Plant Disease Detection using Image Processing

A PROJECT REPORT

Submitted by

BL.EN.U4CSE17077	M.Lakshman Kruthik
BL.EN.U4CSE17096	P.Sai Srujan
BL.EN.U4CSE17097	P.Nithin
BL.EN.U4CSE17118	S.Dinesh Reddy

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING, BANGALORE

AMRITA VISHWA VIDYAPEETHAM

BANGALORE 560 035

MAY-2021

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF ENGINEERING, BANGALORE, 560035

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **“Plant Disease Detection using Image Processing”** submitted by -

BL.EN.U4CSE17077	M.Lakshman Kruthik
BL.EN.U4CSE17096	P.Sai Srujan
BL.EN.U4CSE17097	P.Nithin
BL.EN.U4CSE17118	S.Dinesh Reddy

in partial fulfillment of the requirements for the award of the **Degree Bachelor of Technology** in “**COMPUTER SCIENCE AND ENGINEERING** “is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Engineering, Bangalore.

Ms. Radha D	Dr. Sriram Devanathan
Assistant Professor(SG),	Chairperson,
Computer Science and Engineering	Computer Science and Engineering

This project report was evaluated by us on 20/05/2021

EXAMINER1 EXAMINER2

ACKNOWLEDGEMENT

The satisfaction that accompanies successful completion of any task would be incomplete without mention of people who made it possible, and whose constant encouragement and guidance have been a source of inspiration throughout the course of this project work.

We offer our sincere pranams at the lotus feet of **"AMMA", MATA AMRITANANDAMAYI DEVI** who showered her blessing upon us throughout the course of this project work.

We owe our gratitude to **Br. Viswamrita Chaitanya Swamiji**, Director, Amrita School of Engineering, Bangalore.

We thank **Dr. Sriram Devanathan**, Principal and Chairperson-CSE, Amrita School of Engineering, Bangalore for his support and inspiration.

It is a great pleasure to express our gratitude and indebtedness to our project guide **Ms. Radha D.** Assistant Professor, Department of Computer Science and Engineering, Amrita School of Engineering, Bangalore for her valuable guidance, encouragement, moral support and affection throughout the project work.

We would like to express our gratitude to project panel members for their suggestions, encouragement and moral support during the process of project work and all faculty members for their academic support. Finally, we are forever grateful to our parents, who have loved, supported and encouraged us in all our endeavors.

ABSTRACT

Plant diseases are the most occurring diseases for the plant that causes destruction of the whole crop. The early detection of these kinds of diseases can improve the crop life expectancy. The project is to develop a software program that detects these kinds of harmful diseases in the early stage itself in order to reduce the damage that is caused to the crops. This is done using image processing techniques. The software built in this project acts as a primary check of plant disease detection with the main aim of decreasing false negatives. The software built will take the input as the image of the leaves of the plant and classify the leaf as tend to be diseased or not, using computer aided techniques like machine learning. Dataset used in this process consists of a good ratio of health and sick leaves. Convolution neural network is used to classify the images and has an accuracy of 93.2%. Implementation is done using adam optimizer and 25 epochs for the accuracy check. The dataset is splitted on the basis of having more sick images in the testing set so that it is easy for the classification and it is used 70% on training and 30% on testing approximately. Out of a total 3900 images, around 2700 are used for training and 1200 are used for testing. The training of the model using those 2700 images is done on the basis of all possible angles and lighting conditions of a single leaf image rather than training more individual leaf images. The dataset consists of 39 classes of the diseased leaf which are occurring regularly in the plant of different species. While doing so, it is easy for testing phase and also the classification becomes easier as the number of images used for training are more and also the accuracy can be increased and the prediction of the output will be optimal.

TABLE OF CONTENTS

LIST OF FIGURES page no.

CHAPTER 1 - INTRODUCTION

1.1) Introduction:

India is an agricultural country in which around 70 to 80 percent of the economy comes from the agriculture sector but the 10s and 100s of crops are damaged due to some kind of dangerous diseases that occur regularly and action for those diseases are not taken in the early stage so that it becomes too late for the precaution.

Fig. 1.1 Diseased crops

From to the Fig. 1.1 The crops are diseased in this way by having the spots and the color changes on the leaves.

The leaf infection is identified in many ways but the popular way is to spot the different kinds of spots and color changes in the leaves. Getting affected by these diseases is common in plants due to irregular use of fertilizers or overuse of the fertilisers and environmental conditions as well.

Plant diseases are the most widely occuring in urban and rural parts of India. These types of diseases are increasing drastically in China and East Asia as well and have the smallest growth in West Asia and North Africa.

85 percent of the plant diseases are occurring due to fungal infection on the plants. However other diseases are caused due to viral and bacterial infection as well. Certain nematodes also cause harmful types of plant diseases

From the past 10 to 12 years image processing has been an emerging technique that detects disease based on leaves of the plant. This information tells about the disease whether it is present in the tree or not, so that precaution is taken in the early stage.

The main benefit of this methodology is that doctor supervision is not needed in the initial stage, where plant disease can be detected using computer-aided diagnostic (CAD) in which we compare a healthy leaf and a leaf with disease, so that it can be detected in the early stage.

1.2) Motivation and Need:

In the detection of these diseases of the plant, non destructive techniques are used for the analysis of the leaves because those are the delicate parts of the plants. The evaluation of the agricultural harvesting is dynamic. The main important visual property that to be concentrated on, is the color and the texture of the plant.

The process of the identification will be too slow if the expert is hired to check the plant and analyze it to get to know whether the plant tends to be diseased or not. Sometimes this procedure may have errors in the analysis. If the better quality methods are recorded into the automatic system then the effort of the labor will be negligible by using appropriate techniques.

There are two main characteristics of the machine learning methods of plant disease detection and those two characteristics must be achieved in order to have good and efficient output, those characteristics are speed and accuracy. This will prove useful techniques for farmers to alert them in the early stage and can reduce the crop damage.

The main advantage of this system is to detect the disease in the early stage and reduce the labor effort with the use of computer aided approaches. By acquiring the speed and the accuracy the system becomes more efficient and the output will be more accurate.

CHAPTER 2 – LITERATURE REVIEW

[2.1]Identification of Plant Disease using Image Processing Technique(2019)

In this paper the plant disease is detected in five stages: image acquisition,image preprocessing, image segmentation, feature extraction and classification. For the identification they have used K-means clustering and SVM techniques for classification purposes.

By using this idea the illness identification is done to all types of the leaves properly so that the user can rectify the disease in the proper manner with the least damage. The authors introduced FCM clustering technique for segmentation and SVM technique for classification.

These techniques are used for reducing the analysing time. GLCM methodology is used for the feature extraction purpose and at last random forest and SVM are used to classify them and both are compared in order to give the proper high accuracy between those two algorithms.

The clustering and the classification techniques are used for the detection and identification of the spots on the leaves but the problem comes here, there is no measure for the color change of the leaves which may lead to false negatives if the color change and the spots are in the same color. This can cause high change in the output for prediction.

Types of diseases used in this paper are shown in Fig. 2.1.1. There are different categories that come under bacterial, viral and fungal diseases.

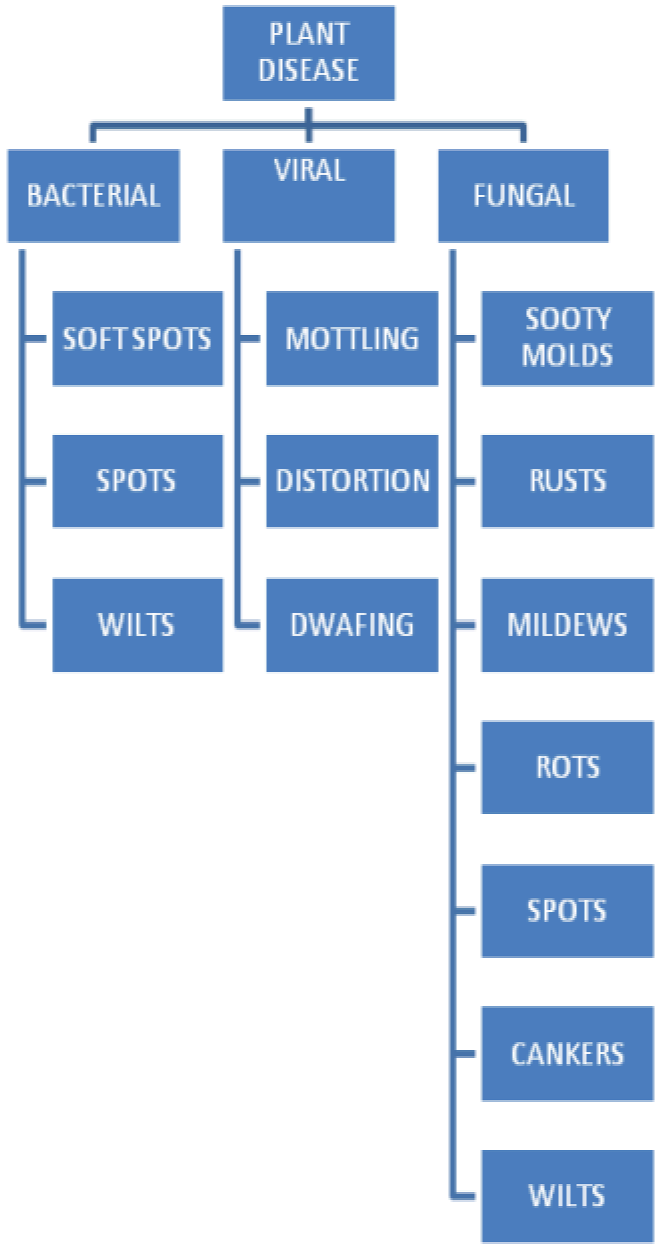


Fig. 2.1.1 Types of plant diseases

Steps involved in the system model are :

1. Image Acquisition
2. Image Preprocessing
3. Image Segmentation
4. Feature Extraction
5. Classification

Output of the paper is shown in Fig. 2.1.2. There is color change in the and spots are also present on the leaf which is detected as the diseased leaf.

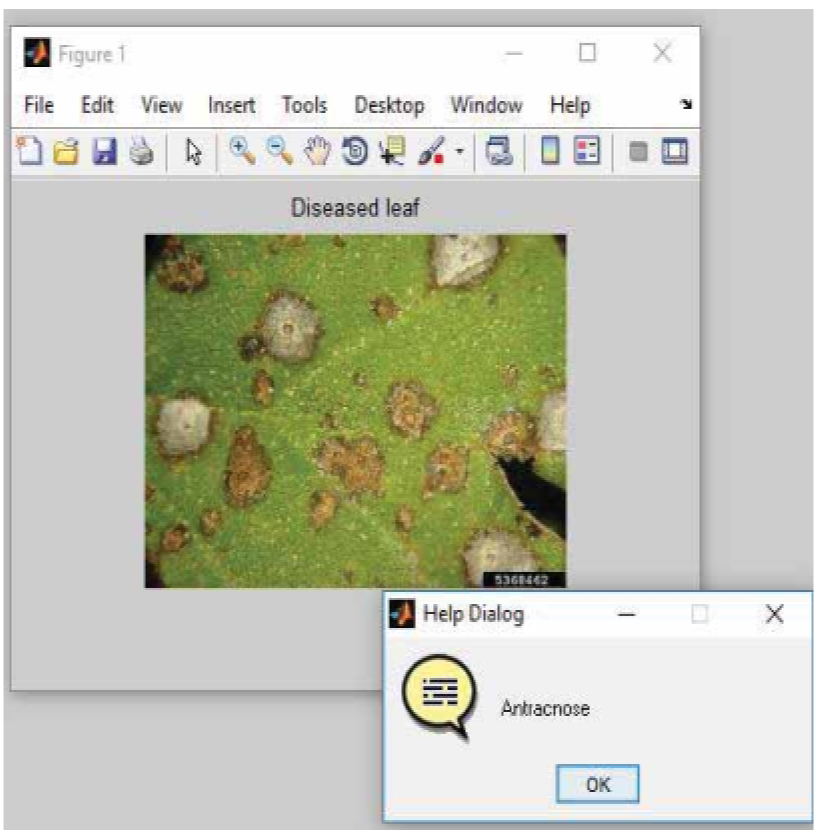


Fig. 2.1.2 Output

[2.2]Image based Plant Disease Detection inPomegranate Plant for Bacterial Blight(2019)

A methodology to build a mechanism with the help of computer aided techniques which can detect plant disease in the early stage and can reduce the issues caused in the next stages of disease which can have high chances of crop death.

In this paper the RGB images of the leaves are converted into the grayscale images in order to remove the noise in the images. To extract the region of interest segmentation techniques are used and consists of three phases in preprocessing that are resizing, filtering, enhancement.

For the feature extraction GLCM methodology is used for the analysis of the image. The main motive of the feature extraction is to extract all the features such as color, texture, morphological and color coherence.

For the classification Ada-Boost Ensemble algorithm is used which has an accuracy of 93.2 percent. Bagging and Boosting techniques are used in order to reduce the error rate and have high accuracy.

In this process the focus is on the single plant or the crop but if more than one type of the species are present in one crop this cannot be used because the disease may vary from one crop to another crop this methodology is very much helpful only if we have the same crop throughout the farm.

Pomegranate bacterial blight in Fig. 2.2.1 which is the most deadly disease in pomegranate plants that reduces the yield and quality of the fruit.

Fig. 2.2.1 Bacterial Blight

Steps involved in this paper are shown in Fig. 2.2.2 which consists of 5 stages

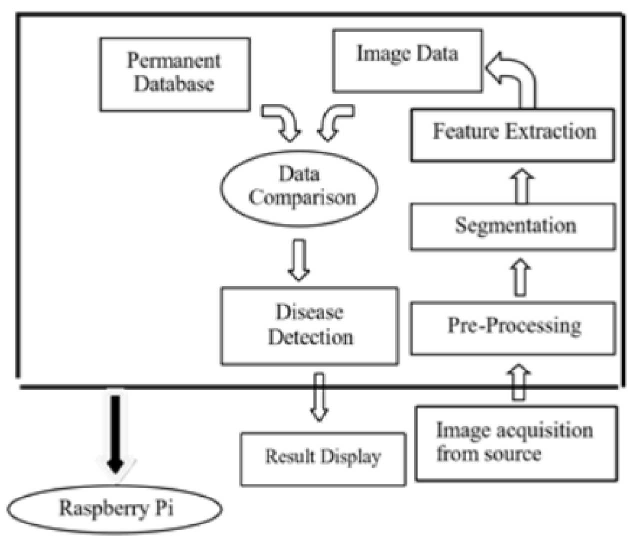
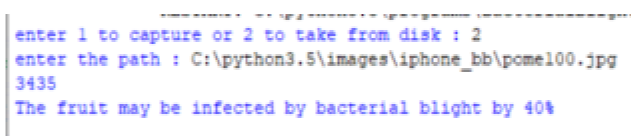


Fig. 2.2.2 Block Diagram

Segmentation in Fig. 2.2.3 is done in order to locate the objects and the boundaries of the image that is lines and the curves

Fig. 2.2.3 Image Segmentation

Output of this paper shown in Fig. 2.2.4 which states that the pomegranate plant is infected by the blight or not.



```
enter 1 to capture or 2 to take from disk : 2
enter the path : C:\python3.5\images\iphone_bb\pome100.jpg
3435
The fruit may be infected by bacterial blight by 40%
```

Fig. 2.2.4 Output

[2.3]Rice Leaf Blast Disease Detection Using MultiLevel Colour Image Thresholding (2018)

In this paper there are five stages for the identification of the disease : image acquisition, preprocessing, segmentation, feature extraction, classification.Image acquisition is the central advance in the image processing techniques. They have used the FLIR SC 620 camera for 640 x 480 pixels.

For preprocessing the gaussian filtering technique is used in order to remove the noise and acquired an accuracy of 90 percent. The image preprocessing includes inversion, color conversion, filtering and detail enhancement of the images.

For feature extraction the GLCM methodology is used which takes all the features such as color, texture, spots and other kinds of features. For classification purposes the Naive Bayes classifier is used which acquired an accuracy of 91 percent. The SVM classifier and random forest classifier are compared in order to give the high accuracy.

In this work the system has been developed using SVM classifier and obtained the accuracy of 90 percent but this methodology is not useful for the disease of the similar type but characteristics are different if we have the similar disease with different characteristics then the use of inappropriate use of the fertilizers can cause the huge damage for the crop.

System model in Fig. 2.3.1 is the five stage model in which the image is taken then pre processed accordingly and segmentation is done to get the boundaries and get the output.

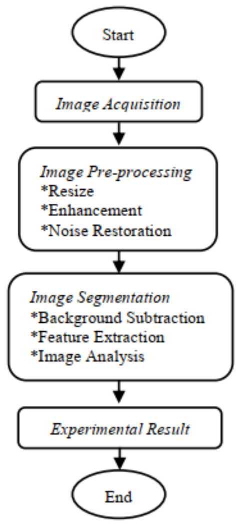


Fig. 2.3.1 Flow Chart

Image Enhancement is the process of enhancing the image by improving the quality and the content of the image

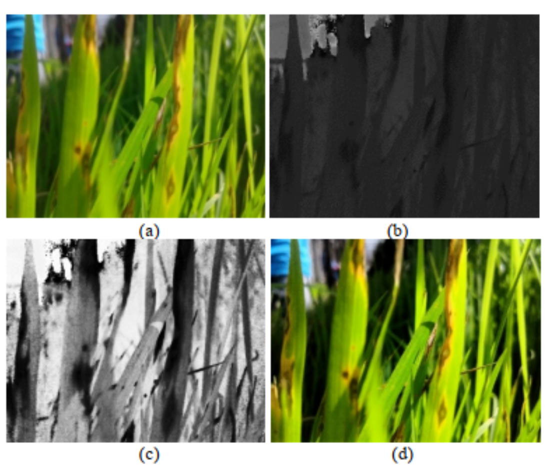


Fig. 2.3.2 Image Enhancement

Canny Edge Detection with Gaussian Filter is used in Fig. 2.3.3 which is a segmentation technique for smoothning of the image.

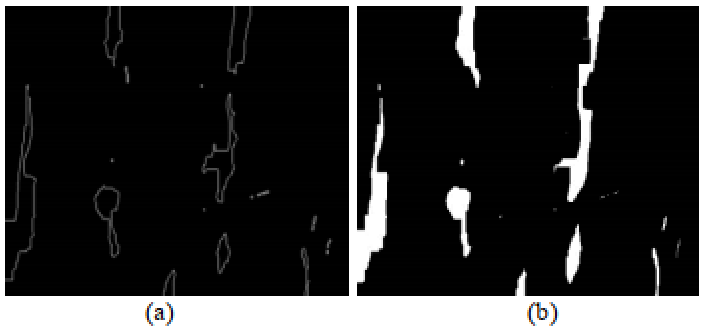


Fig. 2.3.3 Canny Edge Detection

Output of this paper is shown in Fig. 2.3.5. The leaves are detected based on the color change and the spots on the leaves

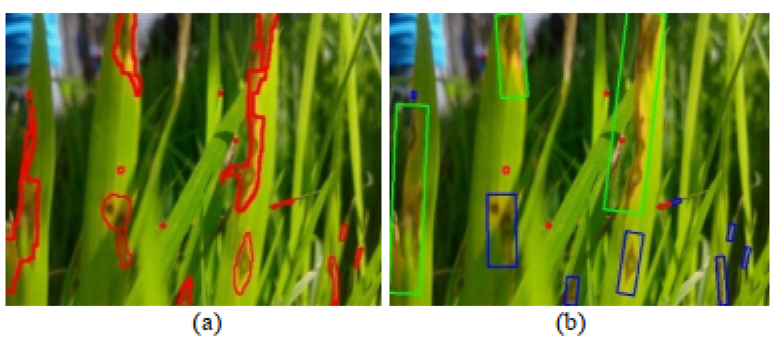


Fig. 2.3.4 Output

[2.4]Plant Disease Detection Using Image Processing (2015)

In this paper there are 5 stages for the identification of the diseased plant; those are image acquisition,preprocessing,segmentation,feature extraction and classification. The dataset is from the research database which consists of around 2500 images of the leaves.

In image preprocessing, image clipping, image cropping and image smoothing are included in order to remove the noise from the images by using the smoothing filter and image enhancement is done for increasing the contrast of the image.

For image segmentation K Means clustering is used for converting the RGB image of the plant into HSI model because there will be high chance of getting good accuracy in the classification.

For the feature extraction part all the features such as color, texture, morphology.The texture of the image in the sense the hardness and the roughness of the image are taken into consideration. The gray scale images are used for better classification accuracy. The SVM classifiers are for the classification of the image whether the plant tends to be diseased or not.

In this paper the system produced an accuracy of 80 percent which is pretty high but by using much more noise in the images this can lead to a high chance of false positives and false negatives which is not at all good for the image processing projects.

Backpropagation network in Fig. 2.4.1 is a neural network which is used for tuning of weights of the network based on the error rates.

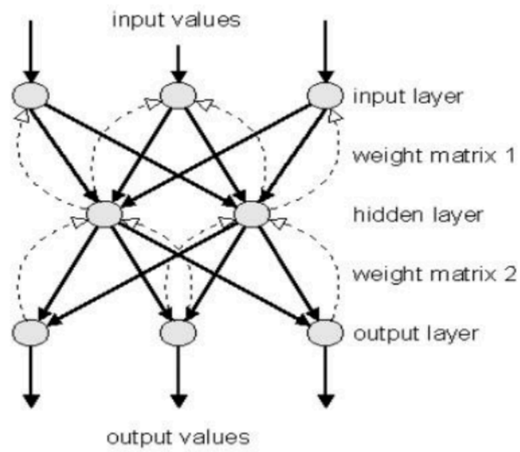


Fig. 2.4.1 Backpropagation Neural Network

Fig. 2.4.2 is the system model and the dataflow of the paper which gives the stages involved in the paper.

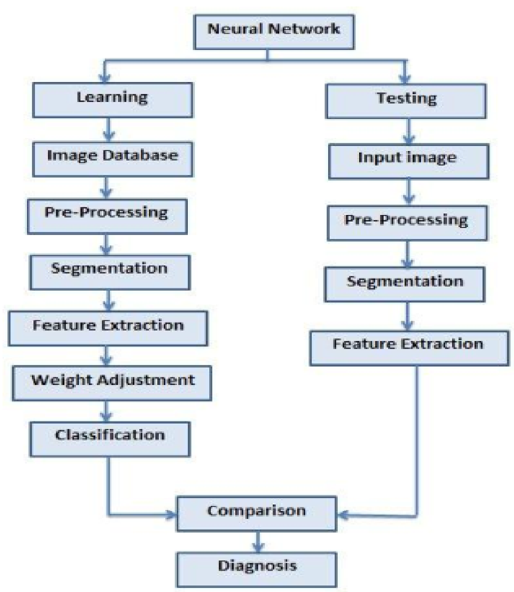


Fig. 2.4.2 Dataflow

CHAPTER 3 – REQUIREMENTS

3.1) Software Requirements :

- Platform - Google colabs. Colab allows anybody to write and execute arbitrary python code through the browser using the notebook by colabs.

- Editor - Jupyter Notebook. Notebook which is used for creating the python files.

- Python Libraries - Matplotlib , CV2 , OS , Skimage , Pandas ,Numpy, PIL.

- Keras and Tensorflow – open source libraries used for machine learning applications.These are used for defining the neural networks.

- Convolution Neural Network - This Network is used for creating the model for the application.

CHAPTER 4 - DESIGN

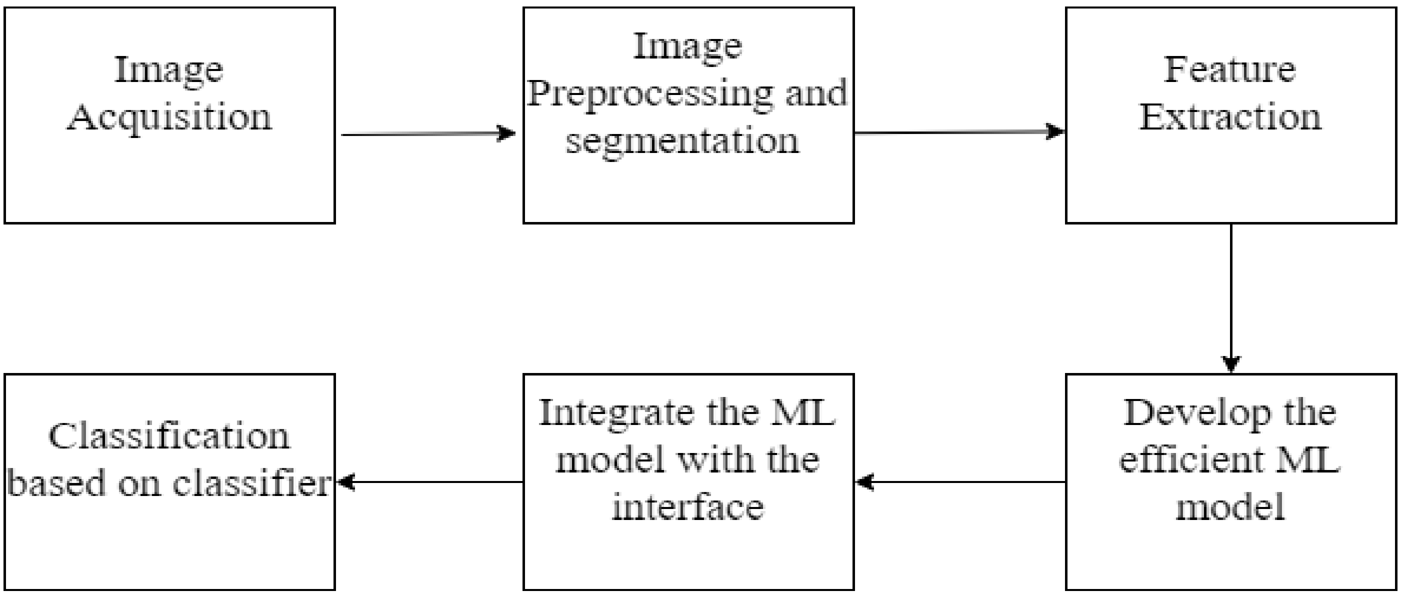


Fig. 4.1 Working Model

The test image is taken from the user for which the disease needs to be detected.Then the acquired image is cropped and the noise is removed from it. The system is mainly focused on the spots on the leaves. Then the neural network model is developed and then it is integrated with the system for prediction of the output class of the leaf whether the plant tends to be diseased or not. This system will be trained on multiple disease classes and each has various number training images so that it can detect efficiently and more accurately. The steps involved in the working model are shown in Fig. 4.1 and the same is explained below.

4.2) Steps Involved in the System Model

- First the system will acquire the images from the user that needs to be checked and analysed whether the disease is present in the leaf or not.

- Then the system will preprocess the image in order to remove the noise and concentrate only on the required part that is only on the green part of the leaf.

- Then the image segmentation is done and by using keras and tensorflow a convolutional neural network is developed.

- The neural network will be able to spot the color changes and the spots on the leaves so that it can predict the output.

- The images are not converted into the grayscale images. If we convert them to the grayscale the output accuracy will be high but the color changes of the leaves cannot be detected in this process.

- Because of not detecting the color changes in the leaves there may be the chance of getting the false positives and false negatives. In this process the false positive rate will be high but we need the false positives and negatives to be low in the image processing techniques.

- By using the neural network layers we can be able to predict the output with the help of epochs and at the end of each epoch the percentage of the accuracy is developed and at the end of all the 16 epochs the final accuracy is done.

CHAPTER 5 – IMPLEMENTATION

5.1) Convolutional Neural Network

Fig. 5.1.1 shows Convolutional neural network architecture which has several improvements compared with the other networks in inception family it has smoothing, 7x7 convolutions and it uses auxiliary classifier to have the label information in the network.In this the final layer will be modified in order to predict the output in the favour of the users.

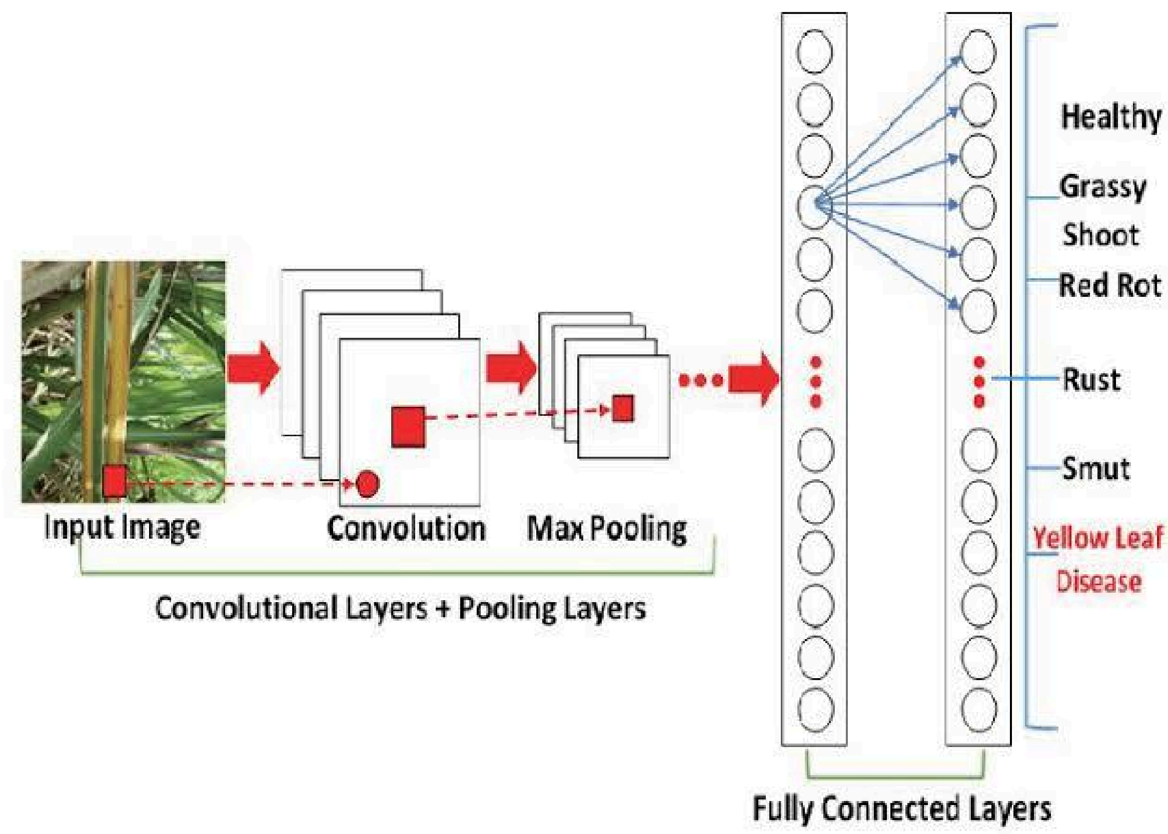


Fig. 5.1.1. CNN architecture

5.2) Libraries Used :

1. TensorFlow
2. Pandas
3. Numpy
4. CV2
5. Skimage
6. Sklearn
7. Random
8. tqdm

5.3) Converting image to numpy array:

```
def convert_image_to_array(image_dir):
    try:
        image = cv2.imread(image_dir)
        if image is not None:
            image = cv2.resize(image, DEFAULT_IMAGE_SIZE)
            return img_to_array(image)
        else:
            return np.array([])
    except Exception as e:
        print(f"Error : {e}")
        return None
```

In this the pixels of the single image are converted to the numpy array so that machine can understand. At the end the reverse of this process is done by the use of label binarizer by converting the numpy array of binary numbers into the text form so that the model can predict the accurate name of the disease for the respective leaf.

5.4) Dataset Used

Fig. 5.4 Dataset

This Fig. 5.4 is the sample dataset used for this project. These types of images are used because the images are pretty much clear and the noise in the images is less and the dataset has a high number of images with the different angles which will be more helpful in the prediction.

5.5) Appending all images to image array:

```python
image_list, label_list = [], []

try:
    print("[INFO] Loading images ...")
    plant_disease_folder_list = listdir(train_dir)

    for plant_disease_folder in plant_disease_folder_list[:8]:
        print(f"[INFO] Processing {plant_disease_folder} ...")
        plant_disease_image_list = listdir(f"{train_dir}/{plant_disease_folder}/")

        for image in plant_disease_image_list[:500]:
            image_directory = f"{train_dir}/{plant_disease_folder}/{image}"
            if image_directory.endswith(".jpg")==True or image_directory.endswith(".JPG")==True:
                image_list.append(convert_image_to_array(image_directory))
                label_list.append(plant_disease_folder)

    print("[INFO] Image loading completed")
except Exception as e:
    print(f"Error : {e}")

# Transform the loaded training image data into numpy array
np_image_list = np.array(image_list, dtype=np.float16) / 225.0
print()

# Check the number of images loaded for training
image_len = len(image_list)
print(f"Total number of images: {image_len}")
```

In this all the images required in the training process will be appended into the image array. After the conversion of each image into the numpy array all the image pixels that are converted into a numpy array will be appended to the image array so that it will be evaluated and analyzed more accurately.

5.6) Augment and split:

Augment and Split Dataset

```
[ ] augment = ImageDataGenerator(rotation_range=25, width_shift_range=0.1,
                                 height_shift_range=0.1, shear_range=0.2,
                                 zoom_range=0.2, horizontal_flip=True,
                                 fill_mode="nearest")
```

```
[ ] print("[INFO] Splitting data to train and test...")
    x_train, x_test, y_train, y_test = train_test_split(np_image_list, image_labels, test_size=0.2, random_state = 42)
```

In this the dataset used is being splitted into training and testing sets of the images and images are augmented appropriately. Augmentation is very important in image processing projects because the accuracy and correctness of the prediction depends upon the augmentation that is cropping and flipping etc…

The dataset is split into 80 percent of the images for training and 20 percent of the images for testing. Among 32 diseases 8 classes of images are taken. On an average each class of disease has around 1000 images from which 800 images are taken for training and the rest 200 are taken for testing.

5.7) Model Building:

Build Model

```
EPOCHS = 16
STEPS = 100
LR = 1e-3
BATCH_SIZE = 32
WIDTH = 256
HEIGHT = 256
DEPTH = 3
```

```
model = Sequential()
inputShape = (HEIGHT, WIDTH, DEPTH)
chanDim = -1

model.add(Conv2D(32, (3, 3), padding="same",input_shape=inputShape))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))

model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))

model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))

model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
```

```
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))

model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))

model.add(Conv2D(128, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))

model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Flatten())

model.add(Dense(1024))
model.add(Activation("relu"))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(n_classes))
model.add(Activation("softmax"))

model.summary()
```

5.8) Training the model:

Train Model

```
[ ]  # Initialize optimizer
     opt = Adam(lr=LR, decay=LR / EPOCHS)

     # Compile model
     model.compile(loss="binary_crossentropy", optimizer=opt, metrics=["accuracy"])

     # Train model
     print("[INFO] Training network...")
     history = model.fit_generator(augment.flow(x_train, y_train, batch_size=BATCH_SIZE),
                                   validation_data=(x_test, y_test),
                                   steps_per_epoch=len(x_train) // BATCH_SIZE,
                                   epochs=16,
                                   verbose=1)
```

The model is trained with the help of adam optimizer which is very much better when compared to other optimizers and the learning rate is set to 0.001 then the model will be compiled on the metrics of accuracy for the training of the model. The epochs will be 16 for prediction of the accuracy and verbose is set to 1 if the verbose is set to 0 the detailed progress of training the network will be hidden but if we set the verbose value to 1 then the number of epochs are shown as the model is being constructed and also it shows the weights of all the items as well.

5.9) Evaluating the model:

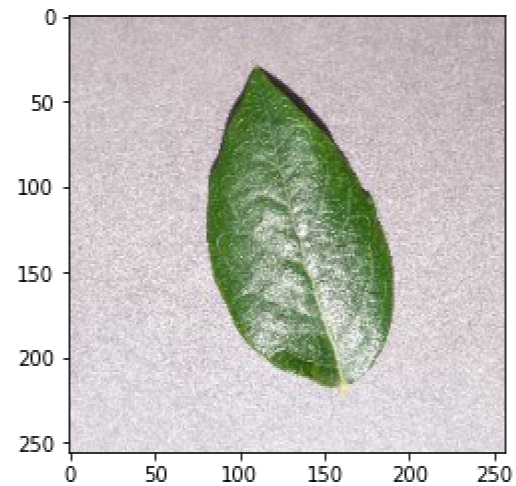
Evaluate Model

```
[ ]  acc = history.history['accuracy']
     val_acc = history.history['val_accuracy']
     loss = history.history['loss']
     val_loss = history.history['val_loss']
     epochs = range(1, len(acc) + 1)

     # Train and validation accuracy
     plt.plot(epochs, acc, 'b', label='Training accurarcy')
     plt.plot(epochs, val_acc, 'r', label='Validation accurarcy')
     plt.title('Training and Validation accurarcy')
     plt.legend()

     plt.figure()

     # Train and validation loss
     plt.plot(epochs, loss, 'b', label='Training loss')
     plt.plot(epochs, val_loss, 'r', label='Validation loss')
     plt.title('Training and Validation loss')
     plt.legend()
     plt.show()
```

In the evaluation of the model the training accuracy vs training loss and validation accuracy vs validation loss graphs are built in which b stands for blue line in the graph and r stands for red line in the graph in which the metrics are the values that shows how much is the validation and training accuracy and loss.

CHAPTER 6 – RESULTS

Healthy Image of the leaf Diseased image of the leaf



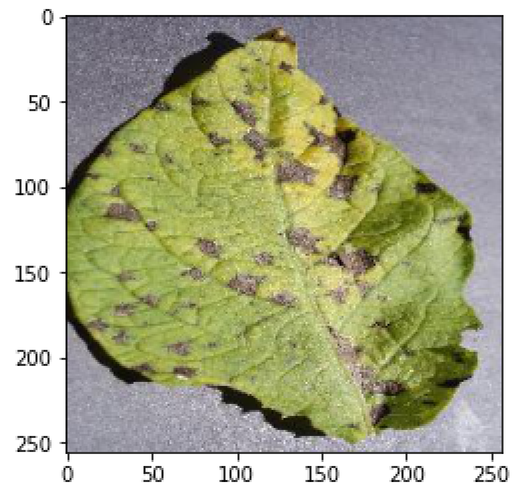


Fig. 6.1.1. **Fig. 6.1.2.**

The left side image of the two leaves is the healthy image of the leaves and the right side image is the diseased image of the leaves that means the plant which is taken tends to be diseased. As Fig. 6.1.1 and Fig. 6.1.2 shows the gray part in the images is the noise part of the image and it needs to be removed when the analysis to be done or else the values of the prediction may change and may have the high chance of getting the false negatives because the model will fail to focus on the relevant parts that are the color change and the spots on the leaves.

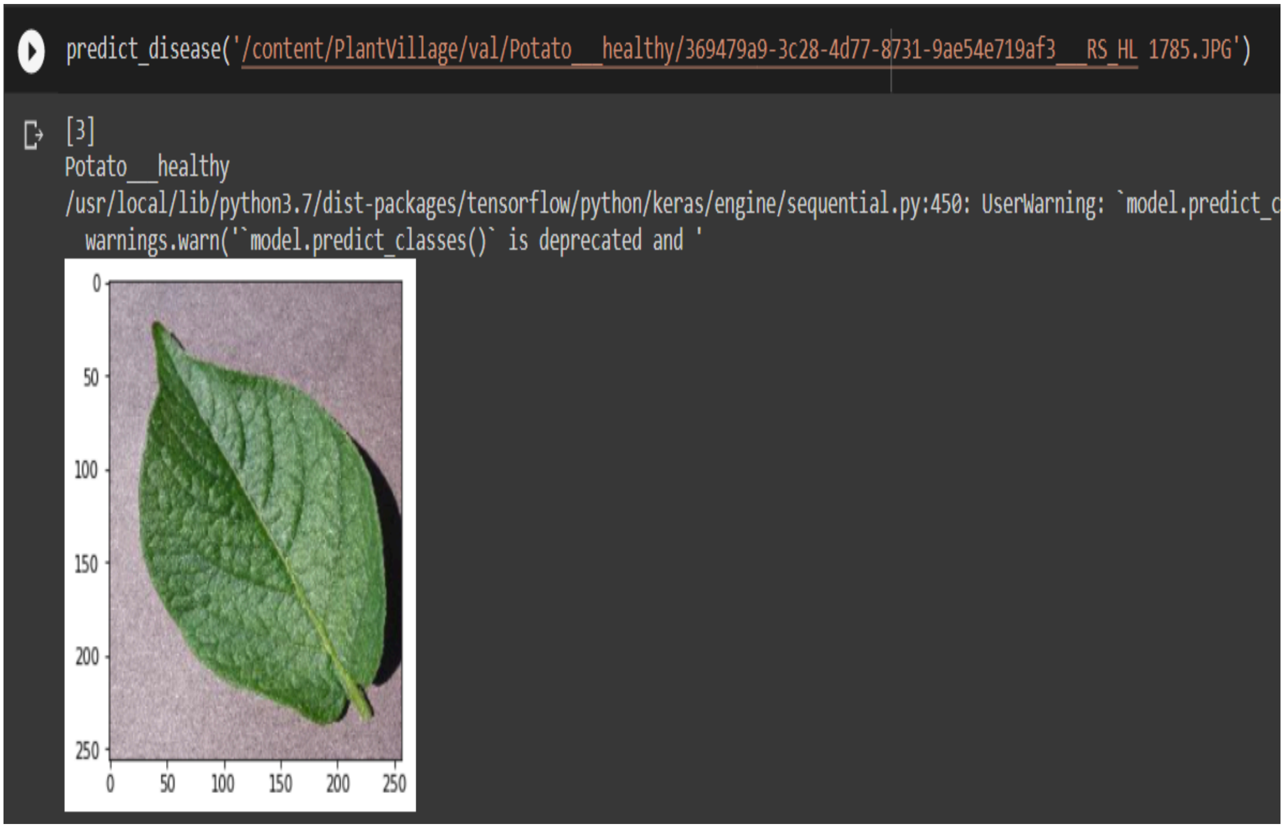


Fig. 6.1.3. Potato Healthy

Fig. 6.1.3 is the healthy image of the leaf of a potato plant and this is the test image of the healthy leaf that the model predicted correctly, as shown the gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This healthy leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased.

Fig. 6.2. Tomato Mosaic Virus

Fig. 6.2 is the diseased image of the leaf of a tomato plant and this is the test image of the diseased leaf that the model predicted correctly, as shown in Fig. 6.2 the gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This diseased leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased. The tomato mosaic virus is the virus that occur very commonly in the plants but it is very dangerous as it is not only destroys the plant but also it destroys the yield and can cause the damage to the ripening of the tomato.

Fig. 6.3. Cherry Powdery Mildew

Fig. 6.3 is the diseased image of the leaf of a tomato plant and this is the test image of the diseased leaf that the model predicted correctly. The gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This diseased leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased. Then cherry powdery mildew is caused due to the biotropic virus it can rotten the cherry even before the proper growth.

Fig. 6.4. Corn Northern Leaf Blight

Fig. 6.4 is the diseased image of the leaf of a tomato plant and this is the test image of the diseased leaf that the model predicted correctly. The gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This diseased leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased.This disease is caused by the fungus which starts at the lower parts of the leaves and spreads all over the tree this disease can cause the yield loss to the susceptible corn hybrids.

Fig. 6.5. Soybean Healthy

Fig. 6.5 is the healthy image of the leaf of a soybean plant and this is the test image of the healthy leaf that the model predicted correctly. The gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This healthy leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased.

```
def predict_disease(image_path):
    image_array = convert_image_to_array(image_path)
    np_image = np.array(image_array, dtype=np.float16) / 225.0
    np_image = np.expand_dims(np_image,0)
    plt.imshow(plt.imread(image_path))
    result = model.predict_classes(np_image)
    #print(result)
    print((label_binarizer.classes_[result][0]))
```

```
[ ] predict_disease('/content/PlantVillage/val/Blueberry___healthy/008c85d0-a954-4127-bd26-861dc8a1e6ff___RS_HL 2431.JPG')
```

```
[0]
Blueberry___healthy
/usr/local/lib/python3.7/dist-packages/tensorflow/python/keras/engine/sequential.py:450: UserWarning: `model.predict_cl
  warnings.warn('`model.predict_classes()` is deprecated and '
```

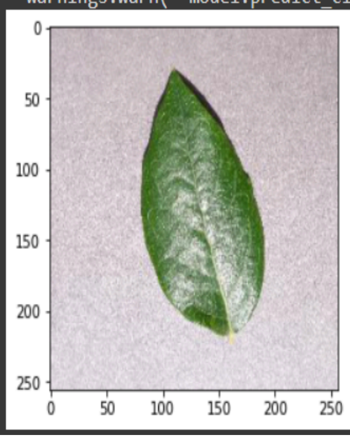


Fig. 6.6. BlueBerry Healthy

Fig. 6.6 is the healthy image of the leaf of a blueberry plant and this is the test image of the healthy leaf that the model predicted correctly. The gray part of the image is less when compared to the actual image that is taken manually that is why the accuracy will be high. This healthy leaf is the part of the images of one of the image classes that are taken into consideration for training and testing purposes. It is one image class among the eight classes of images which are healthy and tend to be diseased.

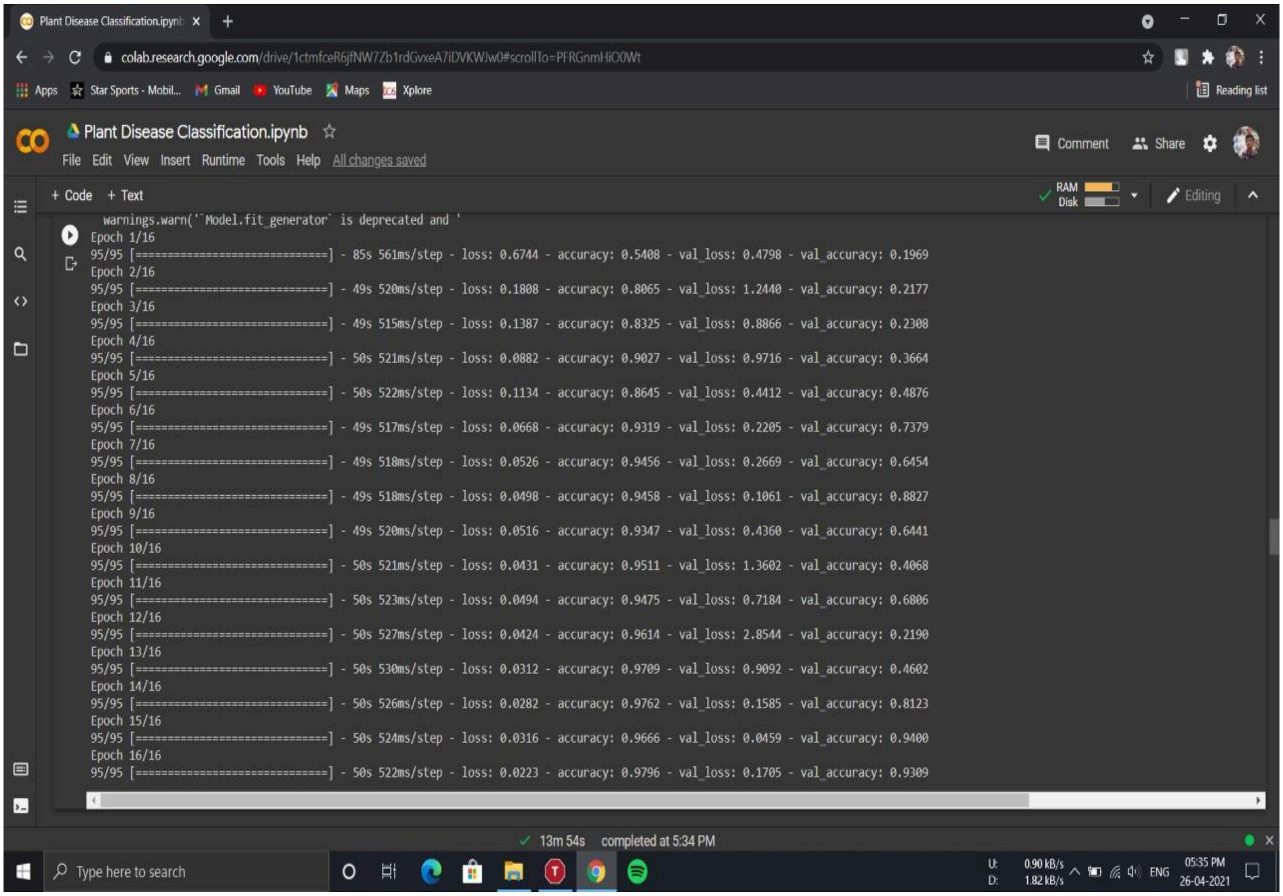
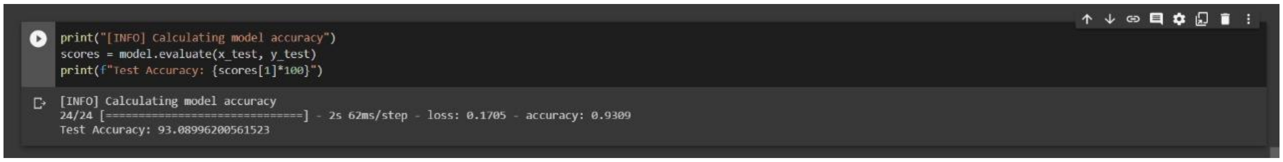


Fig. 6.7. Epochs

As Fig. 6.7 shows, the model trained in 16 epochs in order to the calculation of the accuracy and loss of training and validation and at the end of each epoch the accuracy is calculated and it will be gradually growing but at the end of all the 16 epochs we acquired the accuracy of 93 percent for the 8 classes of the images which are to be classified whether the leaf is to be diseased or not.

Fig. 6.8. The total accuracy is the accuracy after the completion of the final epoch.



```
print("[INFO] Calculating model accuracy")
scores = model.evaluate(x_test, y_test)
print(f"Test Accuracy: {scores[1]*100}")

[INFO] Calculating model accuracy
24/24 [==============================] - 2s 62ms/step - loss: 0.1705 - accuracy: 0.9309
Test Accuracy: 93.08996200561523
```

Fig. 6.8. Total Accuracy

Fig. 6.9. Training and the validation graphs are used for calculating how much the accuracy and loss is getting for the model.

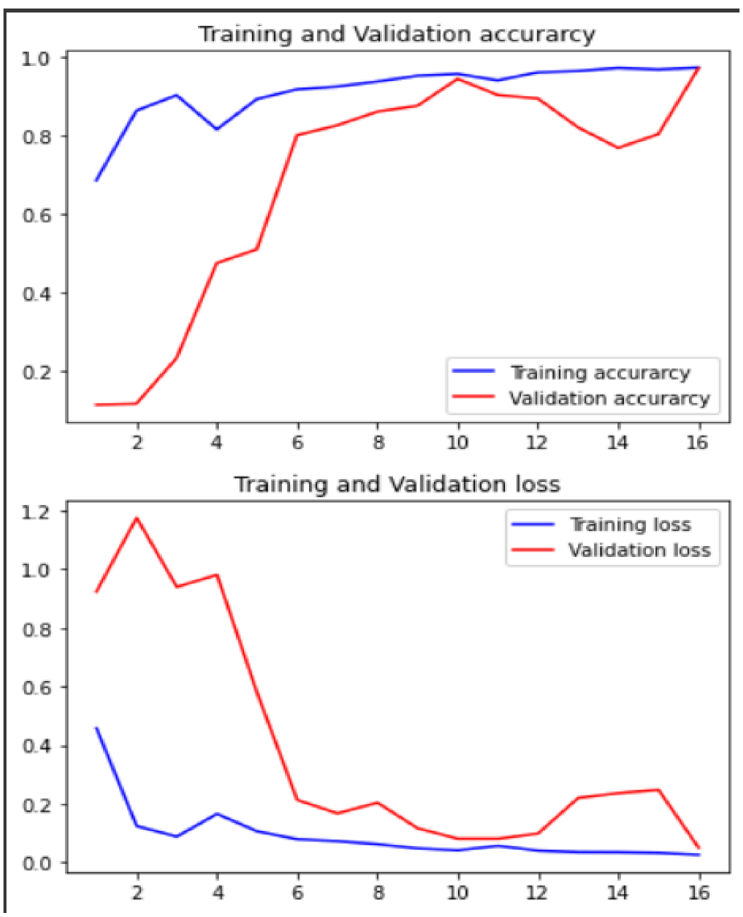


Fig. 6.9. Training and Validation Accuracy

CHAPTER 7 – CONCLUSION AND FUTURE ENHANCEMENT

Conclusion:

The system can now classify the plant disease in the right manner whether it is healthy or sick that is to be cancerous. The system will preprocess the image and then match the grayscale image with the image dataset and predict the output accordingly. The dataset is selected from kaggle which consists of 3900 images, out of that 3900 images 2700 will be the training part images and the rest 1200 images will be in the testing part. While in training, the images are trained in such a way that there will be all possible angles and all possible lighting conditions. The dataset itself consists of ROI images which concentrate mainly on the leaf part of the image. The ROI images which are included in the dataset helps very well because it concentrates on the high cell change in one image. The code is implemented using collabs with the help of jupyter notebook. Keras and Tensor flow are used for creating the model and libraries mentioned in the implementation part for classification of the leaf images. The model is trained with 25 epochs and at the end of each epoch can get the accuracy and plotted in the graph and at the end of the 25th epoch, the final accuracy is achieved is 93.2%. The main issue in this plant disease detection is the speed and the accuracy of classification which can be done using efficient classification algorithms like SVM and random forest. This project implements an innovative idea to identify the affected crops and provide remedy measures to the agricultural industry. By the use of an efficient clustering algorithm, the infected region of the leaf is segmented and analyzed. It provides a good choice for the agriculture community particularly in remote villages. It acts as an efficient system in terms of reducing clustering time and the area of infected regions.

Future Enhancement:

The implemented software model can be used in a hardware system which will be able to detect the diseases that are occurring frequently in plants. The detection and classification can be done in real time if advanced algorithms are used and with the help of a high resolution camera that can focus throughout the farm so that it can detect from one place. The alarming system can also be given so that it can be an alert to the farmer in the early stage of the disease spread. The feed forward and the cascade neural networks can be used with the help of other classes of diseases so that the accuracy of the system can be increased. The Feed Forward and Cascaded Feed algorithms can be expanded for detection of multiple diseases on a significantly large scale. By increasing the number of features and the number of inputs to the Neural Network the algorithms can be enhanced. The system can be integrated with the android system so that it will be the greatest asset in the agricultural sector. It will be most helpful for the farmers in the early stage itself.

Fig. 7.1 Future Scope

Fig 7.1 describes using drones to patrol fields and alert farmers about the disease spread rate to prevent over-infection of crops.

REFERENCES

[1] Abirami Devaraj, Karunya Rathan, Sarvepalli Jaahnavi, K Indira, “Identification of Plant Disease using Image Processing Technique”, International Conference on Communication and Signal Processing, 2019

[2] Sharath D M, Akhilesh, S Arun Kumar, Rohan M G and Prathap C, “Image based Plant Disease Detection in Pomegranate Plant for Bacterial Blight”, International Conference on Communication and Signal Processing, 2019

[3] M.N. Abu Bakar, A.H. Abdullah, N. Abdul Rahim, H. Yazid, S.N. Misman, M.J. Masnan, “Rice Leaf Blast Disease Detection Using MultiLevel Colour Image Thresholding”, Journal of Telecommunication, Electronic and Computer Engineering, Vol. 10 No. 1-15, 2018

[4] Sachin D. Khirade, A. B. Patil, “Plant Disease Detection Using Image Processing”, International Conference on Computing Communication Control and Automation, 2015

[5] Sanjay B. Dhaygude, Nitin P. KumbharAgricultural “Plant leaf disease detection using image processing” Int J Adv Res Electr Electron Instrum Eng, 2 (1) (2013)

[6] 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, 15 (1) (2013), pp. 211-217

[7] Smita Naikwadi, Niket Amoda "Advances in image processing for detection of plant diseases" Int J Appl Innov Eng Manage, 2 (11) (2013)

[8] Bernardes, Alexandre A., *et al.* "Identification of foliar diseases in cotton crop" R.S. Tavares João Manuel, Natal Jorge Renato (Eds.), Topics in medical image processing and computational vision, Lecture Notes in Computational Vision and Biomechanics (2013), pp. 67-85

[9] S. Phadikar, A. Das and J. Sil, "Classification of Rice Leaf Diseases Based on Morphological Changes," International Journal of Information and Electronics Engineering, vol. 2, 2012

[10]K. Padmavathi, and K. Thangadurai, "Implementation of RGB and Gray scale images in plant leaves disease detection –comparative study," Indian J. of Sci. and Tech., vol. 9, pp. 1- 6,Feb. 2016

[11]Dr.K.Thangadurai, K.Padmavathi, "Computer Vision image Enhancement For Plant Leaves Disease Detection", 2014 World Congress on Computing and Communication Technologies.

[12] C.J. Chuwa, R.B. Mabagala2, M. S.O.W. Reuben," Assessment of Grain Yield Losses Caused by Rice Blast Disease in Major Rice Growing Areas in Tanzania", International Journal of Science and Research, Vol. 4, Issue 10, 2015.

[13] Y. Q. Xia, Y. Li, and C. Li, “Intelligent Diagnose System of Wheat Diseases Based on Android Phone,” J. of Infor. & Compu. Sci., vol. 12, pp. 6845-6852, Dec. 2015.

[14] Wenjiang Huang, Qingsong Guan, JuhuaLuo, Jingcheng Zhang, Jinling Zhao, Dong Liang, Linsheng Huang, and Dongyan Zhang, “New Optimized Spectral Indices for Identifying and Monitoring Winter Wheat Diseases”, IEEE journal of selected topics in applied earth observation and remote sensing,Vol. 7, No. 6, June 2014

[15] S. Sankarana, A. Mishraa and R. Ehsania, "A review of advanced techniques for detecting plant diseases," Computers and Electronics in Agriculture, vol. 72, pp. 1-13, 2010

[16] Monica Jhuria, Ashwani Kumar, and RushikeshBorse, “Image Processing For Smart Farming: Detection Of Disease And Fruit Grading”, Proceedings of the 2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)

[17]Zulkifli Bin Husin, Abdul Hallis Bin Abdul Aziz, Ali Yeon Bin MdShakaffRohaniBinti S Mohamed Farook, “Feasibility Study on Plant Chili Disease Detection Using Image Processing Techniques”, 2012 Third International Conference on Intelligent Systems Modelling and Simulation.

[18] S. Phadikar , J. Sil and A. K. Das, "Rice diseases classification using feature selection and rule generation techniques," Computers and Electronics in Agriculture, vol. 90, p. 76–85, 2013.

[19] E. Singh and M. L. Singh, "Automated Colour Prediction of Paddy Crop Leaf using Image Processing," in IEEE International Conference on Technological Innovations in ICT for Agriculture and Rural Development, 2015.

[20]Mrunalini R. Badnakhe, Prashant R. Deshmukh, “Infected Leaf Analysis and Comparison by Otsu Threshold and k-Means Clustering”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 3, March 2012.

[21] Saradhambal .G , Dhivya. R, Latha. S, R. Rajesh “Plant Disease Detection and Its Solution using Image Classification” International Journal of Pure and Applied Mathematics. Volume 119 No. 14 2018, 879-884

[22]Chunxia Zhang, Xiuqing Wang, Xudong Li, “Design of Monitoring and Control Plant Disease System Based on DSP and FPGA”, 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing.

[23] E. S. Sama and A.K. Prabhavathy and J.D. Shree, " A Survey on Outdoor Scene Image Segmentation," International Journal of Computer Applications, vol. 55, no. 9, 2012