GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

LEAF DISEASE DETECTION USING IMAGE PROCESSING

A Report of Capstone Project-2

Submitted by

SHIVAM KUMAR MODANWAL

(1613101691 / 16SCSE101297)

in partial fulfillment for the award of the degree

of

Bachelor of Technology

IN

Computer Science and Engineering

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Under the Supervision of

Sreenarayanan N M

Assistant Professor

APRIL/MAY-2020

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report "LEAF DISEASE DETECTION USING IMAGE PROCESSING" is the bonafide work of "SHIVAM KUMAR MODANWAL (1613101691)" who carried out the project work under my supervision.

SIGNATURE OF HEAD

Dr. MUNISH SHABARWAL,
PhD (Management), PhD (CS)
Professor & Dean,
School of Computing Science & Engineering

SIGNATURE OF SUPERVISIOR

MR. SREENARAYNAN N M
Assistant professor
School of Computing Science & Engineering

ABSTRACT

Agriculture is the backbone of the Indian economy. Almost 65% of people rely upon it & shares major part of the GDP. Disease detection plays a vital role in prevention of quantity and quality of the agricultural product. Leaf shows the symptoms by changing color and showing the spots on it. Earlier methodology was manual observation of leaf for diseases that takes time and was costly. This paper proposes a software solution to automatically detect and classify plant leaf diseases by image processing of leaf images. My aim is to detect the disease in leaf at an early stage and thus take suitable measures to stop this disease. In this we tend to area unit using image processing techniques to classify diseases & quickly diagnosis can be carried out as per disease. This approach will enhance productivity of crops. It includes several steps viz. digital image acquisition, image pre-processing, segmentation, features extraction and neural network-based classification. The diseases considered in this are Late blight, Yellow leaf curl virus and bacterial spot. Aim of my project is to detect the plant diseases and provide solutions to recover from the disease. This software system also shows remedies for the disease.

LIST OF TABLES PAGE NO.

LIST OF FIGURES

TABLE OF CONTENT

1. INTRODUCTION

Agriculture played key role in development of India economy. If there is decrease in farming, then total economy of a country will get affected. In today era farmers are facing many problems regarding fungal infection or bacterial disease. The causes of leaf disease due to climate changes or due to soil and air pollution causes by humans. In our country around 50% of farmer are illiterate and poor, they do not know which pesticides used to save the crop form disease and the plant infection impact the people directly or financially.

The plant disease become important factor which causes significant reduction in the quality and quantity of the crops. The detection and classification of disease is important task in plant disease detection, here we proposed a novel method of the using the CNN algorithm for the detection and provide remedies. Our naked eyes can't detect the which fertilizer used to prevent the crop from disease, so we use computer-based programming data to detect the disease and provide remedies. Plant disease diagnosis is extremely essential in earlier stage in order to prevent and control them, using computer programming system it is easy to detect the disease for larger area and it provides exact detection of disease and remedies because if this process is check by our naked eyes it takes more time and cost to detect the disease and may be not accurately.

As you know small farmer borrow money to purchase basic things like seed, fertilisers and pesticides. Often they borrow money from moneylenders. If the crop get effected and they use pesticides without knowing the proper causes of disease, the plant leaf get damaged due to which

the quality and quantity of the crop not good. Due to which they can't return money to the moneylender which causes farmers comes to suicide.

The proposed software takes input a leaf image and based on CNN algorithm it classifies leaf into (1). Healthy (2). Bacterial Disease (3). Yellow leaf curl virus (4). Late Blight and also provide remedies based on the classified disease.

(a). Healthy leaf

(b). cterial Disease

(c). Yellow leaf curl virus

(d). Late Blight

Fig. 1 shows the images of disease and also healthy leaf

Convolutional neural network (CNN) has a popular method of target detection, has a wide application prospect in the field of crop disease detection. As a kind of machine learning CNN can achieve the purpose of accurate detection by training a large number of images. CNN does not depend on specific feature and has a good detection effect in the field of generalized identification such as target detection, target segmentation and target recognition.

Recently researchers have started applying machine learning methods for plant disease detection. Ramesh et al. extracted the histogram of an oriented gradient feature of the crop image and classified by a random forest, they only classified the plant images by the handcrafted extracting feature without detection.

2. LITERATURE SURVEY

In [1] for identification and classification of leaf disease they used classical neural network algorithm. In this the efficiency of the network by searching and grouping of seeds point having common attribute for feature extraction process. The proposed method helpful to the agriculture sector and attains higher accuracy in identification and classification of leaf disease.

In [2] the image recognition system based on multiple linear regression is done. In image segmentation an improved histogram segmentation method which calculate the threshold value automatically and accurately is proposed. In between this, the regional growth method and the true color image processing are combined with this system to improve the accuracy, while creating the recognition system multiple linear regression and image feature extraction are utilized. After evaluation the result is shown.

In [3] present an algorithm for image segmentation techniques which is used for automatic detection and classification of plant leaf disease. Image segmentation, which is important for disease detection in plant leaf is done by using genetic algorithm. Banana, beans, jackfruit, lemon, mango, potato, tomato and sapota leaf on which proposed algorithm is tested.

In [4] analysis of content based image retrieval for plant leaf disease using color, shape and texture features.

This paper is an attempt to present Content Based Image Retrieval (CBIR) system developed for retrieving disease leaves of soybean.

In [5] the captured image grapes leaf is first preprocessed to resize it and then converted to HSI color space format by using segmentation. The feature such as major axis, minor axis, eccentricity extracted from the image. In last step, the feature given to the classifier to find the disease on leaf. In this segmentation work on two principles (1). Discontinuity (2). Similarity. Discontinuity extract the regions having different properties like texture, color, intensity etc. and similarity groups the image pixels into groups with some predefined area.

In [6] includes four main phases; first phase we create a color transformation structure for the RGB leaf image and then, we apply color space transformation for the color transformation structure, after that image is segmented using the K-means clustering technique. Second phase, unnecessary part (green area) within leaf are removed. Third phase we calculate the texture features for the segmented infected object. Finally, last phase the extracted features are passed through a pre-trained neural network.

In [7] involves 5 steps to identification of disease which are image acquisition, image preprocessing, image segmentation, features extraction, detection and classification in plant disease. the study and analysis are done on cotton leaf. K-mean clustering algorithm and SVM are used in this system.

In [8] uses ANN classification techniques for image disease detection. In this paper there are five steps, first a color transformation structure for the input RGB image is created, after that the unnecessary part is removed using threshold value, then image is segmented with connected component labeling and useful segments are extracted, at last ANN classification used.

3. PROPOSED SYSTEM

Our aim of the project is to detect the plant diseases and provide the solutions to recover from the leaf diseases. We planned to design our project with digital image processing so that a person with lesser expertise in software should also be able to use it easily. In our system we are providing a solution to recover from the leaf diseases and also show the affected part of the leaf by image processing technique. The existing system only identify the type of diseases which affects the leaf. We will provide a result within seconds and guided you throughout the project. In this we have trained data sets and training data sets. Trained data sets which contain number of different infected leaf. Samples of images are collected that comprised of different leaf diseases like Alternaria Alternata, Anthracnose, Bacterial Blight, Cercospora leaf spot and Healthy Leaves. The main objective is to identify the plant diseases using image processing. The remedy of disease is shown by the system after the identification of the diseases.

CNN how works in Classification of Images-

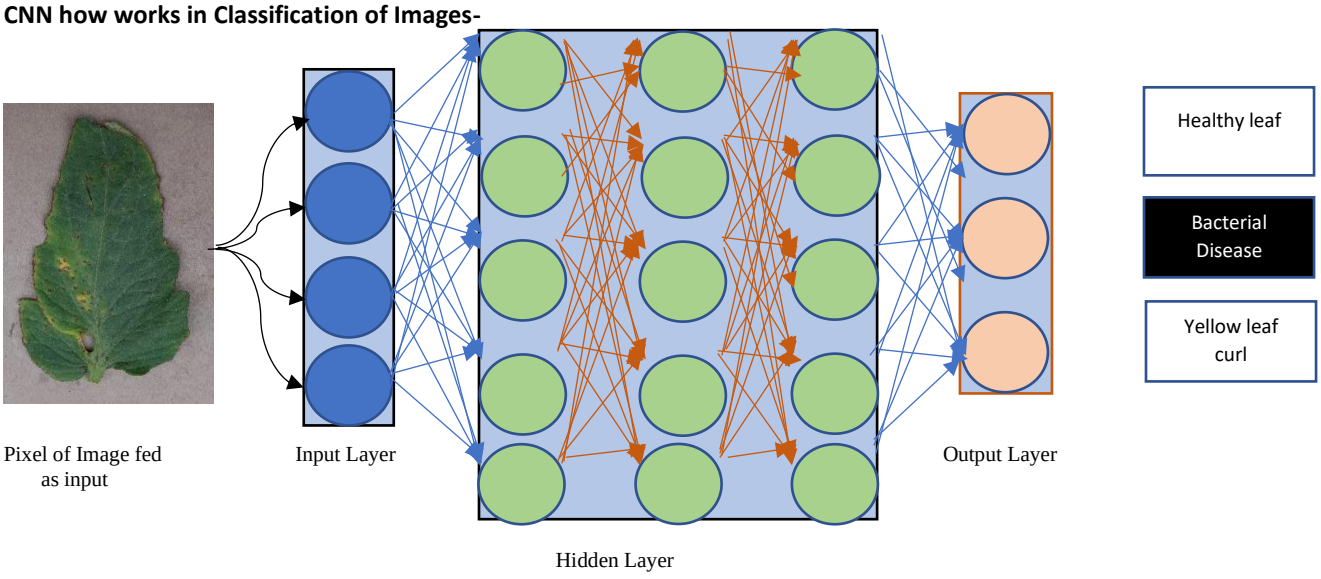


Fig.2 Model of CNN Network

1. Input layer-In the above figure Pixel of image fed as input then the image given in input layer where it accepts the pixel of the image as input in the form of array.

2. Hidden Layer -Hidden layers carry out feature extraction by performing certain calculation and manipulation.

- **Convolution layer-** This layer uses a matrix filter and performs convolution operation to detect patterns in the image.
- **ReLU**- ReLU activation function is applied to the convolution layer to get a rectified feature map of image
- **Pooling**- Pooling layer also uses multiple filters to detect edges, corners, features etc.

3. Output Layer-Finally there is fully connected layer that identifies the object in the image.

The proposed system takes input image for testing after that pre-processing of image are carried out. Input image are converted to an array for compression of image according to image size given. On other side segregated database are created, in this also a pre-processing of data is done. Image form the segregated database are classified using CNN classification. Input image after converted to an array after compression, CNN based classification are applied to the trained dataset (segregated database) and testing dataset (input image)

and finally defect of the leaf are display. If defect is found display disease and Remedy otherwise if defect not found then leaf is Healthy.

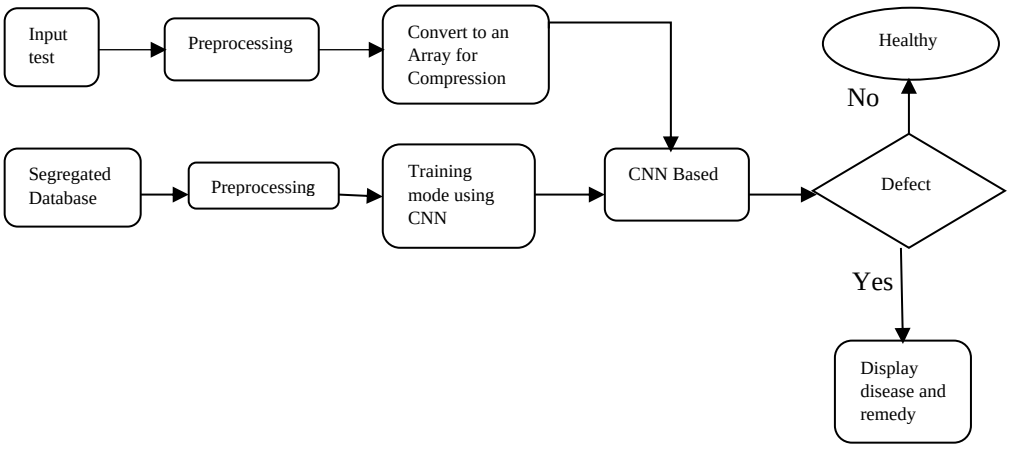
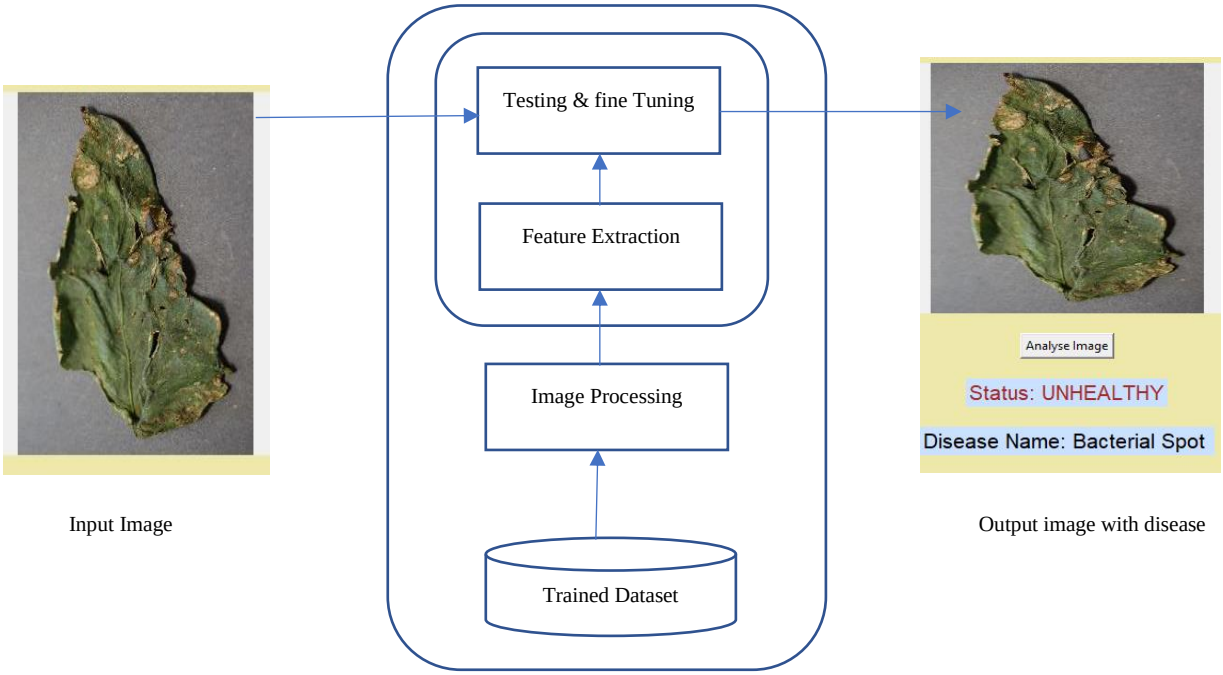


Fig. 3 Flow chart of leaf disease detection



Input Image

Output image with disease

Fig. 4 Architecture of Plant Disease Detection System

6. IMPLEMENTATION

CNN.py

```
import cv2
import numpy as np
import os
from random import shuffle
from tqdm import tqdm
TRAIN_DIR = 'train/train'
TEST_DIR = 'test/test'
IMG_SIZE = 50
LR = 1e-3
MODEL_NAME = 'healthyvsunhealthy-{}-{}.model'.format(LR, '2conv-basic')

def label_img(img):
    word_label = img[0]

    if word_label == 'h': return [1,0,0,0]

    elif word_label == 'b': return [0,1,0,0]
    elif word_label == 'v': return [0,0,1,0]
    elif word_label == 'l': return [0,0,0,1]

def create_train_data():
    training_data = []
    for img in tqdm(os.listdir(TRAIN_DIR)):
        label = label_img(img)
        path = os.path.join(TRAIN_DIR,img)
```

```
        img = cv2.imread(path,cv2.IMREAD_COLOR)
        img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))
        training_data.append([np.array(img),np.array(label)])
    shuffle(training_data)
    np.save('train_data.npy', training_data)
    return training_data

def process_test_data():
    testing_data = []
    for img in tqdm(os.listdir(TEST_DIR)):
        path = os.path.join(TEST_DIR,img)
        img_num = img.split('.')[0]
        img = cv2.imread(path,cv2.IMREAD_COLOR)
        img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))
        testing_data.append([np.array(img), img_num])

    shuffle(testing_data)
    np.save('test_data.npy', testing_data)
    return testing_data

train_data = create_train_data()
#train_data = np.load('train_data.npy')



import tflearn
from tflearn.layers.conv import conv_2d, max_pool_2d
from tflearn.layers.core import input_data, dropout, fully_connected
from tflearn.layers.estimator import regression
import tensorflow as tf
```

```
tf.reset_default_graph()

convnet = input_data(shape=[None, IMG_SIZE, IMG_SIZE, 3], name='input')

convnet = conv_2d(convnet, 32, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)

convnet = conv_2d(convnet, 64, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)

convnet = conv_2d(convnet, 128, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)

convnet = conv_2d(convnet, 32, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)

convnet = conv_2d(convnet, 64, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)

convnet = fully_connected(convnet, 1024, activation='relu')
convnet = dropout(convnet, 0.8)

convnet = fully_connected(convnet, 4, activation='softmax')
convnet = regression(convnet, optimizer='adam', learning_rate=LR, loss='categorical_crossentropy',
name='targets')

model = tflearn.DNN(convnet, tensorboard_dir='log')

if os.path.exists('{}.meta'.format(MODEL_NAME)):
    model.load(MODEL_NAME)
```

```
    print('model loaded!')

train = train_data[:-500]
test = train_data[-500:]

X = np.array([i[0] for i in train]).reshape(-1,IMG_SIZE,IMG_SIZE,3)
Y = [i[1] for i in train]

test_x = np.array([i[0] for i in test]).reshape(-1,IMG_SIZE,IMG_SIZE,3)
test_y = [i[1] for i in test]

model.fit({'input': X}, {'targets': Y}, n_epoch=8, validation_set=({'input': test_x}, {'targets': test_y}),
    snapshot_step=40, show_metric=True, run_id=MODEL_NAME)

model.save(MODEL_NAME)
```

UI.py

```
import tkinter as tk
from tkinter.filedialog import askopenfilename
import shutil
import os
import sys
from PIL import Image, ImageTk

window = tk.Tk()

window.title("Plant Disease Detection using Image Processing")

window.geometry("500x510")
```

```
window.configure(background ="palegoldenrod")


title = tk.Label(text="Click below to choose picture for testing disease....", background = "palegoldenrod",
fg="royalblue", font=("", 15))

title.grid()

def bact():

    window.destroy()

    window1 = tk.Tk()


    window1.title("Plant Disease Detection using Image Processing")


    window1.geometry("500x510")

    window1.configure(background="palegoldenrod")


    def exit():

        window1.destroy()

    rem = "The remedies for Bacterial Spot are:\n\n "

    remedies = tk.Label(text=rem, background="lightyellow",

                        fg="Brown", font=("", 15))

    remedies.grid(column=0, row=7, padx=10, pady=10)

    rem1 = " Discard or destroy any affected plants. \n  Do not compost them. \n  Rotate yoour tomato plants
yearly to prevent re-infection next year. \n Use copper fungicites"

    remedies1 = tk.Label(text=rem1, background="lightyellow",

                         fg="Black", font=("", 12))

    remedies1.grid(column=0, row=8, padx=10, pady=10)


    button = tk.Button(text="Exit", command=exit)

    button.grid(column=0, row=9, padx=20, pady=20)


    window1.mainloop()
```

```
def vir():
    window.destroy()
    window1 = tk.Tk()

    window1.title("Plant Disease Detection using Image Processing")

    window1.geometry("650x510")
    window1.configure(background="palegoldenrod")

    def exit():
        window1.destroy()
    rem = "The remedies for Yellow leaf curl virus are: "
    remedies = tk.Label(text=rem, background="lightyellow",
                        fg="Brown", font=("", 15))
    remedies.grid(column=0, row=7, padx=10, pady=10)
    rem1 = " Monitor the field, handpick diseased plants and bury them. \n  Use sticky yellow plastic traps. \n
Spray insecticides such as organophosphates, carbametes during the seedliing stage. \n Use copper fungicites"
    remedies1 = tk.Label(text=rem1, background="lightyellow",
                         fg="Black", font=("", 12))
    remedies1.grid(column=0, row=8, padx=10, pady=10)

    button = tk.Button(text="Exit", command=exit)
    button.grid(column=0, row=9, padx=20, pady=20)

    window1.mainloop()

def latebl():
    window.destroy()
```

```
window1 = tk.Tk()


window1.title("Plant Disease Detection using Image Processing")


window1.geometry("520x510")
window1.configure(background="palegoldenrod")


def exit():
    window1.destroy()
rem = "The remedies for Late Blight are: "
remedies = tk.Label(text=rem, background="lightyellow",
     fg="Brown", font=("", 15))
remedies.grid(column=0, row=7, padx=10, pady=10)
 rem1 = " Monitor the field, remove and destroy infected leaves. \n  Treat organically with copper spray. \n  Use chemical fungicides,the best of which for tomatoes is chlorothalonil."
remedies1 = tk.Label(text=rem1, background="lightyellow",
                     fg="Black", font=("", 12))
remedies1.grid(column=0, row=8, padx=10, pady=10)
button = tk.Button(text="Exit", command=exit)
button.grid(column=0, row=9, padx=20, pady=20)
 window1.mainloop()
def analysis():
import cv2
import numpy as np
import os
from random import shuffle
from tqdm import \
    tqdm
verify_dir = 'testpicture'
IMG_SIZE = 50
```

```
LR = 1e-3
MODEL_NAME = 'healthyvsunhealthy-{}-{}.model'.format(LR, '2conv-basic')

def process_verify_data():
    verifying_data = []
    for img in tqdm(os.listdir(verify_dir)):
        path = os.path.join(verify_dir, img)
        img_num = img.split('.')[0]
        img = cv2.imread(path, cv2.IMREAD_COLOR)
        img = cv2.resize(img, (IMG_SIZE, IMG_SIZE))
        verifying_data.append([np.array(img), img_num])
    np.save('verify_data.npy', verifying_data)
    return verifying_data

verify_data = process_verify_data()
verify_data = np.load('verify_data.npy')

import tflearn
from tflearn.layers.conv import conv_2d, max_pool_2d
from tflearn.layers.core import input_data, dropout, fully_connected
from tflearn.layers.estimator import regression
import tensorflow as tf
tf.reset_default_graph()

convnet = input_data(shape=[None, IMG_SIZE, IMG_SIZE, 3], name='input')
 convnet = conv_2d(convnet, 32, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)
 convnet = conv_2d(convnet, 64, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)
```

```
 convnet = conv_2d(convnet, 128, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)
 convnet = conv_2d(convnet, 32, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)
 convnet = conv_2d(convnet, 64, 3, activation='relu')
convnet = max_pool_2d(convnet, 3)
convnet = fully_connected(convnet, 1024, activation='relu')
convnet = dropout(convnet, 0.8)
convnet = fully_connected(convnet, 4, activation='softmax')
convnet = regression(convnet, optimizer='adam', learning_rate=LR, loss='categorical_crossentropy',
name='targets')
 model = tflearn.DNN(convnet, tensorboard_dir='log')
  if os.path.exists('{}.meta'.format(MODEL_NAME)):
    model.load(MODEL_NAME)
    print('model loaded!')
import matplotlib.pyplot as plt
 fig = plt.figure()
 for num, data in enumerate(verify_data):
      img_num = data[1]
    img_data = data[0]
    y = fig.add_subplot(3, 4, num + 1)
    orig = img_data
    data = img_data.reshape(IMG_SIZE, IMG_SIZE, 3)
        model_out = model.predict([data])[0]
         if np.argmax(model_out) == 0:
        str_label = 'healthy'
    elif np.argmax(model_out) == 1:
        str_label = 'bacterial'
    elif np.argmax(model_out) == 2:
        str_label = 'viral'
```

```
elif np.argmax(model_out) == 3:
    str_label = 'lateblight'
      if str_label =='healthy':
    status ="HEALTHY"
else:
    status = "UNHEALTHY"
     message = tk.Label(text='Status: '+status, background="lightsteelblue1",
                fg="Brown", font=("", 15))
message.grid(column=0, row=3, padx=10, pady=10)
if str_label == 'bacterial':
    diseasename = "Bacterial Spot "
    disease = tk.Label(text='Disease Name: ' + diseasename, background="lightsteelblue1",
                    fg="Black", font=("", 15))
    disease.grid(column=0, row=4, padx=10, pady=10)
    r = tk.Label(text='Click below for remedies...', background="lightsteelblue1", fg="Brown", font=("", 15))
    r.grid(column=0, row=5, padx=10, pady=10)
    button3 = tk.Button(text="Remedies", command=bact)
    button3.grid(column=0, row=6, padx=10, pady=10)
elif str_label == 'viral':
    diseasename = "Yellow leaf curl virus "
    disease = tk.Label(text='Disease Name: ' + diseasename, background="lightsteelblue1",
                    fg="Black", font=("", 15))
    disease.grid(column=0, row=4, padx=10, pady=10)
    r = tk.Label(text='Click below for remedies...', background="lightsteelblue1", fg="Brown", font=("", 15))
    r.grid(column=0, row=5, padx=10, pady=10)
    button3 = tk.Button(text="Remedies", command=vir)
    button3.grid(column=0, row=6, padx=10, pady=10)
elif str_label == 'lateblight':
    diseasename = "Late Blight "
```

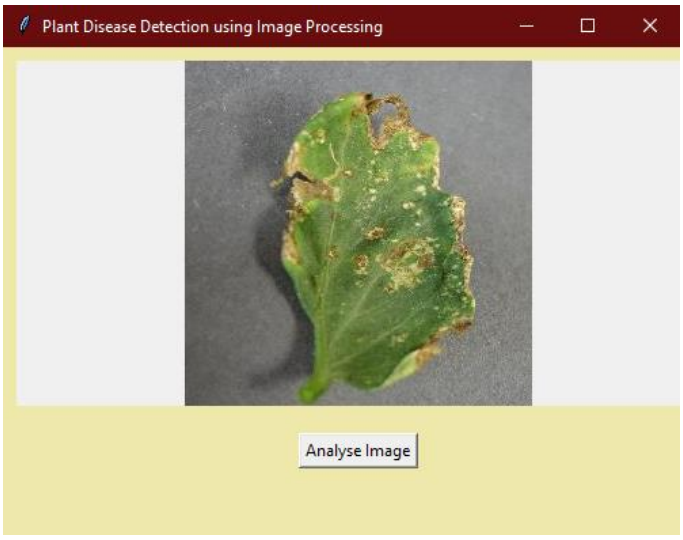
```
        disease = tk.Label(text='Disease Name: ' + diseasename, background="lightsteelblue1",
                           fg="Black", font=("", 15))
            disease.grid(column=0, row=4, padx=10, pady=10)
            r = tk.Label(text='Click below for remedies...', background="lightsteelblue1", fg="Brown", font=("", 15))
            r.grid(column=0, row=5, padx=10, pady=10)
            button3 = tk.Button(text="Remedies", command=latebl)
            button3.grid(column=0, row=6, padx=10, pady=10)
        else:
            r = tk.Label(text='Plant is healthy', background="lightsteelblue1", fg="Black",
                         font=("", 15))
            r.grid(column=0, row=4, padx=10, pady=10)
            button = tk.Button(text="Exit", command=exit)
            button.grid(column=0, row=9, padx=20, pady=20)

def openphoto():
    dirPath = "testpicture"
    fileList = os.listdir(dirPath)
    for fileName in fileList:
        os.remove(dirPath + "/" + fileName)

    fileName = askopenfilename(initialdir='I:\PlantDiseaseDetection-master', title='Select image for analysis ',
                               filetypes=[('image files', '.jpg')])

    dst = "testpicture"
    shutil.copy(fileName, dst)
    load = Image.open(fileName)
    render = ImageTk.PhotoImage(load)
    img = tk.Label(image=render, height="250", width="500")
    img.image = render
```

```
    img.place(x=0, y=0)
    img.grid(column=0, row=1, padx=10, pady = 10)
    title.destroy()
    button1.destroy()
    button2 = tk.Button(text="Analyse Image", command=analysis)
    button2.grid(column=0, row=2, padx=10, pady = 10)
button1 = tk.Button(text="Get Photo", command = openphoto)
button1.grid(column=0, row=1, padx=10, pady = 10)
window.mainloop()
```

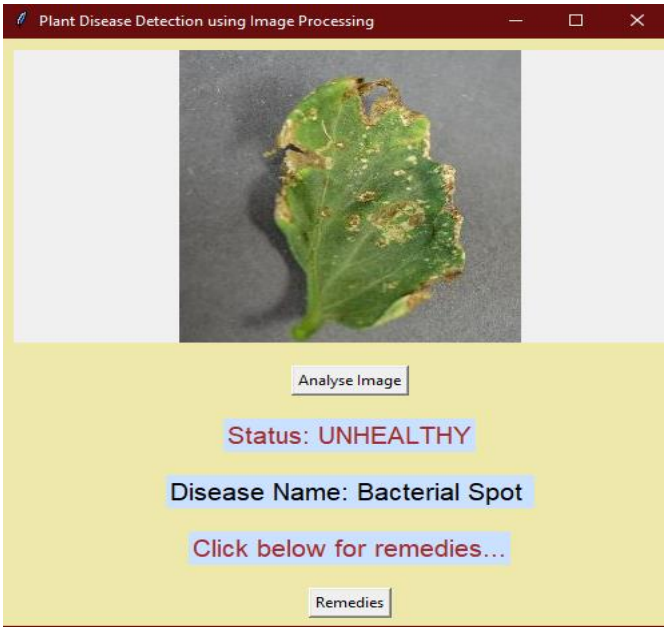
5. OUTPUT

I. Start Screen

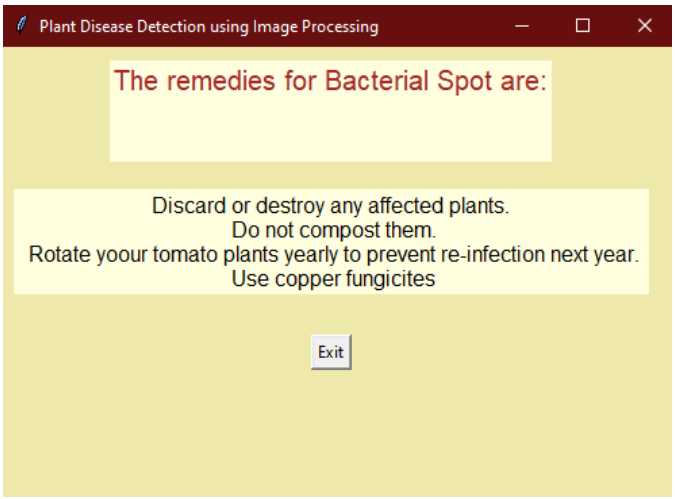
II. Analysis screen



III. Disease Detection Window



IV. Remedies



6. RESULT ANALYSIS

In earlier stage people uses naked eyes to detect the disease which were not tell the correct disease, after hat some of the researcher use different-different Algorithm some paper uses K-means clustering, infection region or some uses shape ,size ,texture of the leaf image to compare with the original data, another paper uses contrast enhancement, HSI, SVM, and take the input image and change into RGB after that disease will be shown. Another paper uses some threshold value to show the image disease, and latest paper uses classical Neural Network to detect the disease of leaf but none of the paper shows the remedy of infected leaf.

In my paper there are availability of trained datasets which help to classify with the given input images using CNN classification and shows the remedy of leaf disease. The proposed system gives an accuracy of around 94.8%.

S. No.	*Literature Survey*	*Efficiency in %*
1.	Paper 1	78
2.	Paper 2	85
3.	Paper 3	91
4.	Paper 4	82
5.	Paper 5	72
6.	Paper 6	88
7.	Paper 7	92
8.	Paper 8	68

Tab.1 shows the efficiency

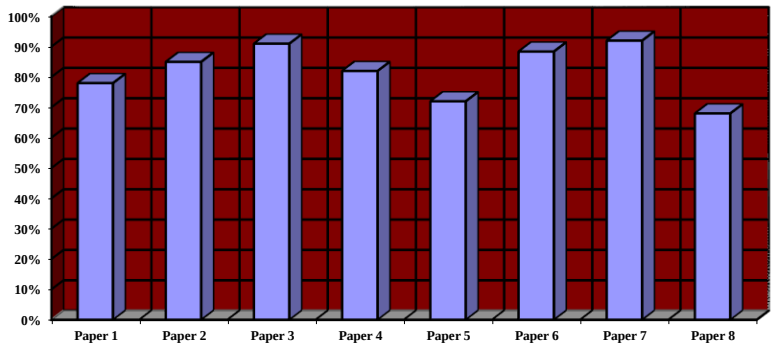


Fig.5 shows the graph of previous work done

7. CONCLUSION

The study summarizes image processing techniques for several plant species that have been used for recognizing leaf disease. In this we use CNN techniques to classifies the disease on given trained data set. Some of the challenges in these techniques are optimization of the techniques for a specific plant. The proposed system was developed taking in mind the benefits of the farmers and agriculture sector. With the help of proper knowledge of the disease and the remedy can be taken for improving the health of the plant.

Using CNN Algorithm it is easy to detect the disease and using proper use of pesticides farmer gives a proper treat to the crops and saves them, which causes increase in the agriculture of our country and help to increase the GDP. Some of the new things will also add in this in future which helps to find disease very easily. The proposed system based on python and gives an accuracy of around 94.8%. The accuracy and the speed can be increased by use of Googles GPU for processing. The system can be installed on Drones so that aerial surveillance of crop field can be done.

8. REFERENCES

1. Monishankar Halder, Ananya Sarkar, ssHabibullah Bahar.” Plant disease detection by using image processing” is published on 7th Feb 2019, Department of Computer Science and Engineering, Jashore University of Science and Technology,Jashore-7408.
2. Guiling Sun,Xinglong,Jia, and Tianyu Geng, “Plant Disease Recognition Based on Image Processing Technology” published on 21 May 2018. Nankai University, College of Electronics Information and Optional Engineering, Tianjin 300350, China. Article ID 6070129.
3. Veena Jayasree Krishnan, Ganesh A Balaji,” Identification and Classification of Plant Leaf Disease Using Neural Networks” published on 2019.
4. S Megha, R. Niveditha, N Sowmyashree, K.Vidya,” Image Processing System for Plant Disease Identification by Using FCM-Clstering Technique”. 2017
5. Vijai Singh[a], A.K Mishra[b],” Detection of plant leaf disease using Image Segmentation and Soft computing techniques”, Computer Science Department, IMS Engineering College, Ghaziabad, UP, India. Available online 4 November 2016.
6. Megha P. Arakeri, Malavika Arun, K PadminiR,” Analysis of Late Blight Disease in Tomato Leaf Using Image Processing Technicques” published on 2015, vol 2. Department of Computer Science.
7. Jayamala K. Patil, Raj Kumar,” Analysis of content based image retrieval for plant leaf disease using colour, shape ,texture features ”.Published on 2017. Department of Mathematics.
8. Basavaraj Tigadi, Blanka Sharma,” Banana Plant Disease Detection and Grading Using Imahe Processing”.2016

9. Prakash M. Mainkar, Shreekant Ghorpade, Mayur Adawadkar,"Plant Leaf Detection and Classification Using Image Procesing Techniques". Department of Electronics & Telecommunication, MAEER's Maharashtra Instute of Technology, Pune, India.2015
10. Saradhambal. G, Dhivya.R, Latha.S, R.Rajesh," Plant Disease Detection and its Solution using Image Classification ", On International Journal of Pure and Applied Mathematics. Department of CSE, IFET College of Engineering and Technology,Villupuram, India.2018
11. Monica Jhuria, Ashwani Kumar, and RushikeshBorse, "Image Processing For Smart Farming: Detection Of Disease", Proceedings of the 2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)
12. Wenjiang Huang, Qingsong Guan, JuhuaLuo, Jingcheng Zhang, Jinling Zhao, Dong Liang,Linsheng Huang, and DongyanZhang, "New Optimized Spectral Indices for Identifying Winter Wheat Diseases", IEEE journal of selected topics in applied earth observation and remote sensing, Vol. 7, No. 6, June 2014
13. 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.
14. Hrishikesh P. Kanjalkar, S. S. Lokhande," Detection and Classification of Plant Leaf disease using ANN". Issued on 2013, Department of Engineering.
15. Reena Tijare, Pawan Khade, Rashmi Jain,"The Survey of Disease Identification of Cotton Leaf", International Journal of Innovative Research in Computer and Communication Engineering, 2015.