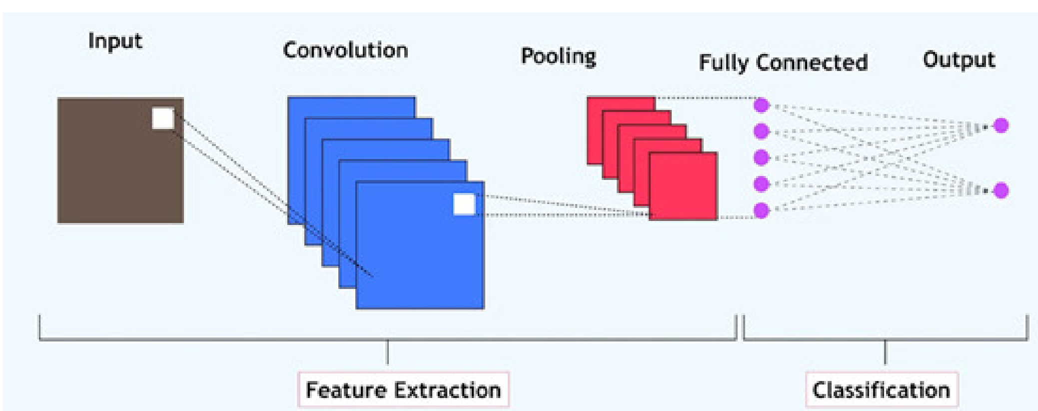

Fig 7) Architecture of CNN

When it comes to Machine Learning, Artificial Neural Networks perform really well. Artificial Neural Networks are used in various classification task like image, audio, words.

Different types of Neural Networks are used for different purposes, for example for predicting the sequence of words we use Recurrent Neural Networks more precisely an LSTM, similarly for image classification we use Convolution Neural Network. In this blog, we are going to build basic building block for CNN.

Before diving into the Convolution Neural Network, let us first revisit some concepts of Neural Network. In a regular Neural Network there are three types of layers:

1. **Input Layers**: It's the layer in which we give input to our model. The number of neurons in this layer is equal to total number of features in our data (number of pixels incase of an image).
2. **Hidden Layer**: The input from Input layer is then feed into the hidden layer. There can be many hidden layers depending upon our model and data size. Each hidden layers can have different numbers of neurons which are generally greater than the number of features. The output from each layer is computed by matrix multiplication of output of the previous layer with learnable weights of that layer and then by addition of learnable biases followed by activation function which makes the network nonlinear.
3. **Output Layer**: The output from the hidden layer is then fed into a logistic function like sigmoid or SoftMax which converts the output of each class into probability score of each class.

The data is then fed into the model and output from each layer is obtained this step is called feed forward, we then calculate the error using an error function, some common error functions are cross entropy, square loss error etc. After that, we back propagate into the model by calculating the derivatives. This step is called back propagation which basically is used to minimize the loss.

Here's the basic python code for a neural network with random inputs and two hidden layers.

Convolution Neural Networks or covnets are neural networks that share their parameters. Imagine you have an image. It can be represented as a cuboid having its length, width (dimension of the image) and height (as image generally have red, green, and blue channels).

Now let's talk about a bit of mathematics which is involved in the whole convolution process.

🎬 Convolution layers consist of a set of learnable filters (patch in the above image). Every filter has small width and height and the same depth as that of input volume (if the input layer is image input).

🎬 For example, if we have to run convolution on an image with dimension 34x34x3. Possible size of filters can be axax3, where 'a' can be 3, 5, 7, etc. but small as compared to image dimension.

🎬 During forward pass, we slide each filter across the whole input volume step by step where each step is called stride (which can have value 2 or 3 or even 4 for high dimensional images) and compute the dot product between the weights of filters and patch from input volume.

🎬 As we slide our filters we'll get a 2-D output for each filter and we'll stack them together and as a result, we'll get output volume having a depth equal to the number of filters. The network will learn all the filters.

Types of layers:

Let's take an example by running a covnets on of image of dimension 32 x 32 x 3.

Input Layer: This layer holds the raw input of image with width 32, height 32 and depth 3.

Convolution Layer: This layer computes the output volume by computing dot product between all filters and image patch. Suppose we use total 12 filters for this layer we'll get output volume of dimension 32 x 32 x 12.

Activation Function Layer: This layer will apply element wise activation function to the output of convolution layer. Some common activation functions are RELU: max(0, x), Sigmoid: 1/(1+e^-x), Tanh, Leaky RELU, etc. The volume remains unchanged hence output volume will have dimension 32 x 32 x 12.

Pool Layer: This layer is periodically inserted in the covnets and its main function is to reduce the size of volume which makes the computation fast reduces memory and also prevents from over fitting. Two common types of pooling layers are max pooling and average pooling. If we use a max pool with 2 x 2 filters and stride 2, the resultant volume will be of dimension 16x16x12.

Fully-Connected Layer: This layer is regular neural network layer which takes input from the previous layer and computes the class scores and outputs the 1-D array of size equal to the number of classes.

Dense Layer: Dense layer is the regular deeply connected neural network layer. It is most common and frequently used layer. Dense layer does the below operation on the input and return the output. Dense layers add an interesting non-linearity property thus; they can model any mathematical function. However, they are still limited in the sense that for the same input vector we get always the same output vector. They can't detect repetition in time, or produce different answers on the same input. Each neuron in a layer receives an input from all the neurons present in the previous layer—thus, they're densely connected. In other words, the dense layer is a fully connected layer, meaning all the neurons in a layer are connected to those in the next layer. Original implementation was using the Tanh function for the activation, it is now more frequent to use the ReLU, it is leading to faster training and lower probability of vanishing gradient. There are two convolutional layers based on 3x3 filters with average pooling.

EXPERIMENTAL ANALYSIS

Images of leaves from Plant Village taken from the Kaggle dataset were used in this study. Moreover, classification models were trained and tested using the dataset that was provided. All the images were divided into 10 categories, with one representing healthy options and the other nine representing unhealthier alternatives. There were nine unique harmful categories, each of which was further broken down into five distinct categories (namely- bacterial, viral, fungal, mold, and mite disease). Figure 8 displays a few examples of different types of leaves.

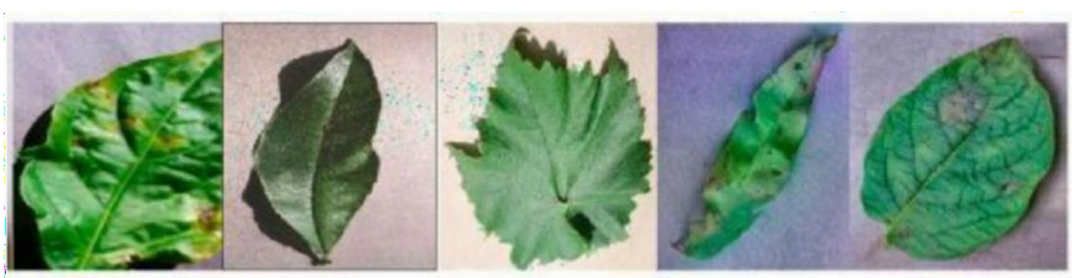

Figure 8: Sample Images from Dataset

We randomly split the plant village data into test and training sets to assess the efficacy of the proposed technique. In addition, we used the individualised leaf images for the training. The test was run on a Windows machine outfitted with an NVIDIA GM107GL Quadro and an Intel Xeon(R) Processor E3-1226 v3 running at 4 GHz x 4 with 32 GB of RAM. We used a learning rate of 0.001 and a batch Size of 32 for our hyper parameters. The Python framework and the Keras v0.1.1 package were used for all the experiments. We used to select a leaf from data set as follows for identification of disease if any.

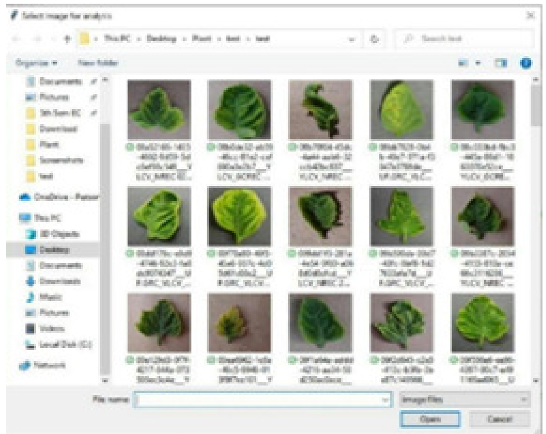

Figure 9) Selection of a leaf

The leaf that was selected from data base as input was processed and the result was displayed as shown in the figure 10. The remedy for the disease was displayed in the figure 11.

Figure 10) Identification of disease

Figure 11) Remedy for the disease

The effectiveness of a classification system can be measured in a number of ways, including accuracy, precision, recall, and F1 score. A model's accuracy is measured by how often it successfully predicts outcomes when used to make predictions. Accuracy refers to the proportion of photos that are correctly labelled by the proposed method. Recall is the fraction of Unhealthy images that were retrieved by the system. The suggested model's accuracy on the dataset is represented by its F1 score. Accuracy is measured by taking the harmonic mean of recall rates. It is a representation of the classifier's stability.

CONCLUSION

Thus a model built for the identification of disease affected plants and healthy plants is done and this proposed work is focused on the accuracy values during the real field conditions, and this work is implemented by having Ten number of plant disease classes. Overall this work is implemented from scratch and produces a better accuracy. The future work is to increase the number of classes present in the open database (Plant Village) and to modify the architecture in accordance with the dataset for achieving better accuracy. The other condition is that field condition; this means that our model has tested with the images taken from the real world conditions (land). Since the lighting conditions and background properties of the images are totally different when we take samples from the real field, there is a chance that our model to produce a very low accuracy, when comparing to the accuracy values acquired during the lab conditions. So, to overcome this impact, we have an idea of having a mixed variety of images during the training phase (heterogeneity).

FUTURE WORK

Our Plant Disease Detection deep learning model is proposed for the classification of plants affected by certain types of diseases. We had done this model from scratch so that this work will focus only on plant disease images. In future, we enhance this model for many types of plants and different types of diseases. Then we are improving the solution by providing Voice Navigation System for many languages. The deep learning model will be embedding into the android, then it transformed to the Android Application.

REFERENCES

[1] K. Elangovan , S. Nalini "Plant Disease Classification Using Image Segmentation and SVM Techniques" IJCIRV ISSN 0973-1873 Volume 13, Number 7 (2017).

[2] R.Rajmohan, M.Pajany, Smart paddy crop disease identification and management using deep convolution neural network & svm classifier, International journal of pure and applied mathematics, vol 118, no 5, pp. 255- 264, 2017.

[3] Garima Tripathi, Jagruti Save, "an image processing and neural network based approach for detection and classification of plant leaf diseases," Int. J.Comput. Eng. Technol. IJCET, vol. 6, no. 4, pp. 14–20, Apr. 2016.

[4] S. Arivazhagan, R. Newlin Shebiah, S. Ananthi, S. Vishnu Varthini, "Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features," Agric Eng Int CIGR J., vol. 15, no. 1, pp. 211–217, Mar. 2017.

[5] Rahman, Chowdhury R., Preetom S. Arko, Mohammed E. Ali, Mohammad A. Iqbal Khan, Sajid H. Apon, Farzana Nowrin, and Abu Wasif. "Identification and recognition of rice diseases and pests using convolutional neural networks." Bio systems Engineering 194 (2020):112-120.

[6] Hossain, Syed Md Minhaz, Md Monjur Morhsed Tanjil, Mohammed Abser Bin Ali, Mohammad Zihadul Islam, Md Saiful Islam, Sabrina Mobassirin, Iqbal H. Sarker, and SM Riazul Islam. "Rice Leaf Diseases Recognition Using Convolutional Neural Networks." In International Conference on Advanced Data Mining and Applications, pp. 299-314. Springer, Cham, 2020.

[7] Sethy, Prabira Kumar, Nalini Kanta Barpanda, Amiya Kumar Rath, and Santi Kumari Behera."Deep feature based rice leaf disease identification using support vector machine." Computers and Electronics in Agriculture 175 (2020): 105527.

[8] Liu, Bin, Zefeng Ding, Liangliang Tian, Dongjian He, Shuqin Li, and Hongyan Wang. "Grapeleaf disease identification using improved deep convolutional neural networks." Frontiers in Plant Science 11 (2020): 1082.

[9] SandeshRaut,, Amith Fulsunge, "Plant Disease Detection in Image Processing Using MATLAB"' IJIRSET Voll 6, Issuse 666, June 2017.

[10] Houda Orchi, Mohamed Sadik, Mohammed Khaldoun Essaid Sabir "Automation of Crop Disease Detection through Conventional Machine Learning and Deep Transfer Learning Approaches", *Agriculture* 2023, *13*(2), 352.