PLANT DISEASE CLASSIFICATION USING DEEP LEARNING

CHITRALEKHA.CH
STUDENT, DEPARTMENT OF COMPUTING TECHNOLOGIES
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
CHENNAI, INDIA
cc8269@srmist.edu.in

ROHITH REDDY. A
STUDENT, DEPARTMENT OF COMPUTING TECHNOLOGIES
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
CHENNAI, INDIA
ar8175@srmist.edu.in

DR K.R.JANSI
ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTING TECHNOLOGIES
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
CHENNAI, INDIA
jansik@srmist.edu.in

Abstract - The role of agriculture in human life is crucial. Now almost 60% of the population seems to have some involvement in agriculture. The conventional approach has the capability to identify illnesses in various crops. Farmers are not motivated to increase their agricultural productivity day by day because of the agricultural environment. Convolutional neural networks have demonstrated to be very efficient for a variety of computer vision tasks, including the detection of plant diseases and image classification. Many approaches have been put forth in recent years to address this significant deep learning field. This paper suggests a methodology for categorising plant disease symptoms using convolutional neural networks. To increase the sample size, data augmentation is applied to two plant disease datasets, which are then used to train and evaluate the model. The proposed model utilizes multiple convolution and pooling layers in a CNN architecture and is trained on the Plant Village dataset. Following training, the model is thoroughly tested to validate the results, using a 10% sample from the Plant Village dataset containing images of both healthy and diseased plants. The proposed model achieved a testing accuracy of 98.7% in detecting and recognizing the plant variety and the type of diseases the plant was infected.

***Keywords*—**PD DETECTION, DEEP LEARNING KERAS MODEL, CNN MODEL.

I. INTRODUCTION

Recent years have seen a dramatic increase in agricultural difficulties due to a combination of factors, including sudden climatic shifts and a deficiency of crop immunity. This destroys crops on a big scale, lowers civilization, and causes growers to lose money. Identification and treatment of the complaint have become extremely difficult due to the rapid expansion in the diversity of conditions and acceptable knowledge of growers. The texture and visual similarities of the leaves contribute to the classification of complaint type. Productivity is hindered by the unfavourable working conditions in the factory. The insecurity of food supplies, however, will only worsen if industrial conditions are not found in time. The operation and decision-timber of agricultural product rely heavily on pre-emptive discovery, which provides the foundation for efficient manufacturing condition avoidance and control. Recent years have highlighted the need for properly identify complaints. Innovative techniques were utilised to boost the accuracy and precision of the outcomes. Many subfields of study have developed within the classic machine literacy umbrella, including artificial neural networks, arbitrary timber, and support vector machine (SVM), fuzzy logic, knowledge-based systems, fuzzy inference, and convolutional neural networks. CNN is a neural network-based learning technology called deep learning is used. This modern technique has the capability of extracting features from images. The neural network learns how and where to extract features throughout preparation. CNN, a multi-layer feed-forward neural network, is the most recognized deep learning model. By providing a vast number and variety of images of healthy and sick plants, the Convolution neural network can be trained to recognize plant diseases, and the classification model/technique may later be utilized for predicting plant diseases using photographs of the leaves of the diseased and healthy plants.

II. LITERATURE SURVEY

General-The purpose of a literature review is to examine the major findings and research methods that have been applied to a specific issue. Secondary sources are those that describe previously published information on a topic or, more specifically, about that topic within a defined time frame. Its ultimate purpose is to bring the reader up to speed on the state of the art in the literature on a particular topic; it lays the groundwork for another objective, such as potential future research needs in the area; it comes before a research proposal, and it may be as simple as a synopsis of sources. It often follows a certain structure and incorporates aspects of both summary and synthesis. Synthesis involves rearranging and rearranging material, while summary just restates key points from the source. It could offer a fresh perspective on previously established ideas, synthesize new and established ways of thinking, or chart the development of thought in the area from its inception to the present day, including its most heated conflicts. In some cases, a literature review will provide an assessment of the available sources and recommendations for those that are most useful to the reader. Image processing has been used extensively for the finding of splint conditions throughout history, and this area of study continues to pique the interest of explorers. The use of machine learning and image processing techniques for the automatic prediction of agricultural diseases has been on the rise in recent years.

Review of Literature Survey:-

[1] Suggested a series of procedures to detect whether a leaf is sick or healthy. These steps include data cleansing, feature extraction, classifier training, and classification. Ramesh et al. suggested an RFC-based ML strategy for distinguishing between healthy and diseased papaya leaves, which is a significant step toward creating a reliable system. However, the accuracy is poor because they applied the method to more plants, which led to a larger and less reliable dataset in addition to improving disease classification with a robust descriptor and ML models.

[2] First, you gave it a picture of a leaf as an example. The image of the leaf is then broken down into its constituent colour channels, and the green pixels are removed using a mask. There is a noticeable increase in recall rate when compared to other classifier methods while using CNN. They employed spectroscopic methods to identify back then. The main disadvantage, according to this paper, is that these methods are extremely costly and can only be used by skilled professionals.

[3] They produced a successful technique for identifying and categorising YVMV in okra leaves using grayscale conversion of RGB images so that the image of the vein from each leaf can be extracted.

[4] Authors used Bhattacharya's similarity calculation to compare visually sick paddy plants to healthy ones. 100 reference photographs of healthy plants. It is helpful because it cuts down on the amount of time spent keeping an eye on crops on a vast farm, and it can see the first signs of illness right where they show up on the leaves. Not enough information to diagnose or categorize sickness. the inability to linearly divide training data.

[5] Indices Based Histogram is a technique presented by for isolating diseased areas of leaves on a plant. In the authors' opinion segmentation by slicing, approximating polygons, and shifting the image by a certain amount to get a segmentation by means of a mean shift all fall short. For Kaleem et al. to take them into consideration, images were reduced in size to 300x300 pixels, noise was eliminated, brightness was raised, and contrast was turned up to 11. By using K-means clustering for segmentation, statistical GLCM, and a support vector machine classifier are used to extract attributes helpful for identifying leaf diseases.

[6] Prediction and classification of leaf diseases using digital image processing, Innovations in Information, Communication Systems and Embedded Systems: IEEE International Conference.

[7] A classification of pomegranates was proposed in “Disease Classification Based on Back Propagation Neural Network. "Colour and texture are employed as characteristics to segment the area where the defect is most apparent. The categorization in this case was performed by a neural network classifier.

[8] Disease severity is determined by a combination of K-means clustering, GLCM texture feature extraction, and Fuzzy logic. As a classifier, artificial neural network (ANN) was utilised to determine how severely infected a leaf was.

[9] Pattern Recognition Methods for Cotton Leaf Disease Detection uses snake segmentation; in this case, Hu's moments serve as a defining feature. With an active

contour model to reduce vitality within the infected region, BPNN classifier can handle a wide variety of classification issues. On average, 85.52% of cases are correctly classified (IJCER).

[10] Ernest, James Lwasa, Godliver Owomugisha, and John proposed " Automatic Vision-Based Diagnosis of Black Sigatoka Disease and Banana Bacterial Wilt Disease." The RGB, HSV, and L*a*b colour spaces are all represented by histograms, which are then extracted and modified. To classify data, we employ area under the curve analysis and a maximum tree constructed from peak components.

III. PLANT DISEASE DETECTION SYSTEM

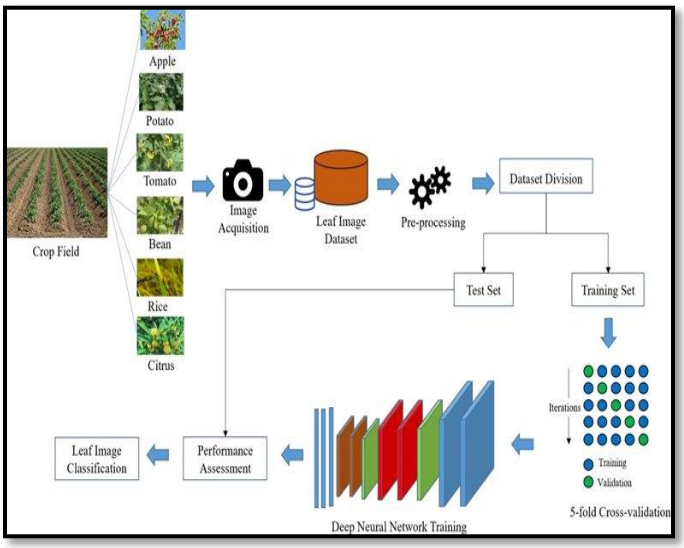


Figure 1: Architecture Diagram

As shown in figure 1, We first upload an image of a plant which needs to be tested and then the image gets processed and sent to a classifier. The classifier compares the uploaded image's characteristics to those of the photos already stored in the database; if a match is found, the corresponding illness is returned, otherwise, a message stating that the leaf is healthy is shown. In the training phase initially the user uploads an image of the leaf, from there the RGB image will be ready for the feature extraction from the leaf and among the extracted features some features will be selected, these selected features will be sent to classifier which identifies the disease and labels it with that disease name. In the testing phase, we will be uploading an image and features will be extracted and sent to the classifier. Here, the classifier will compare it with the Trained dataset and detect the plant disease.

Several techniques and processes are used to find if the leaf is afflicted or healthy. These following steps are involved, Image Acquisition, Pre-processing, Feature Extraction, Classification, and Classifier Training. Pre-processing involves numerous ways such as deblurring the blurred images, reducing the picture to same sizes to a single here we did it by dividing it with 255, achieving the uniform value called as normalization. The following step is to use a Histogram of oriented gradients to extract features from a previously processed image. A Feature descriptor known as Histogram of oriented gradients is used for object detection. The gradients in the histogram define the presence of the object and the image's contour.

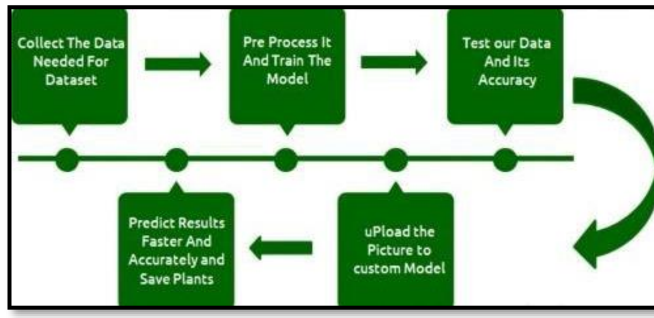


Figure 2: Step-by-step Process

There is a specific procedure that must be carried out to determine whether the plant is affected or healthy. That is done by preparing the data, extracting features from the input images while training, training a classifier, and classifying data. After being pre-processed, all the photographs have a uniformly smaller file size. Additionally, CNN is evolution of simple ANN that used to root the features of a pre-processed image. CNN's primary benefit over its precursors is that it can detect crucial traits automatically with little to no human intervention. A convolutional neural network (CNN) is a popular network model in image bracketing, target recognition, and other applications. The process is facilitated by image- processing as well and it has four stages as shown in Fig. 2

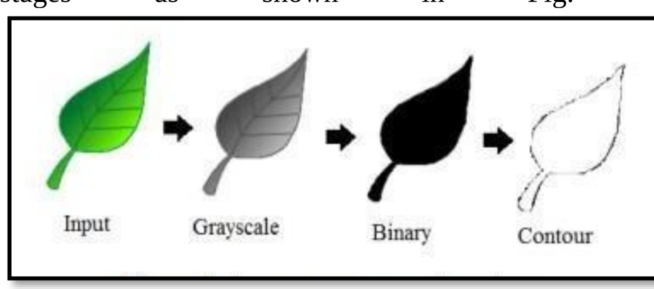


Figure 3: Image to contour conversion

3.1 CNN WORKING PROCESS

One kind of ANN is a Convolutional ANN. Image processing, classification, and segmentation are just a few applications of convolutional neural networks, which are neural networks with one or more convolutional layers.

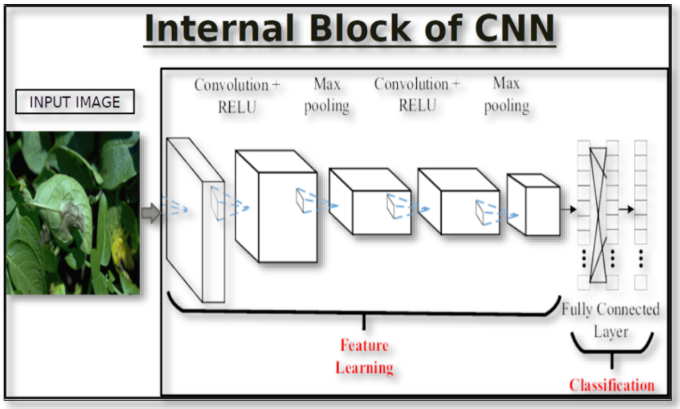


Figure 4: CNN Working Process

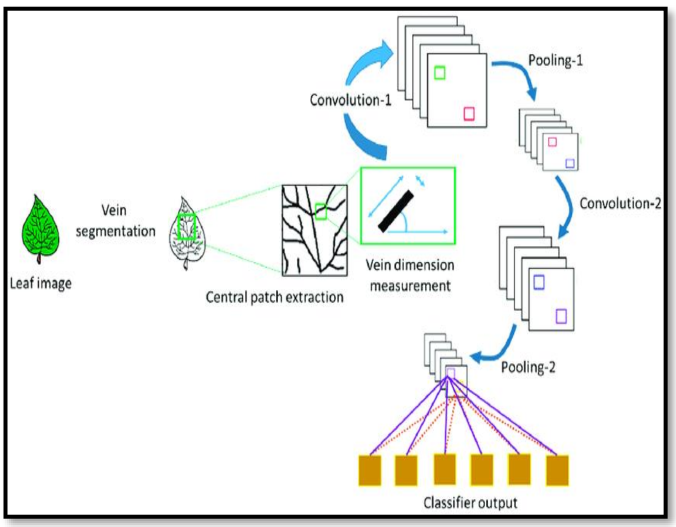


Figure 5: CNN Layers

The leaves of infected plants may be analysed by Convolutional Neural Networks (CNNs) to identify the illness. In contrast to earlier versions of ANN, which relied on provides a higher quality image result. For the reason pictures tend to have patterns of the same thing in them repeatedly. CNN's primary features are convolution and pooling. Edge detection in patterns can be accomplished with convolution, while picture compression can be achieved with pooling. Jupiter notebook is used for the training of these models.

3.2 PREPARING THE DATASET:

This dataset contains approximately 6237 images which are further classified into 4865 for training images and 1372 for testing images, which were then

```
class_names

['APPLE HEALTHY LEAVES',
 'APPLE LEAF BLOTCH',
 'APPLE ROT LEAVES',
 'APPLE SCAB LEAVES',
 'Citrus  Canker',
 'Citrus  Greening',
 'Citrus  Healthy',
 'Citrus  Melanose',
 'Citrus Black spot',
 'Citrus Leaf Disease Image',
 'Potato___Early_blight',
 'Potato___Late_blight',
 'Potato___healthy',
 'Rice Bacterial leaf blight',
 'Rice Brown spot',
 'Rice Leaf smut',
 'Tomato_Leaf_Mold',
 'Tomato_Septoria_leaf_spot',
 'Tomato_Spider_mites_Two_spotted_spider_mite',
 'Tomato__Target_Spot',
 'Tomato  Tomato YellowLeaf  Curl Virus',
```

classified into 21 classes as shown in figure 6:

Figure 6: Class Names

DATA SET:

Our dataset is in the format of .jpg in which we divided them into early blight, late blight and healthy images.
Training: 4865 images.
Testing: 1372 images
Batch Length: 100
Batch Size: 64
Epochs: 50

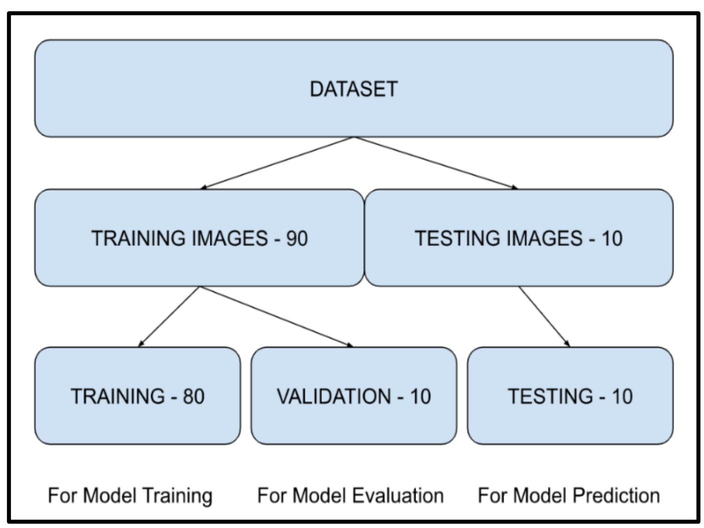


Figure 7: Dataset Segregation

IV. RESULTS AND DISCUSSION:

4.1 EXPERIMENTAL SETUP

Plant disease detection is carried by using two different data sets. The first data set has 15 categories, whereas the second has 38. Multiple photographs of each plant are included in both databases. The number of photos in the first batch is 2952. This work concludes on the village plant dataset, which includes 38 types of plants. The internet also makes it freely accessible. The three modules part of implementation are:

a. Image acquisition & pre-processing
b. Image Enhancement & Feature Extraction
c. Disease Classification

a. Image acquisition & pre-processing:

The dataset of diseased plant images has been collected from the website Kaggle and this dataset is used for the training of the model for the classification of diseases on plants. The dataset includes diseased plants of potato leaves and tomato leaves. It involves techniques such as resizing, normalization and data augmentation to improve the efficiency of the model. The sample photos of the all the leaf diseases and sick leaves from the gathered dataset as mentioned in figure 8 & 9.

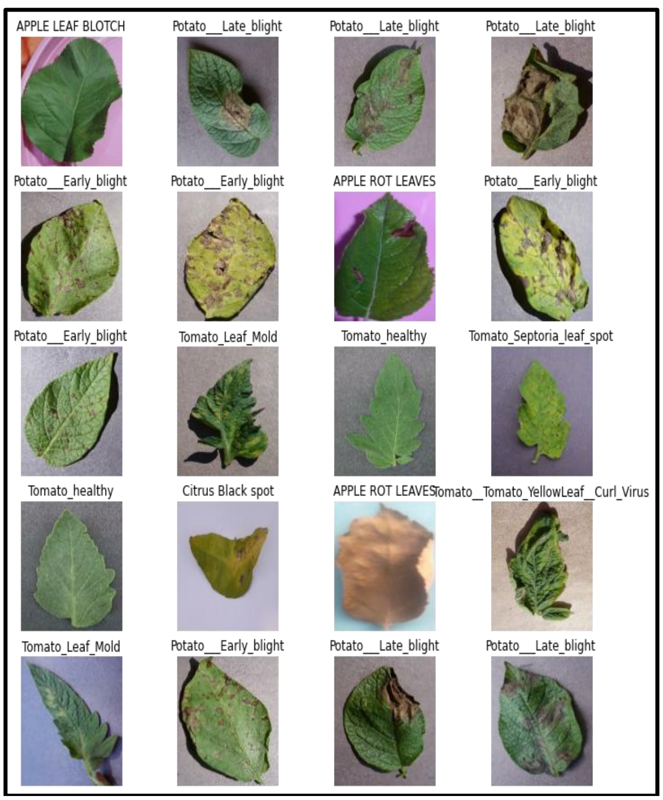


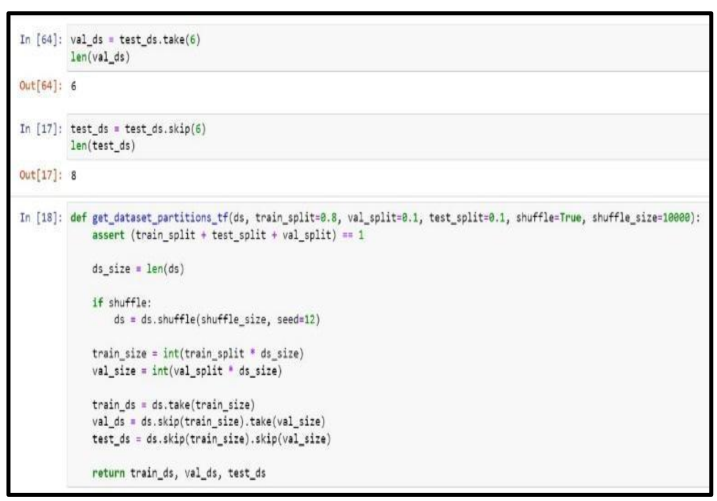
Figure 8: Diseased and healthy leaf samples

Here dataset is split in different percentages, with one is used for training and the other for testing. There is a purpose to the 80/20 split of the data collection (so that most of images get trained and can achieve best results). Eighty percent for training set, and the left over twenty for the test set. The train dataset has 54 images whereas the testing dataset has 8 and validation has 6, So total batch size is 68 in size. The model was trained on a total of 50 epochs so that it gets trained of max dataset.

Figure 9: Diseased leaf samples

b. Image Enhancement & Feature Extraction:

Data cleaning has been done and is now free of duplicate and error images from the dataset, we need to maintain the quality of the dataset as based on training dataset diseased leaves are detected. We will be training the model with the training dataset with about 80% of the images will be trained to the model.



```
In [64]: val_ds = test_ds.take(6)
         len(val_ds)
Out[64]: 6

In [17]: test_ds = test_ds.skip(6)
         len(test_ds)
Out[17]: 8

In [18]: def get_dataset_partitions_tf(ds, train_split=0.8, val_split=0.1, test_split=0.1, shuffle=True, shuffle_size=10000):
             assert (train_split + test_split + val_split) == 1

             ds_size = len(ds)

             if shuffle:
                 ds = ds.shuffle(shuffle_size, seed=12)

             train_size = int(train_split * ds_size)
             val_size = int(val_split * ds_size)

             train_ds = ds.take(train_size)
             val_ds = ds.skip(train_size).take(val_size)
             test_ds = ds.skip(train_size).skip(val_size)

             return train_ds, val_ds, test_ds
```

Figure 10: Training the Data

c. Disease Classification:

In this project eighty percent of the total dataset is used/utilised for training and the left-over twenty percent is used for validation and as well as testing. We validate it by simple formulae it is by dividing the whole dataset with the length of batch and then we divide to training, testing and validation in the above ratio. The training dataset is tested with CNN algorithm. To test the dataset necessary libraries like TensorFlow are imported.

```
In [19]: train_ds, val_ds, test_ds = get_dataset_partitions_tf(dataset)

In [20]: len(train_ds)
Out[20]: 54

In [21]: len(val_ds)
Out[21]: 6

In [22]: len(test_ds)
Out[22]: 8

In [24]: train_ds = train_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
         val_ds = val_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)
         test_ds = test_ds.cache().shuffle(1000).prefetch(buffer_size=tf.data.AUTOTUNE)

In [25]: resize_and_rescale = tf.keras.Sequential([
           layers.experimental.preprocessing.Resizing(IMAGE_SIZE, IMAGE_SIZE),
           layers.experimental.preprocessing.Rescaling(1./255),
         ])
```

Figure 11: Testing the Data dataset division

4.2 PERFORMANCE EVALUATION

The model is trained using CNN algorithm with around 6000 images consisting in the dataset which are in format of jpg images and then they are trained and tested and ran over 50 epochs because higher the number pf epochs higher the accuracy.

The Model gone through the whole process attained an accuracy between 90-93 percentage as shown in figure 12.

```
54/54 [==============================] - 711s 13s/step - loss: 0.2463 - accuracy: 0.9044 - val_loss: 0.7028 - val_accuracy: 0.8125
Epoch 36/50
54/54 [==============================] - 710s 13s/step - loss: 0.2415 - accuracy: 0.9118 - val_loss: 0.7925 - val_accuracy: 0.7812
Epoch 37/50
54/54 [==============================] - 711s 13s/step - loss: 0.2493 - accuracy: 0.9069 - val_loss: 0.7978 - val_accuracy: 0.7729
Epoch 38/50
54/54 [==============================] - 732s 14s/step - loss: 0.2259 - accuracy: 0.9176 - val_loss: 0.7075 - val_accuracy: 0.8250
Epoch 39/50
54/54 [==============================] - 711s 13s/step - loss: 0.2194 - accuracy: 0.9178 - val_loss: 0.7673 - val_accuracy: 0.8104
Epoch 40/50
54/54 [==============================] - 703s 13s/step - loss: 0.1995 - accuracy: 0.9241 - val_loss: 0.8149 - val_accuracy: 0.7979
Epoch 41/50
54/54 [==============================] - 700s 13s/step - loss: 0.1975 - accuracy: 0.9294 - val_loss: 0.8286 - val_accuracy: 0.8062
Epoch 42/50
54/54 [==============================] - 692s 13s/step - loss: 0.2234 - accuracy: 0.9130 - val_loss: 0.9355 - val_accuracy: 0.7750
Epoch 43/50
54/54 [==============================] - 695s 13s/step - loss: 0.2450 - accuracy: 0.9106 - val_loss: 0.6861 - val_accuracy: 0.7854
Epoch 44/50
54/54 [==============================] - 695s 13s/step - loss: 0.2380 - accuracy: 0.9100 - val_loss: 0.5090 - val_accuracy: 0.8354
Epoch 45/50
54/54 [==============================] - 691s 13s/step - loss: 0.2265 - accuracy: 0.9169 - val_loss: 0.5750 - val_accuracy: 0.8354
Epoch 46/50
54/54 [==============================] - 691s 13s/step - loss: 0.1896 - accuracy: 0.9303 - val_loss: 0.6924 - val_accuracy: 0.8083
Epoch 47/50
```

Figure 12: Epochs and Accuracy

After training the model, the next step is to test the CNN model with testing dataset. What happens here is that the CNN model when given a healthy or diseased leaf image and actual label as the input shows the predicted label as the output as shown in figure. 13.

```
import numpy as np
for images_batch, labels_batch in test_ds.take(1):

    first_image = images_batch[0].numpy().astype('uint8')
    first_label = labels_batch[0].numpy()

    print("first image to predict")
    plt.imshow(first_image)
    print("actual label:",class_names[first_label])

    batch_prediction = model.predict(images_batch)
    print("predicted label:",class_names[np.argmax(batch_prediction[0])])
first image to predict
actual label: Potato___Late_blight
1/1 [==============================] - 1s 854ms/step
predicted label: Potato___Late_blight
```

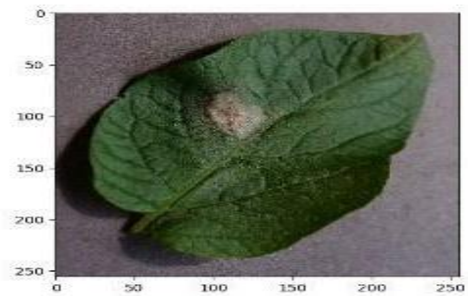


Figure 13: Final Output

The output shows an actual label which is the actual image description whether it is diseased or healthy and shows the predicted label which is the actual output that the CNN model detects. If the actual label and the predicted label shows the same result then it will display the accuracy on terms of confidence. The same is shown in Fig.14.

```
1/1 [==============================] - 0s 46ms/step
1/1 [==============================] - 0s 50ms/step
1/1 [==============================] - 0s 57ms/step
1/1 [==============================] - 0s 48ms/step
1/1 [==============================] - 0s 45ms/step
1/1 [==============================] - 0s 47ms/step
1/1 [==============================] - 0s 49ms/step
1/1 [==============================] - 0s 47ms/step
1/1 [==============================] - 0s 45ms/step
1/1 [==============================] - 0s 46ms/step
1/1 [==============================] - 0s 52ms/step
1/1 [==============================] - 0s 57ms/step
```

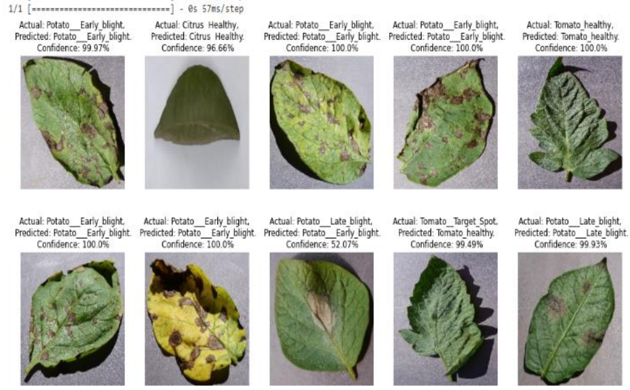


Figure 14: Final Output

Below are the graphical representations of the accuracy vs loss between training and validation for various plants that are trained and tested with CNN model.

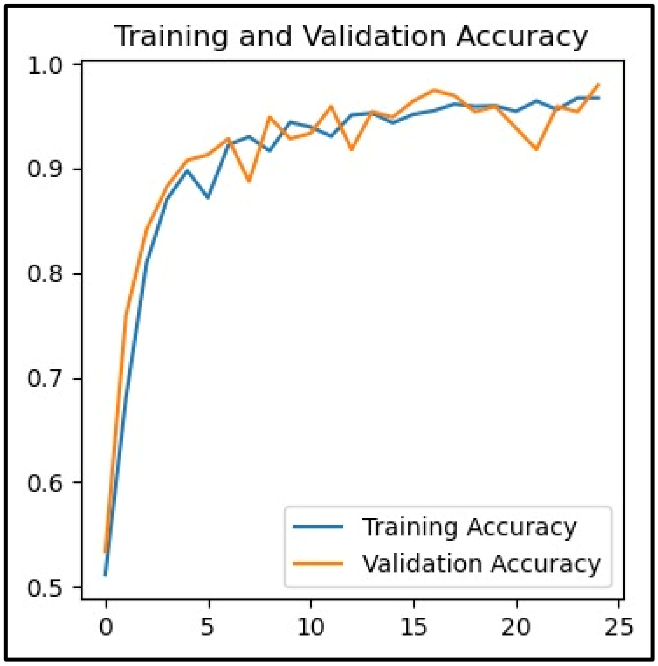


Figure 15: Training Accuracy vs Validation Accuracy for potato plant

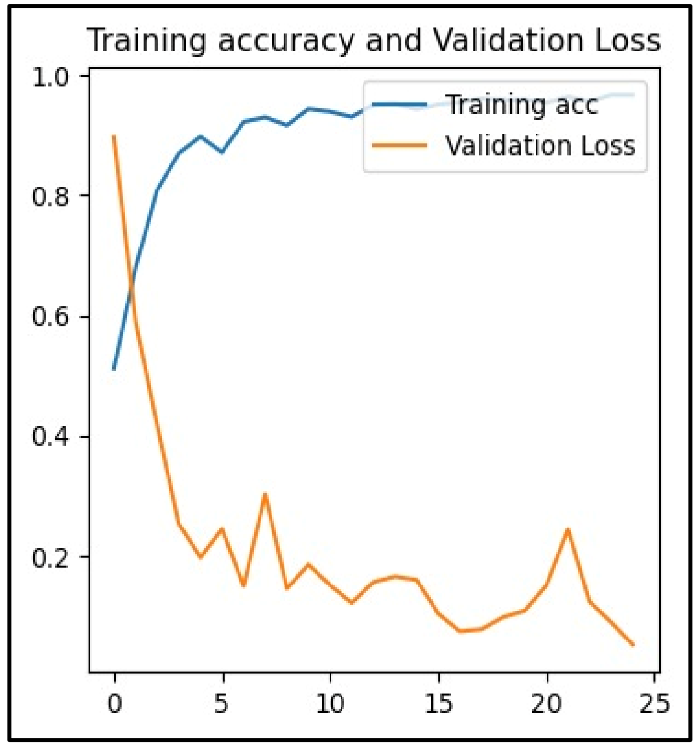


Figure 17: Training Accuracy vs Validation Loss for potato plant

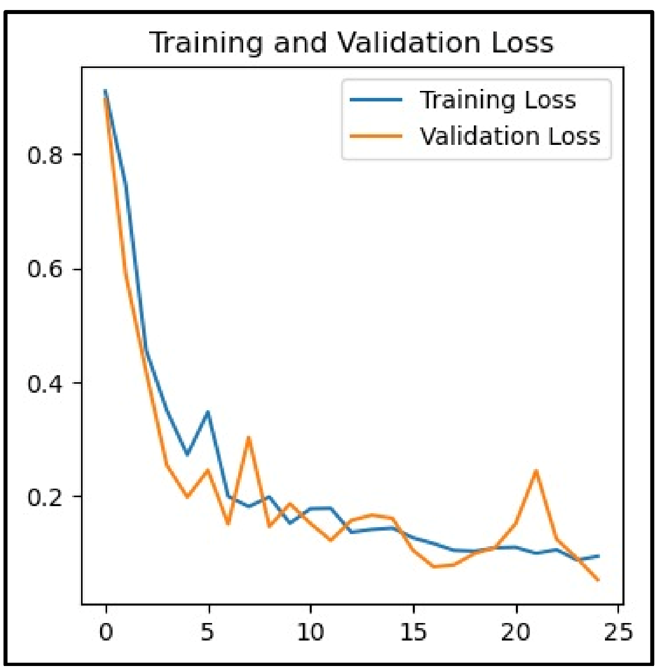


Figure 16: Training Loss vs Validation Loss for potato plant

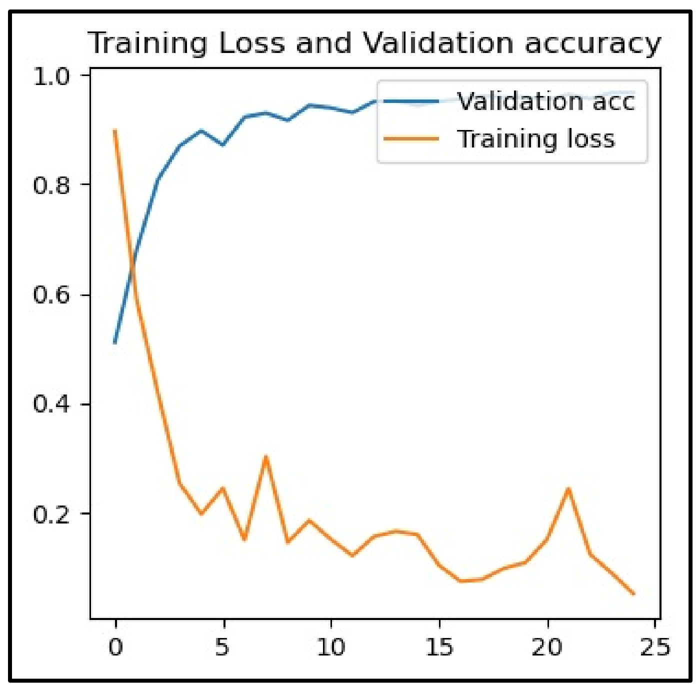


Figure 18: Training Loss vs Validation Accuracy for potato plant

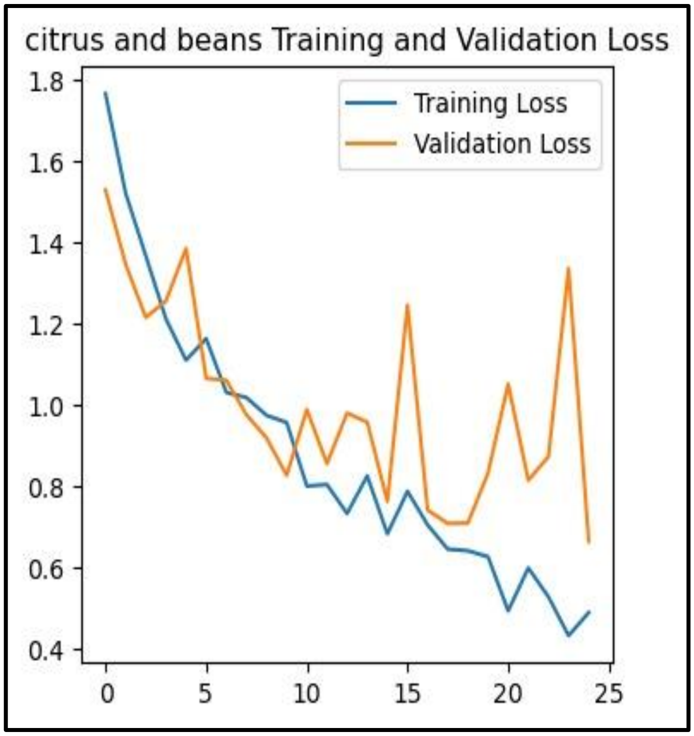


Figure 19: Training Loss vs Validation Loss for citrus and beans plant

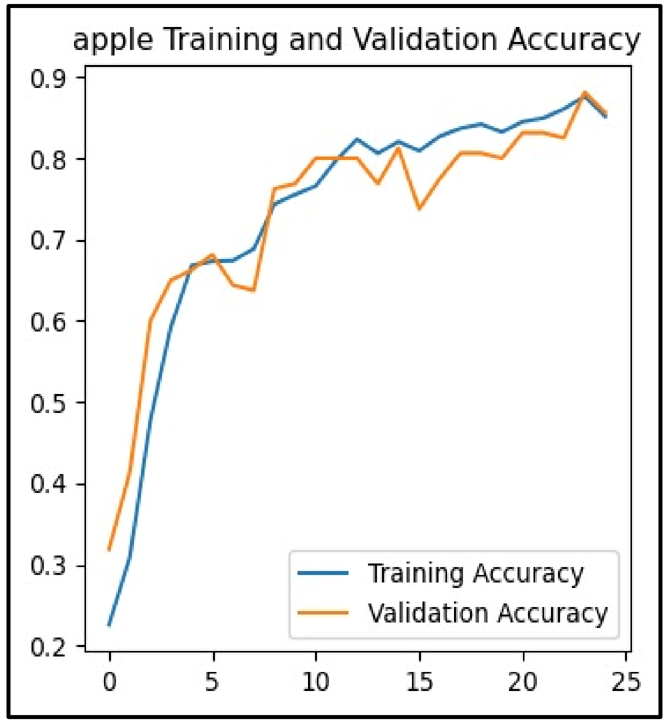


Figure 21: Training Accuracy vs Validation Accuracy for apple plant

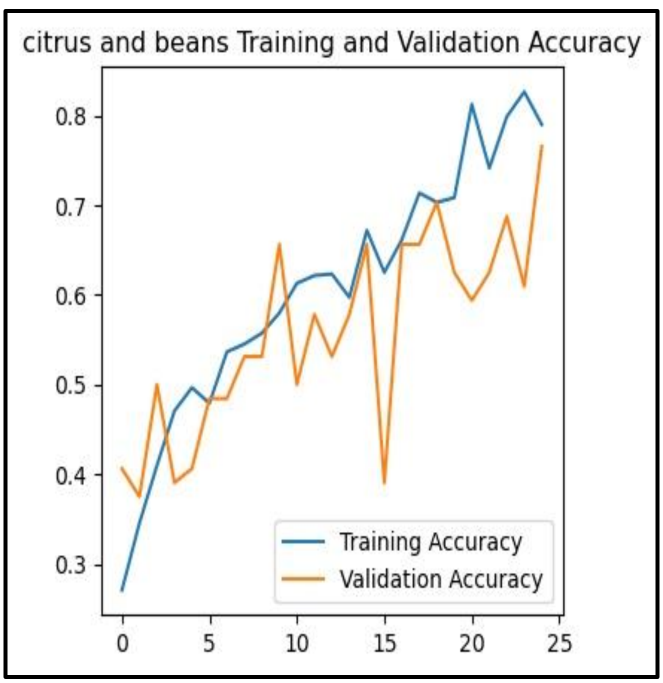


Figure 20: Training Accuracy vs Validation Accuracy for citrus and beans plants

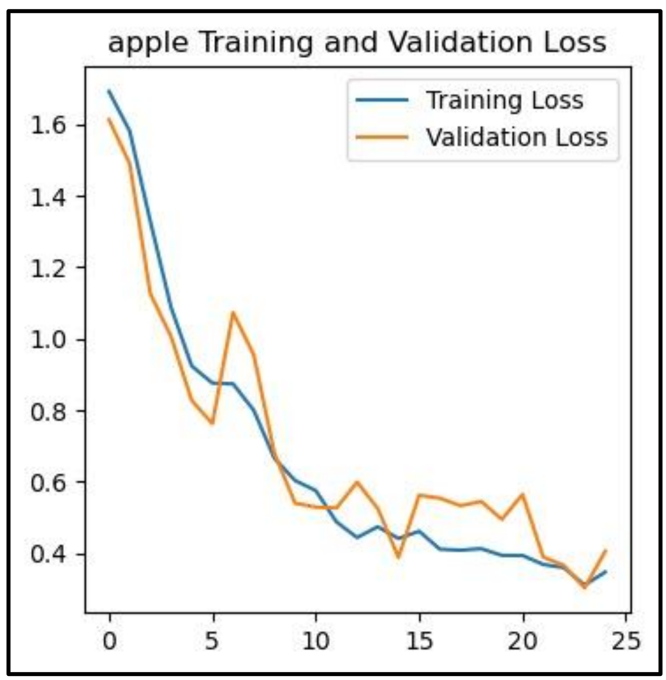


Figure 22: Training Loss vs Validation Loss for apple plant

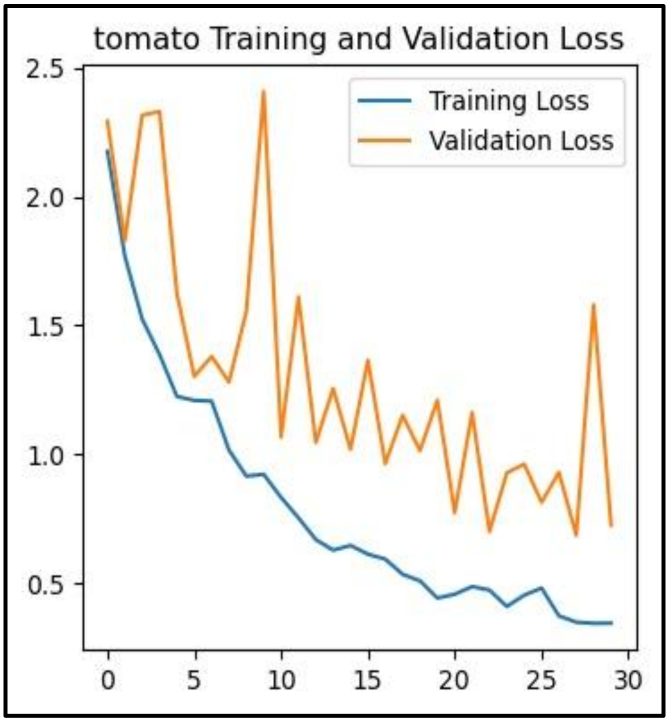


Figure 23: Training Loss vs Validation Loss for tomato plant

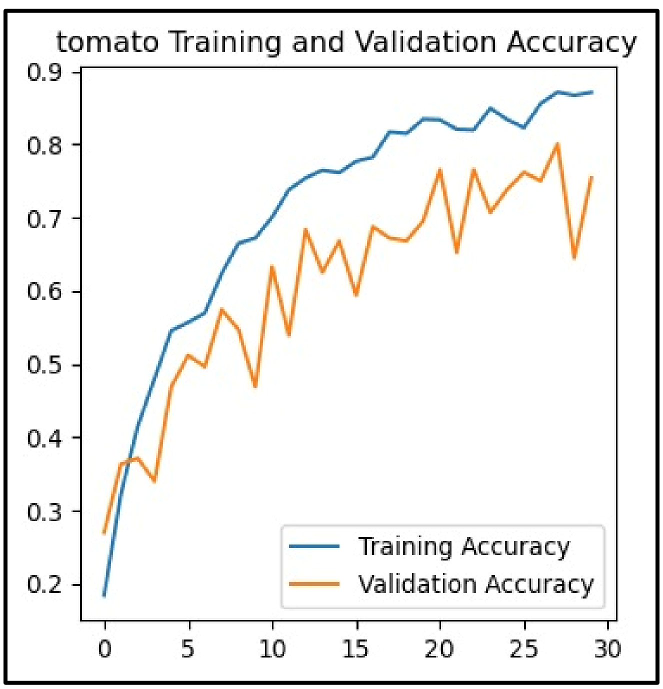


Figure 24: Training Accuracy vs Validation Accuracy for apple plant

V. CONCLUSION:

A method has been created with the goal of offering an accurate prediction of plant disease detection. The system utilises Convolutional Neural Network in the best conceivable way (CNN). Farmers will be content by the model's accurate prediction and description details of illness. The model fundamental working efficiency can be increased without disturbing other useful features for farmers. A significant improvement in the future would be a smartphone app that accurately diagnoses a disease by snapping a photo of the plant in question, allowing for greater precision and better decision-making. The suggested model analysis demonstrates that the convolutional neural networks (machine learning) classification approach achieves the required accuracy when there is a comparison with the other innovative systems. The model may be improved in the future using emulsion approaches for the emergence of major traits and tested for the different leaf samples of datasets. So that it can form latest information for innovative technologies.

REFERENCES

[1] I. Sutskever, A. Krichevsky, and G. H. E. Hinton, Image Classification with Deep Convolutional Neural Networks, in Neural Information Processing Systems, (2012).

[2] Aruna Chakraborty, D. Dutta Majumder. Used a technique called Bhattacharya's similarity calculation that can be "used for detecting paddy plant disease" International Journal of Applied Information Systems. (2014).

[3] J. S. Eun , H. Park and S. H. Kim, Disease image-based diagnosing and predicting of the crops with the deep learning mechanism, In Information and Communication Technology Convergence (ICTC), IEEE 2017 International Conference on, (2017).

[4] Zhang & Dai, "Picture of Plant Disease Segmentation Model Using Shuffle Frog Leap Algorithm and Pulse Coupled Neural Networks.", (2018).

[5] Fuentes,Yoon,J. Lee, "High-performance deep neural network-based tomato plant diseases and pests' diagnosis system with refinement filter bank", Frontiers Plant Sci., vol. 9, p. 1162. (2018).

[6] Ritika Arora, Aakanksha Rastogi, and Shanu Sharma. Disease severity detection using k-means clustering," As a classifier, artificial neural network (ANN) was utilised to determine how severely infected a leaf was.(2018).

[7] V. S. Rajpurohit and S. S. Sannakki proposed a "Classification of Pomegranate in Diseases Based on Back Propagation Neural Network", colour and texture are employed as characteristics to segment the area where the defect is most apparent. The categorization in this case was performed by a neural network classifier. (2019).

[8] P. R. Rothe and R. V. Kshirsagar. "Pattern Recognition Techniques for Detection of Cotton Leaf Disease". (2020).

[9] Kumar, Madhav & Sachin, "Coffee Plant Disease Identification Using Convolutional Neural Network", (2020).

[10] Andre, Ferreira "Detecting plant diseases on photos with convolutional neural networks", (2020).

[11] uan Tian, Shenglian Lu, Xinyu Guo and Chunjiang Zhao, "SVM Multiple Classification System for the Prediction of Wheat Leaves Diseases," Colour features are changed to HIS from RGB, via GLCM, with seven various moments used as parameters. Specifically, they made use of an SVM classifier with MCS to detect wheat plant diseases in a non-real time setting. (2020).

[12] Ernest Mwebaze, John A. Quinn, and James. They suggested " Automatic Vision-Based Diagnosis of Black Sigatoka Disease and Banana Bacterial Wilt Disease " by the RGB, HSV, and L*a*b colour spaces are all represented by histograms, which are then extracted and modified. To classify data, we employ area under the curve analysis and a maximum tree constructed from peak components. Remarkably high scores may be obtained using randomised trees in seven different classifiers, and the application benefits from the real-time data and adaptability they offer. (2020).

[13] Shima Ramesh, Vinod, "Cucumber leaf disease detection with multiclass support vector machines". IEEE International Conference on the topic Signal Processing, Wireless Communications and Networking (2021).

[14] Dhiman Mondal, Dipak Kumar Kole, "Leaves Diseases Prediction and Grading using Fuzzy Logic & Computer Vision Technology", second International Conference on the topic of Integrated Networks and Signal Processing (2021).

[15] A. Blessy, Dr. Joy Winnie Wise "Digital image processing for the prediction and segregation of leaf diseases". IEEE International Conference on the topic Innovations in Information, Embedded and Communication Systems (2021).