VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

Accredited by NAAC & NBA*

SVCE BENGALURU

SRI VENKATESHWARA COLLEGE OF ENGINEERING

— Affiliated to VTU, Approved by AICTE, Recognised by UGC u/s 2(f) & 12(B)—

Department of Electronics & Communication Engineering

Project presentation on

"Plant Growth monitoring System Using Artificial Intelligence"

Presented by:

MITHUN KUMAR J N (1VE18EC054)

PRAJWAL M S (1VE18EC070)

SUMANTH GOWDA V (1VE18EC092)

YOGESH J (1VE18EC103)

Under the guidance of:

Dr. POORNIMA G R

Professor & Dean Academics

Department of E&CE

SVCE, Bengaluru

CONTENTS

INTRODUCTION

- Agriculture is the backbone of human existence on the earth. As the population increases, demand for the food increases this depend on the plant growth for food.
- It results in great pressure on agriculture industry to secure the growing demands for the food.
- Innovations in agriculture are increasingly needed to secure a growing world demand for food, in order to conserve and optimize the use of limited natural resources and to sustain the environment’s ability to provide economic, social, and environmental services to society.
- So, to maintain the growth and food production properly we are working on the Plant Growth Monitoring System using Artificial Intelligence.

INTRODUCTION

- AI based Image processing is the process of implementation, analysis and manipulation of a digitized image, especially in order to improve its quality.
- Our project deals with the capturing the image of the plant in its growth and then based on appearance of the plants it would be processed for predicting plant growth parameters.
- After preprocessing the images it will be classified and trained using some Machine learning algorithm.

PROBLEM STATEMENT

- Plant growth plays a key role in getting a good yield for the farmer. The factors that affect plant growth are light, water, temperature and nutrients. These four elements affect the plant's growth hormones, making the plant grow more quickly or more slowly. Changing any of the four can cause the plant stress which stunts or changes growth, or improves growth .

LITERATURE SURVEY

Sl No	Paper	Pro's	Con's
1.	Monirul Islam Pavel, Syed Mohammad Kamruzzaman, Sadman Sakib Hasan, "An IoT Based Plant Health Monitoring System Implementing Image Processing", Volume-5, Nov-2019.	• Adding these functionalities to any current agribusiness industry can possibly incredibly recognize plants wellbeing and ready to take choices a lot quicker.	• Gives just information about execution of the project but not executed practically
2	Rohit Nalawade, Apoorv Nagap, Lakhan Jindam, "Agriculture field monitoring and Plant Leaf Disease Detection", Volume-10, Issue-6, 2020.	• This system can periodically monitors the cultivated field and successfully shows the status on the application. • Water pump can be controlled through the application. • The accuracy of this model is 98.07%.	• The proposed system can be further extended by adding extra functionalities like location of stores present nearby user, list of pesticieds and fertilizers, real-time interaction with agriculture experts via chatting or video call, etc.

Contd..

3	Siddharth Singh Chouhan, Ajay Kaul, Uday Pratap Singh, “A deep learning approach for the classification of diseased plant leaf images”, Volume-6, issue-9, 2019.	• Used multilayer convolutional neural network for the classification of diseased plant leaf images. • The results were validated on the database acquired for four different plant leave images categorized among healthy and diseased. • The average accuracy of this model is 98.24%.	• In future, the presented model can be further enhanced for the classification of different plant leave and diseases.
4	Yingying Dong, G Fang Xu, “Monitoring and forecasting for disease and pest in crop based on WebGIS system”, Volume-2, issue-10, 2019.	• System development includes data, calculation and production modules, each module maintains a high cohesive and low coupling relationship. Through the browser to access the system, simple, quick, and easy to operate.	• This is the sophisticated project which include huge amount of data. So maintain the data and the result is the challenging task for the user.

OBJECTIVES

The objectives of the project are:

- To design a appropriate monitoring system.
- Use of Artificial Intelligence image processing to find the Growth of the Plant.
- To analyze the growth and to predict the health condition of the plant using leaves.
- To Identifying the Plant Disease using Machine learning algorithm. After the disease is identified, Identifying the suitable pest for that disease.

METHODOLOGY

- BLOCK DIAGRAM :

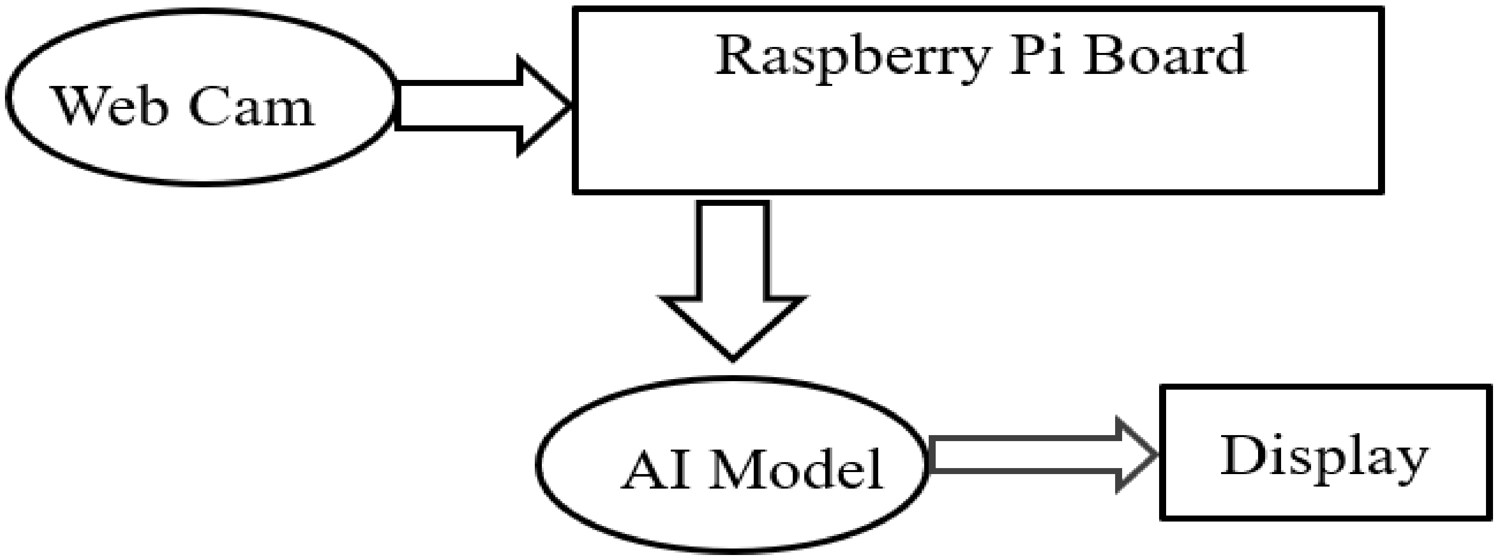


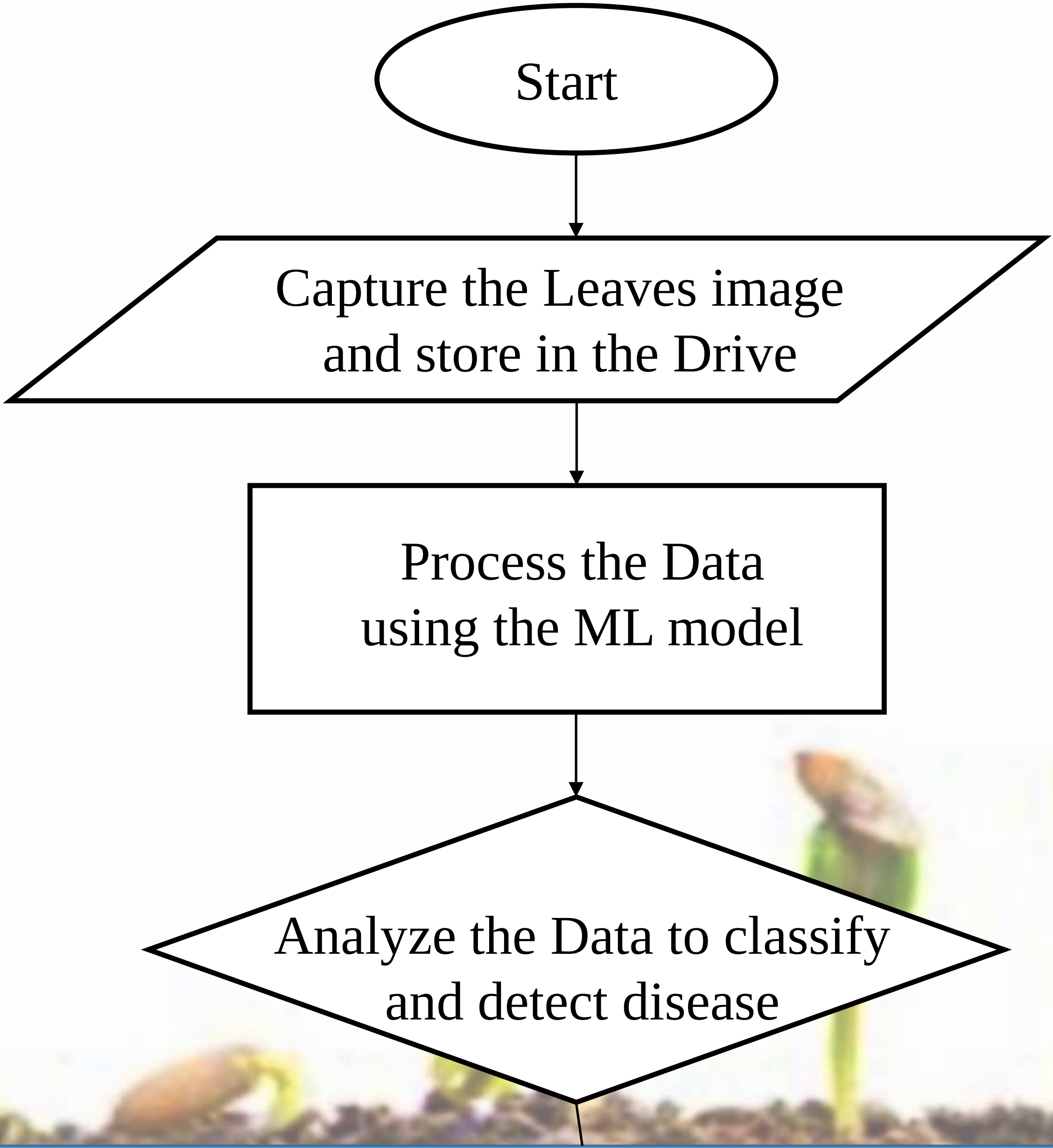
Figure 1: Block Diagram of Plant Monitoring System.

Contd..

The Block diagram consist of mainly 3 components: Webcam, Raspberry pi board and Ai model.

- Raspberry pi Board: The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. We are using Raspberry pi board because we will be able to interface camera to the board.
- Webcam: A webcam is a video camera that feeds or streams an image or video in real time to or through a computer network, such as the Internet. We are using webcam to capture of leaf image if the plant. Then the Captured image is made to store, which is further is used for detecting a plant condition using a machine learning algorithm.

Flow Diagram:



Contd..

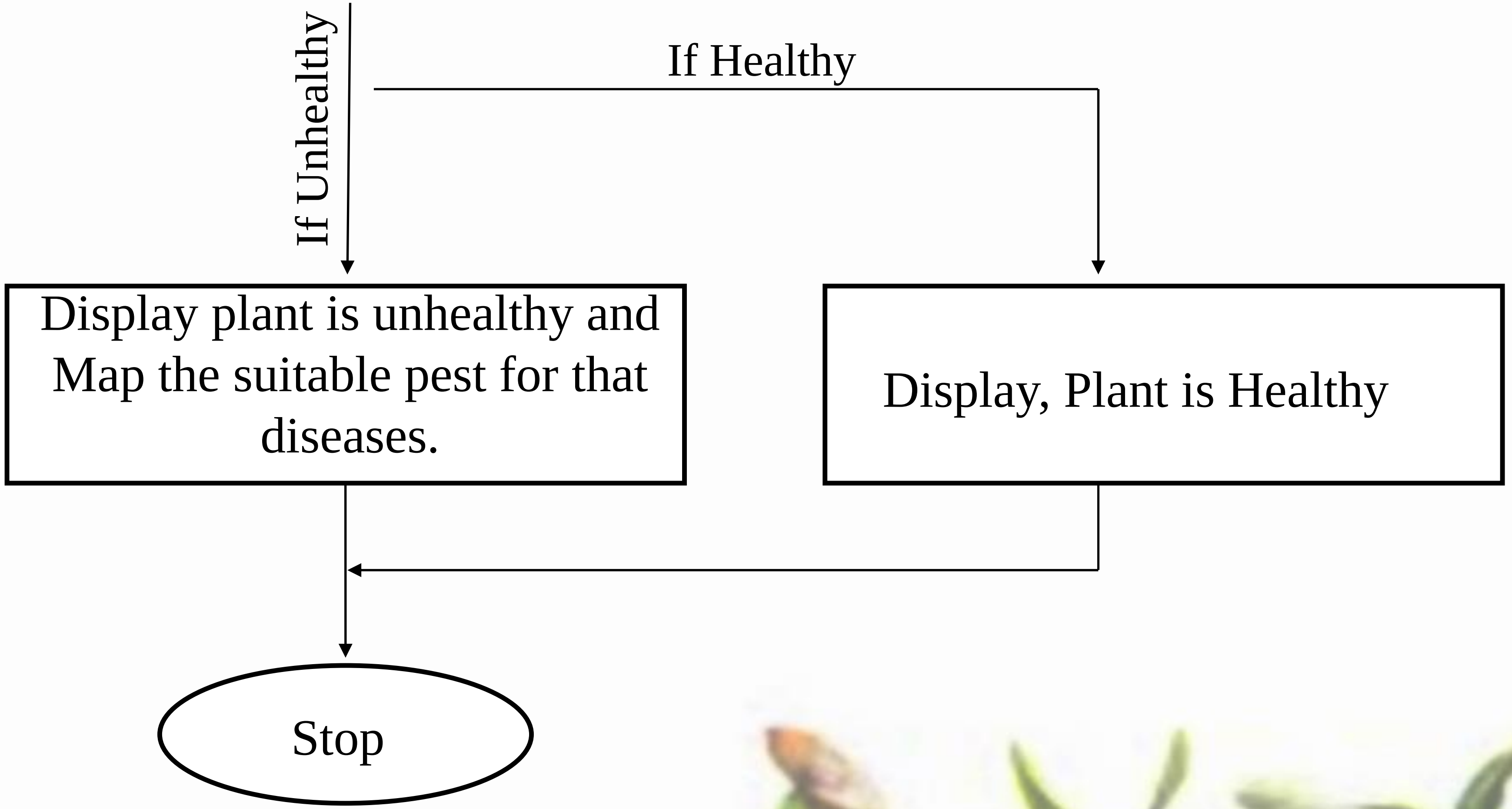


Figure 2: Flow Chart of the Plant Monitoring System

PROPOSED SYSTEM

WORKING

Hardware Used :

- As our project deals with the maintaining of the plant health, we require few hardware components to capture the image of the leaves.
- So we are using a Raspberry Pi board with a web camera to capture the image and to store in the system.
- Further this stored images are used as a data for the machine learning model in order to detect the Plant health condition.

Figure 3: Hardware of Plant Monitoring System

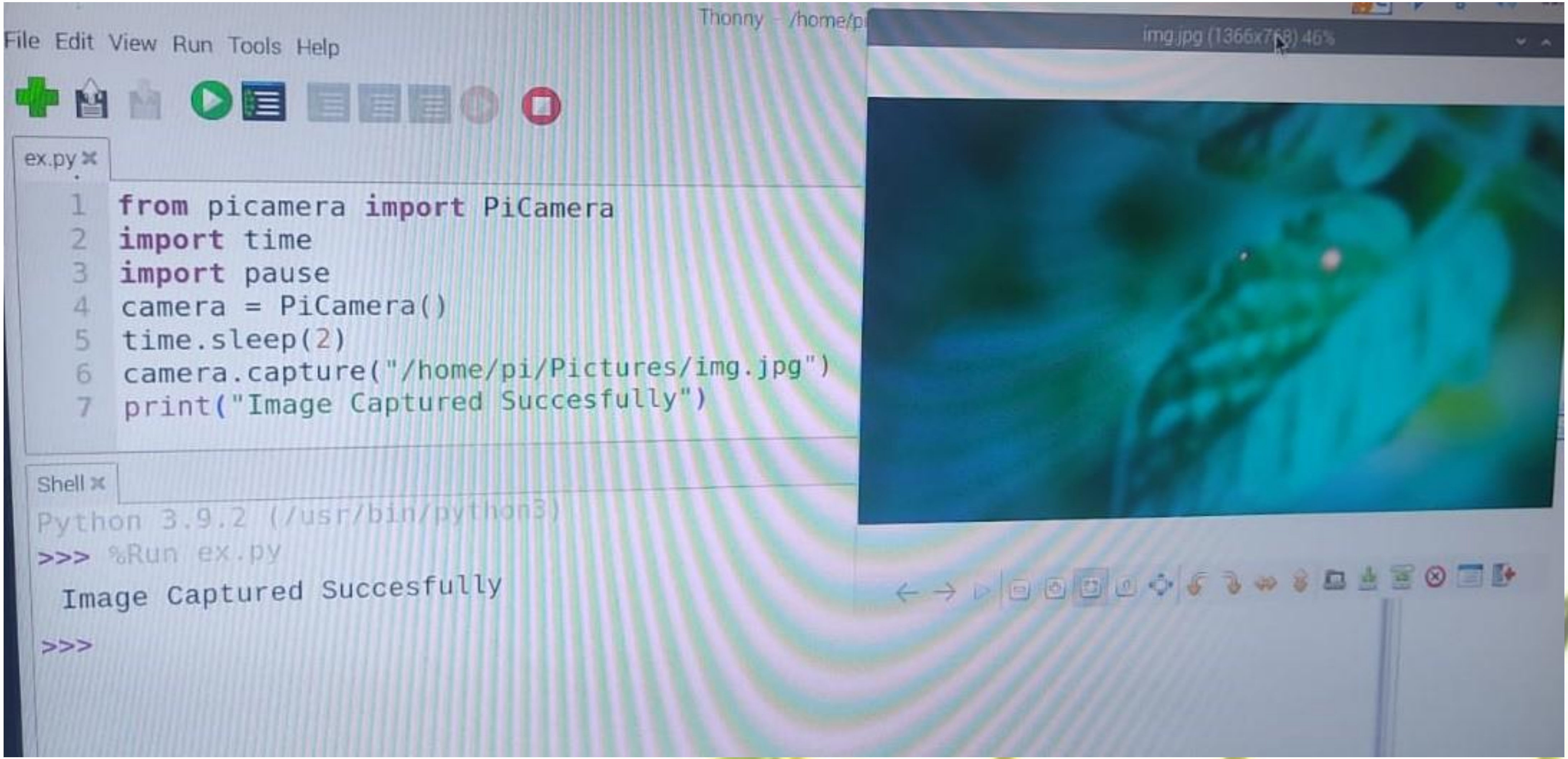


Figure 4: Capturing of Plant Image

Software process

- The development of system consist of 2 parts
- 1: Training of machine learning module with data sets
- 2: Detection of disease
- Figure 3 shows the block diagram of software part.

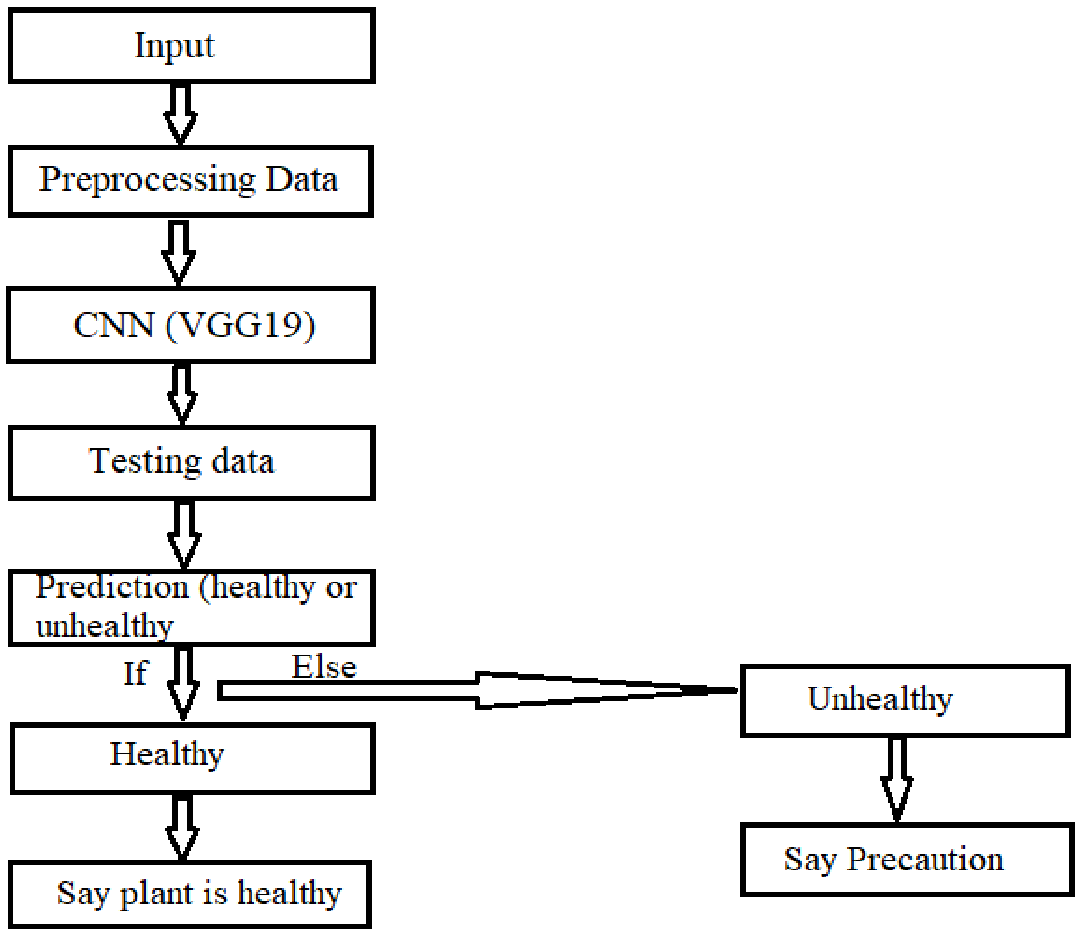


Figure 3: Flow chart of Training model for Plant Monitoring System

Contd..

- **Training of machine learning module with data sets:** The neural machine learning module is created using keras library and python in google collab editor The cnn model is built on vgg19 architecture.

- For training the module 1000's of images of plant leaves with disease are collected and stored in a file. The file is divided into two parts training datasets and testing data sets training data sets is used to train the machine learning module .It consists 38 classes of plant disease images labelled respectively. These images will be fed into machine learning module, and trained module is extracted

- **Detection of plant disease:** The detection of plant disease starts with the collection of plant leaf image sample at the farm, the image is then passed into trained neural module for diagnosis. The trained machine learning module will give output whether the sample image contains any disease and if disease is detected a precaution for that disease will also be displayed..

Contd..

- **SYSTEM ARCHITECTURE:**

- The system is machine learning based approach to detect the plant disease. It uses convolution neural network built on vgg 19 architecture.to detect disease.

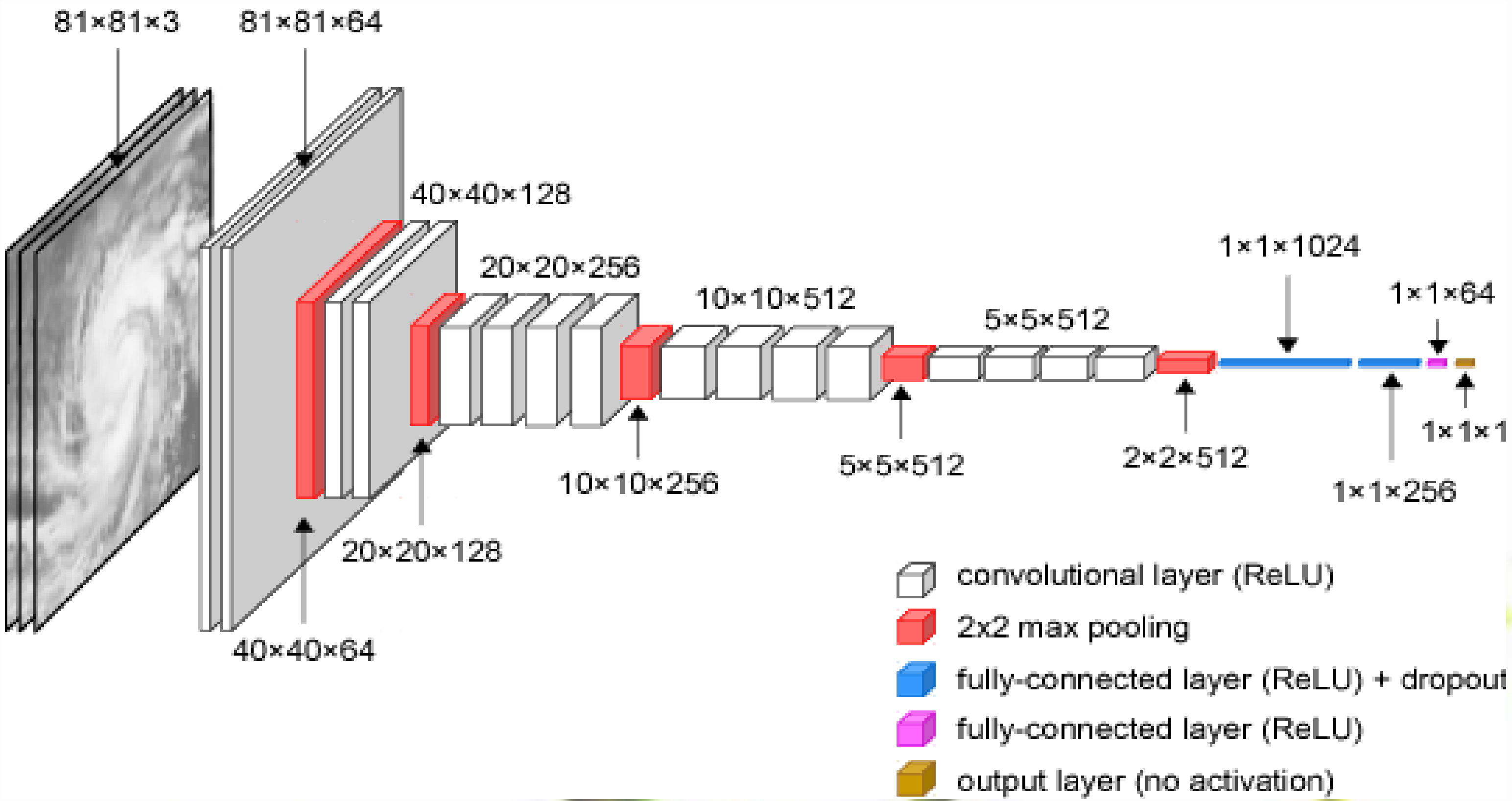


Figure 6: Plant Monitoring System – VGG19 Architecture

Contd..

- A fixed size of (224 * 224) RGB image was given as input to this network which means that the matrix was of shape (224,224,3).
- The only pre-processing that was done is that they subtracted the mean RGB value from each pixel, computed over the whole training set.
- Used kernels of (3 * 3) size with a stride size of 1 pixel, this enabled them to cover the whole notion of the image. Spatial padding was used to preserve the spatial resolution of the image.
- Max pooling was performed over a 2 * 2 pixel windows with stride 2. This was followed by Rectified linear unit(ReLu) to introduce non-linearity to make the model classify better and to improve computational time as the previous models used tanh or sigmoid functions this proved much better than those.
- implemented three fully connected layers from which first two were of size 4096 and after that a layer with 1000 channels for 1000-way *ILSVRC* classification and the final layer is a SoftMax function.

Contd..

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
[ ] !unzip /content/drive/MyDrive/archive.zip_dl=0
```

```
Streaming output truncated to the last 5000 lines.
  inflating: new plant diseases dataset(augmented)/New Plant Diseases Datase
  inflating: new plant diseases dataset(augmented)/New Plant Diseases Datase
  inflating: new plant diseases dataset(augmented)/New Plant Diseases Datase
  inflating: new plant diseases dataset(augmented)/New Plant Diseases Datase
```

Figure 7: Plant Monitoring System - Software Output

Contd..

Figure 8: Plant Monitoring System – Train Data

Contd..

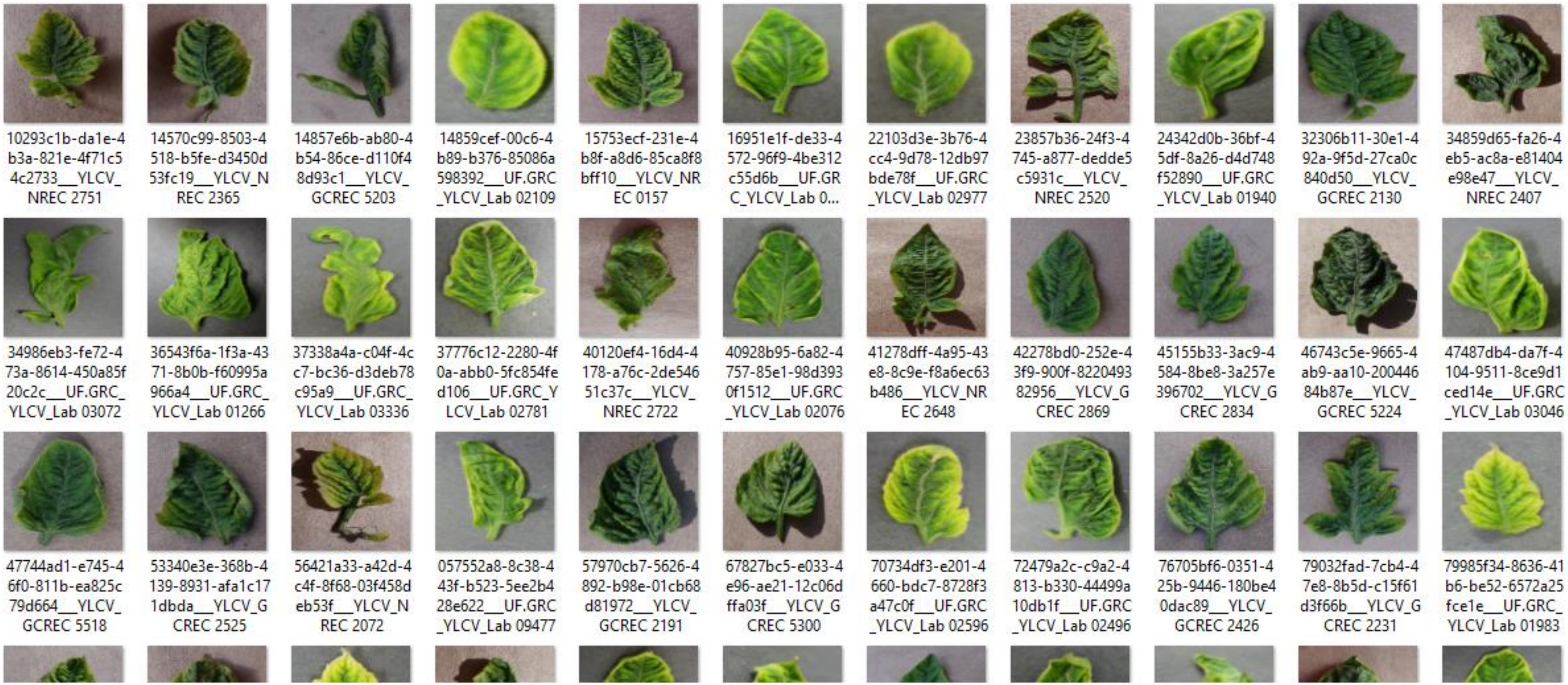


Figure 9: Plant Monitoring System – Valid Data

Contd..

```
] import numpy as np
import matplotlib.pyplot as plt
import keras
import pandas
from keras.preprocessing.image import img_to_array
import os
from keras.preprocessing.image import load_img
from keras.preprocessing.image import ImageDataGenerator
from keras.applications.vgg19 import VGG19,preprocess_input,decode_predictions
training_data_generator= ImageDataGenerator(zoom_range=0.5, shear_range=0.3, rescale=1/255, horizontal_flip=True)
validation_data_generator= ImageDataGenerator(rescale= 1/255)
train = training_data_generator.flow_from_directory(directory="/content/drive/MyDrive/train",target_size=(256,256),batch_size=32)
val = validation_data_generator.flow_from_directory(directory="/content/drive/MyDrive/valid",target_size=(256,256),batch_size=32)
from keras.layers import Dense, Flatten
from keras.models import Model
from keras.applications.vgg19 import VGG19
import keras
base_model =VGG19(input_shape=(256,256,3),include_top=False)
for layer in base_model.layers:
  layer.trainable=False
x =Flatten()(base_model.output)
x= Dense(units=38, activation='softmax')(x)
model =Model(base_model.input, x)
model.compile(optimizer='adam',loss=keras.losses.categorical_crossentropy,metrics=['accuracy'])
from keras.callbacks import ModelCheckpoint, EarlyStopping
es =EarlyStopping(monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1)
mc =ModelCheckpoint(filepath="best_model.h",monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1,save_best_only=True)
cb=[es,mc]
his = model.fit_generator(train,steps_per_epoch=16,epochs=50,verbose=1,callbacks=cb,validation_data=val,validation_steps=16)
```

Figure 10: Plant Monitoring System - Software Output

Contd..

```
def prediction(path):
  img=load_img(path,target_size=(256,256))
  i=img_to_array(img)
  im=preprocess_input(i)
  img=np.expand_dims(im,axis=0)
  pred =np.argmax(model.predict(img))
  print(pred)
  print(f"The plant diagnosed as{ref[pred]}")
  path="/content/drive/MyDrive/precaution/"+f'{pred}'+".txt"
  f=open(path)
  print(f.read())
```

```
Found 17034 images belonging to 11 classes.
Found 17582 images belonging to 38 classes.
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/vgg19/vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5
80142336/80134624 [==============================] - 1s 0us/step
80150528/80134624 [==============================] - 1s 0us/step
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:29: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`,
Epoch 1/50
```

Figure 11: Plant Monitoring System - Software Output

Contd..

```
path="/content/drive/MyDrive/livetest/leaveimage.jpg"
prediction(path)
```

```
32
The plant diagnosed asTomato___Septoria_leaf_spot
Tomato_Septoria_leaf_spot

1.      Removal and destruction of the affected plant parts.
2.     Seed treatment with Thiram or Dithane M-45 (2 g/kg seed) is useful in checking seed borne infection.
3.     In the field spraying with Mancozeb 0.2 % effectively controls the disease.
```

Figure 12: Plant Monitoring System - Software Output

Contd..

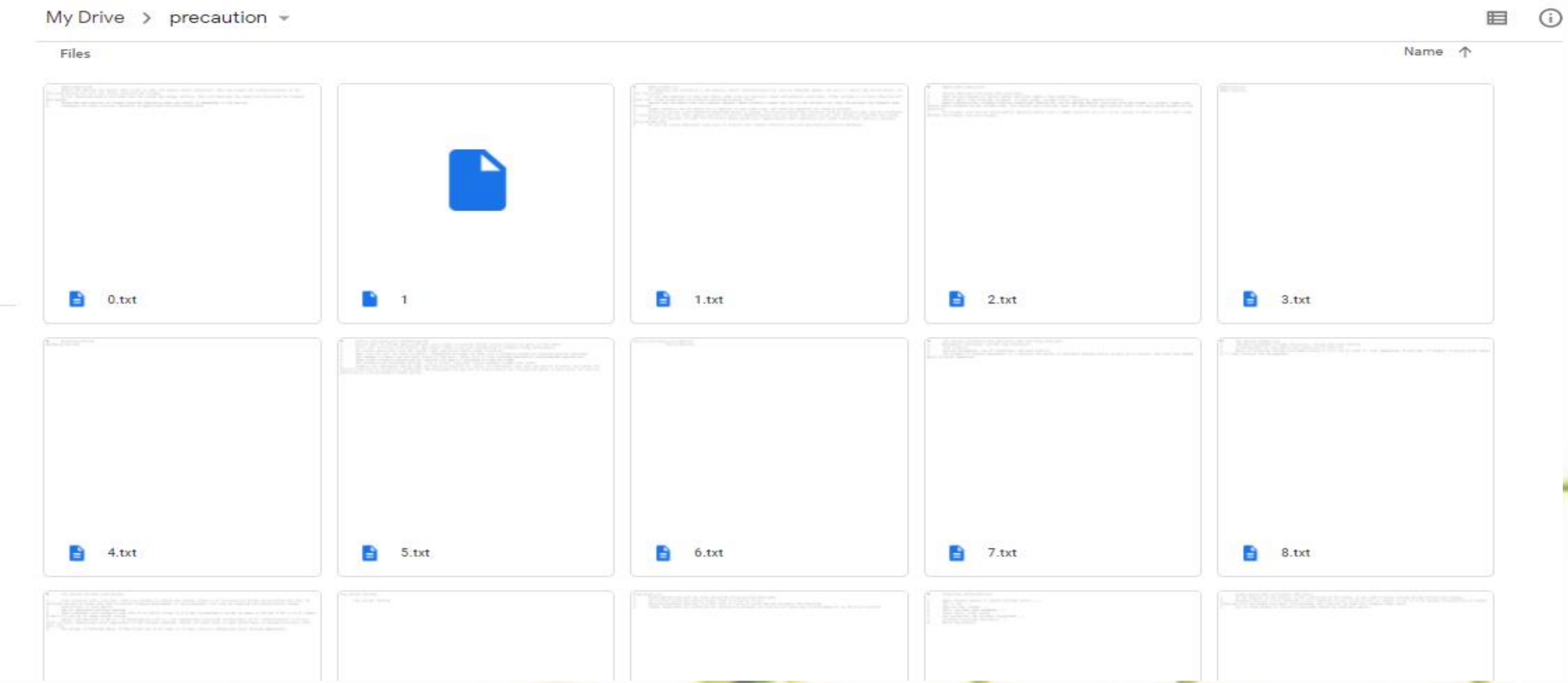


Figure 13: Plant Monitoring System – Derived Precautions

Contd..

Apple_Apple_Scab

1. Remove and destroy the fallen leaf litter so that the fungus cannot overwinter. This may reduce the disease pressure in the following spring, but will not likely eliminate the disease.
2. Do not overcrowd plants, and make sure the canopy has proper airflow. This will decrease the conditions favorable for disease development.
3. Fungicide applications at 2-week intervals beginning when new growth is expanding in the spring.
4. Crabapple cultivars that are resistant to apple scab are widely available

Figure 14: Plant Monitoring System – Derived Precautions (Apple plant)

- The Plant Growth Monitoring System using AI process and here is the progress of our project, we have collected the dataset of few plants from the trained using the machine learning algorithm and here is output of the software part.
- Here we have used two types of data set one is Train data and other is Validation data. Train data is for training and validation is for validation of the train data.

Contd..

- The dataset contains the different 3- dimensional plant images to train and classify the plant as healthy and unhealthy.
- Above image (Figure 7) show the plant growth monitoring system. Ipynb file. The first two lines of code that is:

```
from google.colab import drive
drive.mount('/content/drive')
```

- This means we are mounting the google drive to our python file. So that we can access our dataset which is present in the drive. Here the dataset present in the drive is in the zip folder so we are unzipping the folder to train our data.
- Figure 8 and 9 shows the dataset of tomato plant. Figure 8 is a train data used for training the model and Figure 9 is the valid data used for validation of the rain data.
- Figure 10 shows the importing of libraries and preprocessing of dataset. We have used NumPy, matplotlib, keras, pandas and few other libraries.

Contd..

- Figure 11 and 12 shows the classification of plant as healthy and unhealthy using Machine learning algorithm and convolution neural network (CNN). If the plant is identified as unhealthy, then we are going to specify the necessary precautions for that disease. The Derived precaution are shown in the figure 13 and 14.

- Software working video link:
- https://drive.google.com/file/d/1FHi3X20fjGPC3A6FAxgtwpOBMUsnfOvU/view?usp=sharing

Contd..

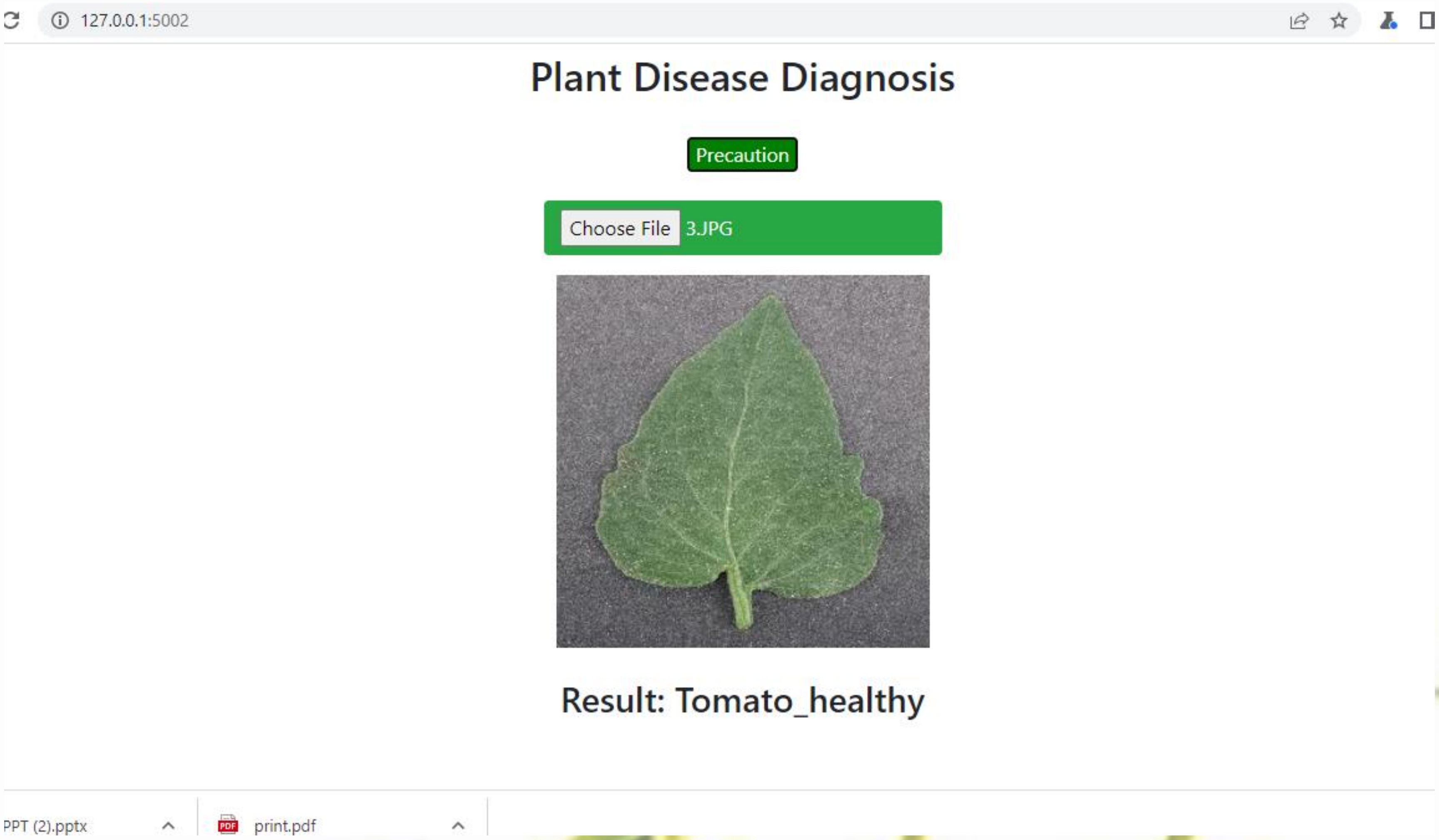


Figure 14: Plant Monitoring System – WebPage

Contd..

127.0.0.1:5002

Plant Disease Diagnosis

Precaution

Disease name	Precaution to Be given
Tomato_Late_blight	Apply a copper based fungicide (2 oz/ gallon of water) every 7 days or less, following heavy rain or when the amount of disease is increasing rapidly
Tomato_Septoria_leaf_spot	1. Removal and destruction of the affected plant parts. 2. Seed treatment with Thiram or Dithane M-45 (2 g/kg seed) is useful in checking seed borne infection. 3. In the field spraying with Mancozeb 0.2 % effectively controls the disease.
Tomato_Leaf_Mold	1. Scout for tomato leaf mold during periods of high humidity (over 85%). 2. Optimal temperature is between 71 °F and 75 °F, but disease can occur at temperatures as low as 50 °F and as high as 90 °F. 3. The first leaf mold infections of the season have been observed in the first week of June in Minnesota high tunnel tomatoes. 4. Stake, string or prune to increase airflow in and around the plant. 5. Sterilize stakes, ties, trellises, etc. with 10% household bleach or commercial sanitizer. 6. Circulate air in greenhouses or tunnels with vents and fans and by rolling up high tunnel sides to reduce humidity around plants.
	1. Disease-free seed and seedlings should always be used and the crop should be rotated with non-host crops so as to avoid last years crop residue. 2. Seed treatment with mercuric chloride (1:1000) is also

Figure 14: Plant Monitoring System – WebPage

ADVANTAGES

- It helps in managing irrigation systems more effectively and efficiently.
- It helps farmers to increase yields and to increase quality of the crop.

CONCLUSION

- The proposed system helps in identification of the plant disease and provides remedies that can be used as a defense mechanism against the disease.
- The database obtained from the internet is properly segregated and the different plant species are identified and are renamed to form a proper database then obtain test database which consists of various plant diseases that are used we will train our classifier and then output will be predicted with optimum accuracy.
- We use Convolution Neural Network (CNN) which comprises of different layers which are used for prediction.
- High resolution camera is attached and will capture images of the plants which will act as input for the software, based of which the software will tell us whether the plant is healthy or not.

REFERENCES

- Monirul Islam Pavel, Syed Mohammad Kamruzzaman, Sadman Sakib Hasan, “An IoT Based Plant Health Monitoring System Implementing Image Processing”, Volume-5, Nov-2019.
- Rohit Nalawade, Apoorv Nagap, Lakhan Jindam, “Agriculture field monitoring and Plant Leaf Disease Detection”, Volume-10, Issue-6, 2020.
- Siddharth Singh Chouhan, Ajay Kaul, Uday Pratap Singh, “A deep learning approach for the classification of diseased plant leaf images”, Volume-6, Issue-9, 2019.
- Yingying Dong, G=Fang Xu, “Monitoring and forecasting for disease and pest in crop based on WebGIS system”, Volume-2, Issue-10, 2019.
- Wahyu Saputra and Rudiati Evi Masithoh, “Development of Growth Monitoring system using Image Processing Technique based on Multiple Image”, Vol. 2, No. 5, January 2017,
- Gaurav Patil, Shashank Pathmudi and Akash Patil, “Plant Monitoring System”, Volume 10, Issue 09, September 2021, I.S.Akila, A.Shivakumar, S.Swaminathan, “Automation in plant growth monitoring using high precision image classification and virtual height measurement techniques”, Volume 10, February 2018.

Q & A…


Thank You!