

DETECTION AND IDENTIFICATION OF PLANT DISEASE BASED ON PYTHON

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Arts and Science Degree in Computer Science



DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

JEPPIAAR NAGAR, RAGIV GANDHI SALAI,

CHENNAI – 600119, TAMILNADU

MARCH 2021



SATHYABAMA UNIVERSITY
(Established under Section 3 of UGC Act, 1956)
(A CHRISTIAN MINORITY INSTITUTION)
Declared as Category "A" University by MHRD, Government of India
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119.



DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of B.NITHYAKALYANI ALAIS NITHISHA (Reg. No.38290058) and H.KIRTHIKA (Reg.No.38290038). Who carried out the project entitled "DETECTION AND IDENTIFICATION OF PLANT DISEASES BASED ON PYTHON" under our supervision from November 2020 to March 2021.

Internal Guide

Mr.A.YOVAN FELIX

Head of the Department

Dr.L.LAKSHMANAN, M.E.,Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

B.NITHYAKALYANI ALAIS NITHISHA (Reg. No.38290058) hereby declare that the project Report entitled "DETECTION AND IDENTIFICATION OF PLANT DISEASES BASED ON PYTHON" done by me under the guidance of Mr. A. YOVAN FELIX is submitted in partial fulfillment for the award of Bachelor of Arts and Science in Computer Science.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr.T.SASIKALA,M.E.,Ph.D., Dean, School of Computing and Dr.S.VIGNESHWARI,M.E.,Ph.D., and Dr.L.LAKSHMANAN,M.E.,Ph.D., Head of the Department, Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Mr. A. YOVAN FELIX for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in the Completion of the project.

ABSTRACT

Plant disease is an ongoing challenge for smallholder farmers, which threatens income and food security. The recent revolution in smartphone penetration and computer vision methods has created an opportunity for image classification in agriculture. Convolutional Neural Networks(CNN) are considered state of the art in image recognition and offer the ability to provide a prompt and definite diagnosis. In this paper, the performance of a pre-trained ResNet34 model in detecting crop disease is investigated.

The developed model is deployed as a web application and is capable of recognizing 7 plant diseases out of healthy leaf tissue.

A dataset containing 8,685 leaf images, captured in a controlled environment, is established for training and validating the model. Validation result show that the proposed method can achieve an accuracy of 97.2% and an F1 score of greater than 96.5%. This demonstrates the technical feasibility of CNNs in classifying plant diseases and presents a path towards AI solutions for small holder farmers.

The project focuses on the approach based on image processing for detection of diseases of plants. In this paper, we propose an Android application that helps farmers for identifying plant disease by uploading a leaf image to the system. The system has a set of algorithms which can identify the type of disease. Input image given by the user undergoes several processing steps to detect the disease and results are returned back to the user via android application.

TABLE OF CONTENTS

Chapter No.	TITLE	PageNo.
	ABSTRACT	
	LIST OF FIGURES	
	LIST OF ABBREVIATIONS	
1	INTRODUCTION	8
	OVERVIEW OF THE PROJECT	
2	LITERATURE REVIEW	10
	2.1 RESEARCH	
3	CONTRIBUTIONS, MATERIALS AND METHODS	13
	3.1 MATERIALS AND METHODS	
4	THE STEPS OF THE PROPOSED SYSTEMS	20
5	RESULTS	22

LIST OF ABBREVIATIONS

ML - Machine Learning

DL – Deep Learning

DCNN – Deep Convolutional Neural Network

DNN – Deep Neural Network

ILSVRC – ImageNet Large Scale Visual Recognition Channel

CHAPTER 1 - INTRODUCTION

The most widely used method for plant disease detection is simply naked eye observation by experts through which identification and detection of plants diseases are done. For doing so, a large team of experts is required, which costs very high when farms are large. At the same time, in some countries, farmers don't have proper facilities or even idea that they can contact to experts.

Due to which consulting experts even cost high as well as timeconsuming too. In such a condition, the suggested technique proves to be beneficial in monitoring large fields of crops. And automatic detection of the diseases by just seeing the symptoms on the plant leaves makes it by easier as well as cheaper.

Plant disease identification by the visual way is a more laborious task and at the same time less accurate and can be done only in limited areas. Whereas if automatic detection technique is used it will take fewer efforts, less time and more accurately. In plants, some general diseases are bacterial, black spotted, and others are Rust, viral and Red cotton Leaf. Image processing is the technique which is used for measuring the affected area of disease, and to determine the difference in the color of the affected area [1]. Image segmentation is the process of separating or grouping an image into different parts.

There are currently many different ways of performing image segmentation, ranging from the simple thresholding method to advanced color image segmentation methods. The segmentation process is based on various features found in the image. This might be color information, boundaries or segment of an image.

By 2050, global crop production must increase by at least 50% to support the predicted demand [1]. The majority of production currently occurs in Africa and Asia, where 83% of farmers are family run with little to no horticultural expertise [2, 3]. Due to this, yield losses of greater than 50%, as a result of pests and diseases are common [4].

In classifying crop diseases, the traditional method of human analysis by visual inspection is no longer feasible. The development of computer vision models offers a quick, standardised and accurate solution to this issue. Once trained, a classifier can also be deployed as an application [5]. Easy to use, all that is required is an internet connection and cameraequipped smartphone. Popular commercial apps 'iNaturalist' [6] and 'PlantSnap' [7] demonstrate how this can be executed.

Each year, smartphones continue to become more accessible and affordable. In 2020 there are approximately 5 billion smartphone users in the world [8]. Of this, one billion users are located in India and a further one billion are located in Africa. According to Statista, these figures have consistently risen every year for the last decade [9]. With these facts in mind, it is believed that AI apps will play an important role in shaping the future of farming.

CHAPTER 2 – LITERATURE SURVEY

2.1 RESEARCH

Extensive research has been conducted to explore various methods for automated identification of plant diseases. The disease can manifest in various parts of the plant such as roots, stem, fruit or leaves. As stated before, this work concentrates, particularly on leaves.

Discussed a methodology for recognition of plant diseases present on leaves and stem. The proposed work is composed of K-Means segmentation technique and the segmented images are classified using a neural network. They developed a method for detecting the visual signs of plant diseases by using the image processing algorithm. The accuracy of the algorithm was tested by comparing the images, which were segmented manually with those automatically segmented.

Discussed various techniques to segment the diseased part of the plant. This paper also discussed some Feature extraction and classification techniques to extract the features of infected leaf and the classification of plant diseases. The use of ANN methods for classification of disease in plants such as self-organizing feature map, back propagation algorithm, SVMs, etc. can be efficiently used. From these methods, we can accurately identify and classify various plant diseases using image processing techniques.

An approach based on image processing is used for automated plant diseases classification based on leaf image processing the research work is concerned with the discrimination between diseased and healthy soybean leaves using SVM classifier. They have tested our algorithm over the database of 120 images taken directly from different farms using different mobile cameras.

The SIFT algorithm enables to correctly recognize the plant species based on the leaf shape. The SVM classifier can help in recognizing normal and Diseased soybean leaves with an average accuracy as high as 93.79%. The main aim of the proposed work is to provide inputs to an autonomous DSS which will provide necessary help to the farmers as and when required over the mobile. This system will provide help to the farmer with minimal efforts. The farmer only needs to capture the image of the plant leaf using a mobile camera and send it to the DSS, without any additional inputs.

The work represents groundnut leaf disease extraction and classification using color imagery. The color imaginary transform, color co-occurrence matrix, feature extraction will be done and get an efficiency output with a neural network, Back propagation gives efficient groundnut leaf detection with a complex background, in this work we classified only four different diseases with 97 AI % of efficiency. But in the future, the work carried out more diseases by using this method.

Contain the study of detection of plant diseases and the detection of the infected part of plants. Initially, input images are taken and then image processing is started. Background and Black pixels are both segmented in the first step. Then Hue and Saturation part of the image is also separated. And finally infected part and infected area % and a name of the disease is acquired which is main work using our proposed methodology. The main aim of this work is to provide the advancement and enhancement in computing classifiers of a neural network approach and provide better results. This study contains a unique work that is it will calculate the % of an infected area of plants.

2.2 Block Diagram

The Basic Process of the Proposed methodology

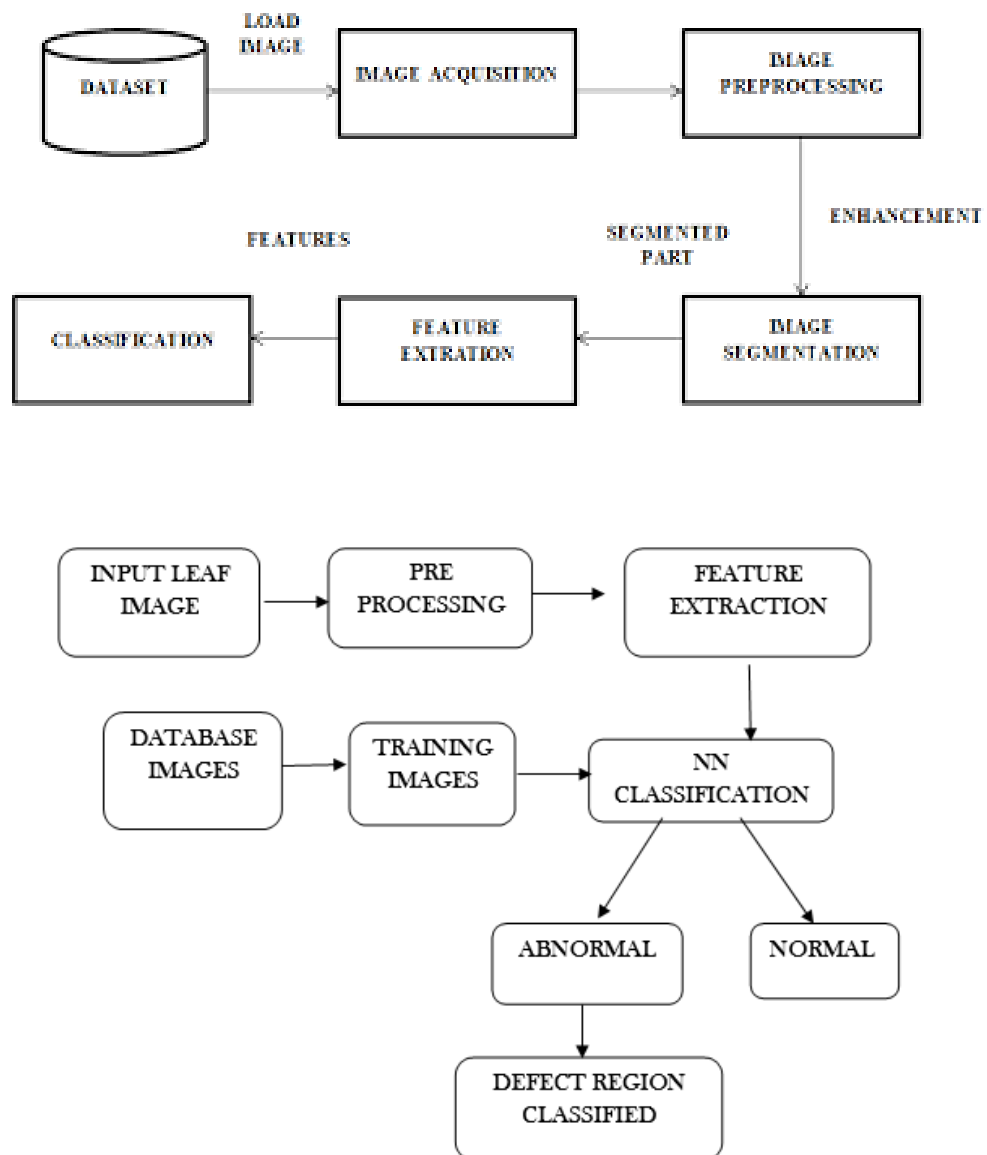


FIG 1. . Generalize framework to train and test the deep learning (DL) meta-architectures for plant disease identification.

CHAPTER 3 – CONTRIBUTIONS, MATERIALS AND METHODS

This study aims to evaluate the use of a pre-trained ResNet34 model in training a plant disease classifier. Three plant species will be focused on. These include potato (*Solanum tuberosum*), tomato (*Solanum lycopersicum*) and rice (*Oryza sativa*). For each species the model will be trained to recognize a select number of diseases or state of healthiness.

The specific goals of this research are to:

- i) Determine the model's overall effectiveness in classifying diseases using both a validation and test dataset.
- ii) Compare the model's accuracy when tested with various image sizes and the augmentation settings.
- iii) Deploy the trained model to create an easy to use the web applications.

Due to an uneven class distribution, both the f1-score and accuracy metrics will be examined in accessing the model's performance. Once the model reaches an accuracy and F1score of greater than 80%, it will be accepted.

This research will be carried out with the needs of smallholder farmers in mind. The classifier and web application will require both a smartphone and internet connection, which as previously mentioned, continue to reach remote regions. In acknowledging the limitations of basic camera phones, the model will be tested with a variety of image sizes and augmentation settings.

MATERIALS AND METHODS

This section describes the steps involved in creating and deploying the classifier. Classification by CNN is divided into three phases which tackle separate tasks. All work involved in this research was completed on one machine, with specifications listed.

A. Data Acquisition

All Potato and Tomato imagery derive from ‘The PlantVillage Dataset’ [35], an open-access repository which contains in total 54,323 images. All Rice imagery originates from the “Rice Diseases Image Dataset” Kaggle dataset [36]. For each species, a select number of classes are chosen, with details viewable in Table II.

All images are captured in a controlled environment. Due to this, model bias is expected. To access this, a test dataset containing 50 images, sourced from Google is also established. These images contain additional plant anatomy, infield background data and varying stages of disease.

TABLE I. MACHINE SPECIFICATIONS

Hardware & Software	Characteristics
Memory Processor	8.0GB Intel(R) Core™ i5-9300H
Graphics	CPU @ 2.40GHz NVIDIA GeForce RTX 2060 6GB GDDR6
Operating system	Windows 10 Home 64

B. Data Pre-Processing.

The dataset is divided into 80% for training and 20% for validation. First, augmentation settings are applied to the training data. These are generated ‘on the fly’, with each operation carrying a weighted probability of appearing in each epoch.

Each directory of the plant disease dataset folder varies in the number of images. Instead of taking them all, we select the first **N_IMAGES** from each directory to train our model.

Finally, we set the path of the dataset in the **root_dir** to access plant images.

The settings applied include flipping (random), padding mode (reflection) and zoom with crop (scale = (1.0,1.5)). 'Zoom with crop' was later omitted after discovering that it had inappropriately cropped areas of infected leaf. Finally, all images are re-sized and normalized. Resizing is carried out using a compress function, to 150 x 150. As a pretrained model is used, the RGB ImageNet statistics are used to normalize. A sample of the final pre-processed images is viewable.

C. Classification by CNN

1) Phase One – Trialling of Image size

Phase one aims to investigate the effect that image size has on model performance. In total, five images sized are tested ranging from 150 x 150 to 255 x 255.

To begin, the Resnet34 pre-trained weights are downloaded. As a default of transfer learning, all layers with the except of the final two layers are frozen. These contain new weights and are specific to the plant disease classification task. Freezing allows these layers to be disease separately trained, without backpropagating the gradients. In exactly this way, the 1cycle policy is used to train the final layers.

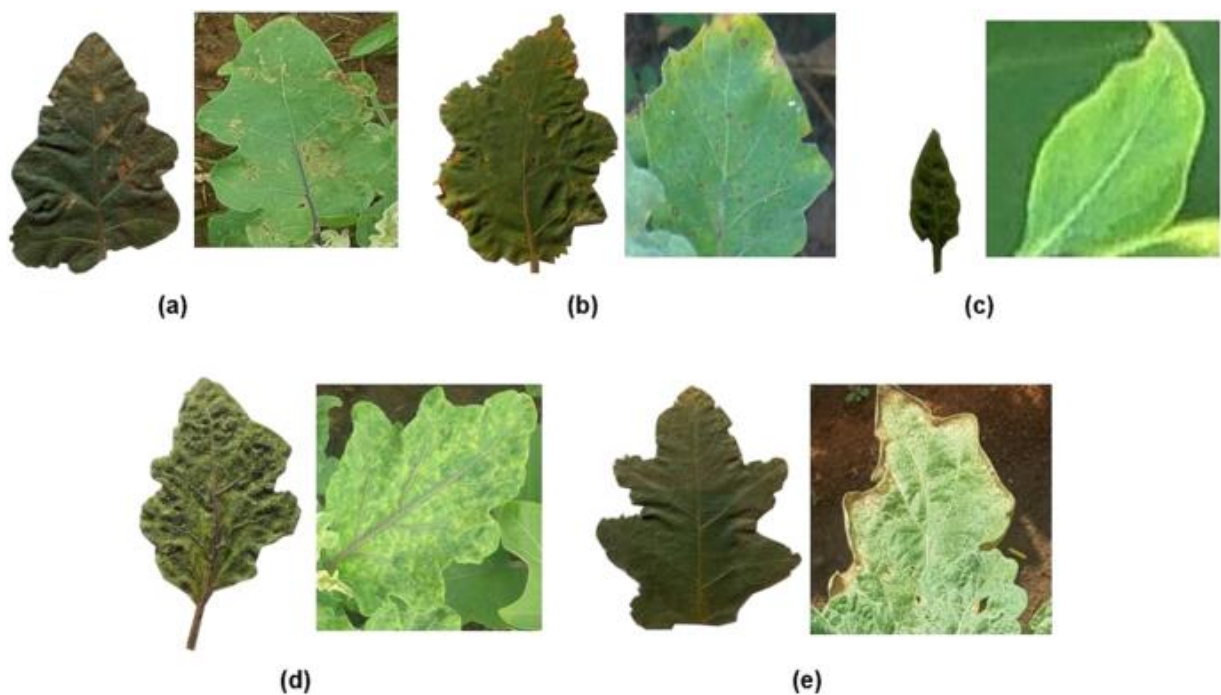
With this complete, the remaining layers are released. To aid the fine-tuning process, a plot displaying learning rate vs loss is generated and analysed. From this, a suitable learning is selected, and the model is run.

2) Phase Two – Model Optimisation

Using the most suitable image size, the ResNet34 model is optimised. To further improve the model's performance, additional augmentation settings are added (Fig. 2). Operations include brightness changes (0.4,0.7) and warp (0.5).

Next, the final two layers are isolated and trained at the default learning rate. With this complete, fine tuning is performed, running multiple trials to test a series of learning rates and number of epochs. Pre-processed image - augmentation settings = flipping (random), padding mode (reflection).

Fig 1. (I) Disease Classification in Eggplant Using Pre-trained VGG16 and MSVM



An example images of leaf acquired in laboratory and field condition

(a) *Epilachna* beetle, **(b)** *Cercospora* leaf spot, **(c)** Little leaf disease, **(d)** Tobacco Mosaic Virus (TMV), **(e)** Two spotted spider mite.

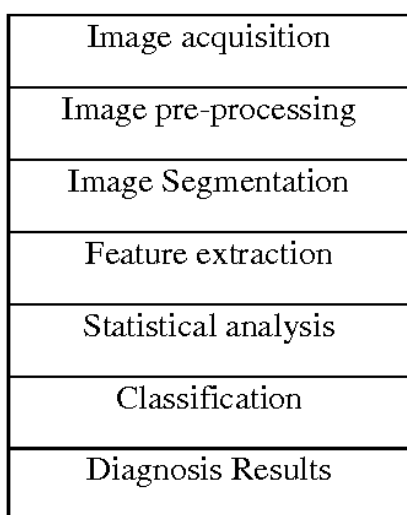
3) Phase Three – Visualisations

For the purpose of interpretation, a series of visualisations are generated based on the validation and test datasets. Additionally, the model is deployed to create a web application. To achieve this, the completed essential files are stored in a GitHub repository and the model is exported as a pickle file. To deploy the model, the repository is connected to the unified platform; Render. In carrying out this task, the 'Render Examples' GitHub repository was used as a guide.

Fig 2. (ii) SAMPLE IMAGES FROM THE DATASETS



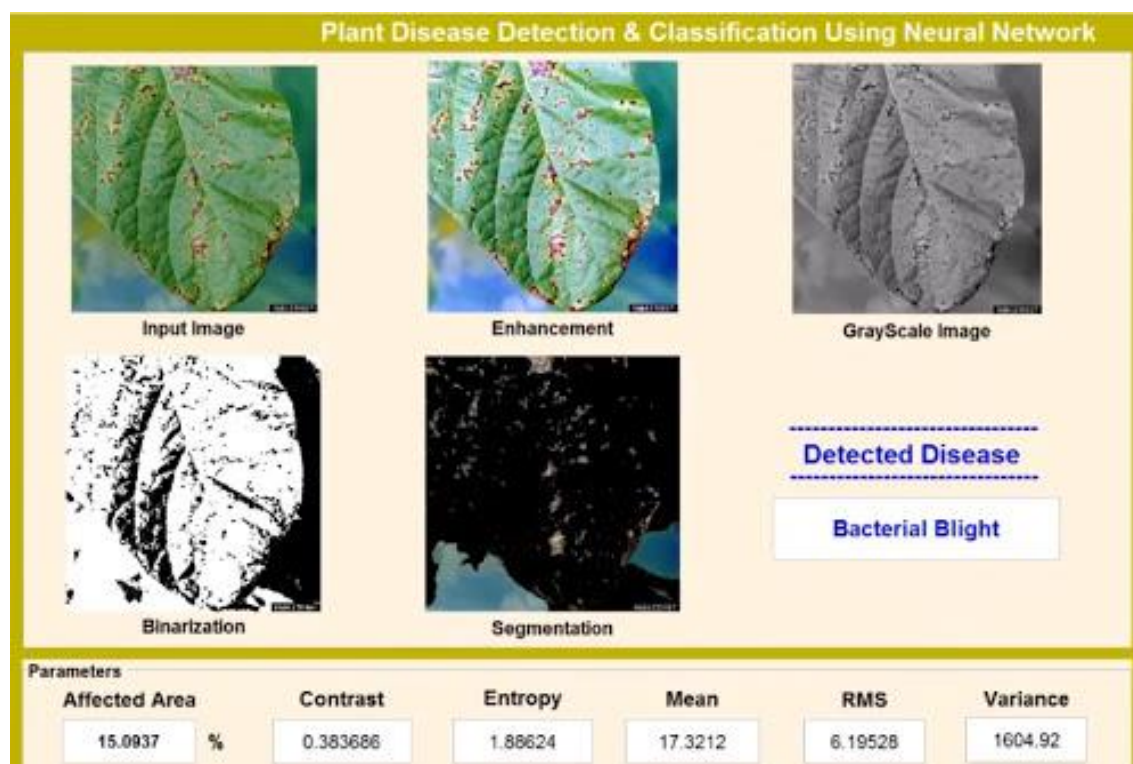
Fig 3. (iii) BLOCK DIAGRAM OF PROPOSED APPROACH



Using Neural Networks

The aim is to detect the symptoms of the disease occurred in leaves in an accurate way. Once the captured image is pre-processed, the various properties of the plant leaf such as intensity, color and size are extracted and sent to with Neural Network for classification.

Fig 4. (I) PLANT DISEASE DETECTION USING NEURAL NETWORK



CHAPTER 4 - THE STEPS OF THE PROPOSED SYSTEM

A. Image Acquisition- The images of the plant leaf are captured through the camera. This image is in RGB (Red, Green, and Blue) for color transformation structure for the RGB leaf image is created, and then, a device-independent color space transformation for the color transformation structure is applied .

B. Image Pre-processing To remove noise in the image or other object removals, different pre-processing techniques is considered. RGB to Gray Converter-Weighted method or luminosity method-You has seen the problem that occurs in the average method. The weighted method has a solution to that problem.

New grayscale image = ($(0.3 * R) + (0.59 * G) + (0.11 * B)$). According to this equation, Red has contributed 30%, Green has contributed 59% which is greater in all three colors and Blue has contributed 11%.

Image Resize- The resolution of document images is typically higher than 2000 _ 2000, which is too large to be fed to a CNN with the current availability of computing resources.

CLAHE -Ordinary AHE tends to over amplify the contrast in near-constant regions of the image since the histogram in such regions is highly concentrated. As a result, AHE may cause noise to be amplified in near constant regions. Contrast Limited AHE (CLAHE) is a variant of adaptive histogram equalization in which the contrast amplification is limited, so as to reduce this problem of noise amplification.

Gaussian blur -In image processing, a Gaussian blur (also known as Gaussian smoothing) is the result of blurring an image by a Gaussian function (named after mathematician and scientist Carl Friedrich Gauss). It is a widely used effect in graphics software, typically to reduce image noise and reduce detail.

Convolutional Neural Networks—After removing noise from the image it required to extracts the feature. We propose to use a CNN for document image classification.

The main idea is to learn a hierarchy of feature detectors and train a nonlinear classifier to identify complex document layouts.

CHAPTER 5 - RESULTS

All the experiments are performed in MATLAB. For input data disease, samples of plant leaves like rose with bacterial disease, beans leaf with bacterial disease, lemon leaf with Sun burn disease, banana leaf with early scorch disease and fungal disease in beans leaf are considered. Original images which are followed by output segmented images. Segmented image can be classified into different plant diseases. The input and output image where input image is a banana leaf with early scorch disease and output image shows the classification of disease using feature extraction method.

The co-occurrence features are calculated after mapping the R, G, B components of the input image to the thresholded images. The co-occurrence features for the leaves are extracted and compared with the corresponding feature values that are stored in the feature library. The classification is first done using the Minimum Distance Criterion with K-Mean Clustering and shows its efficiency with accuracy of 86.54%. The detection accuracy is improved by proposed algorithm.

Therefore, compared with the new learning, this paper uses the transfer learning to converge faster and achieve better model identification effect. It can meet the requirements of smart agriculture for low hardware resources, fast training time, and high training efficiency.

Then, the image is input into transfer learning model based on the segmentation of Chan–Vese algorithm. As a contrast, the image that has not been processed in this paper is input.

5.1 (I) Image-Based Plant Disease Identification by Deep Learning Meta-Architectures

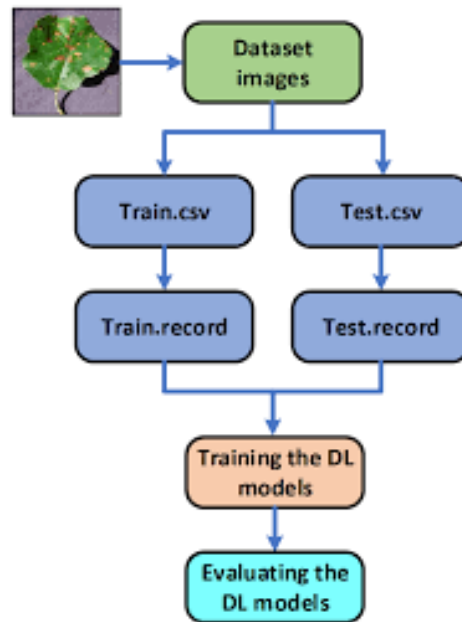


Fig 5. The DL detectors were constructed by taking training images with bounding box coordinates and then evaluated their performance on the testing dataset.

5.2 Detection of plant leaf diseases using image segmentation and soft computing techniques

Displays type of diseases along with rate of occurrence. Red spots indicates the type of the disease.

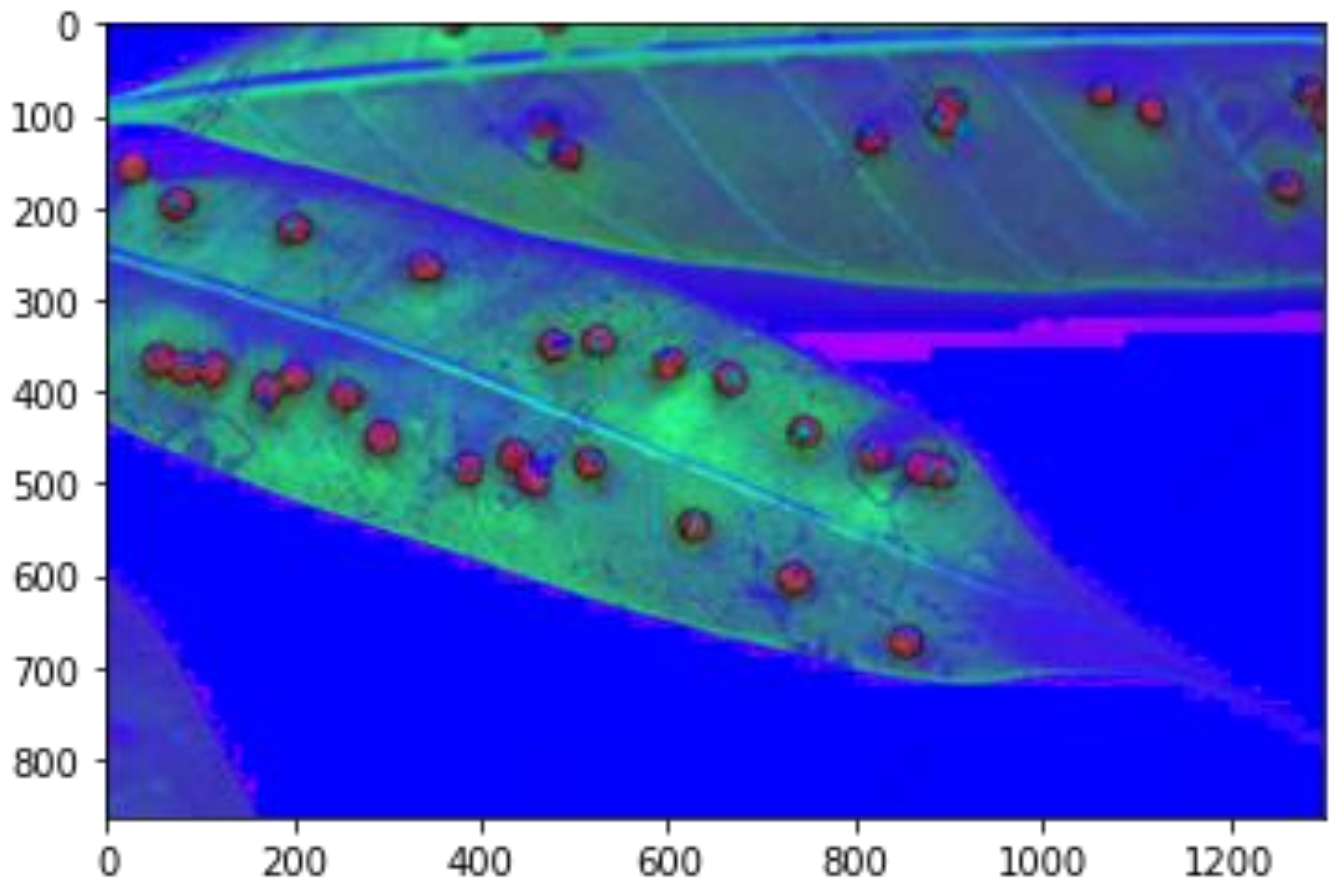


Fig 6. (a) represents the one type of disease along with its rate of occurrence

Indicates three types of plant disease (miner, temperature, and gray mold) in a single image, one class of disease but contains different patterns on the front and back side of the image, displays different patterns of gray mold in the starting and end stages.

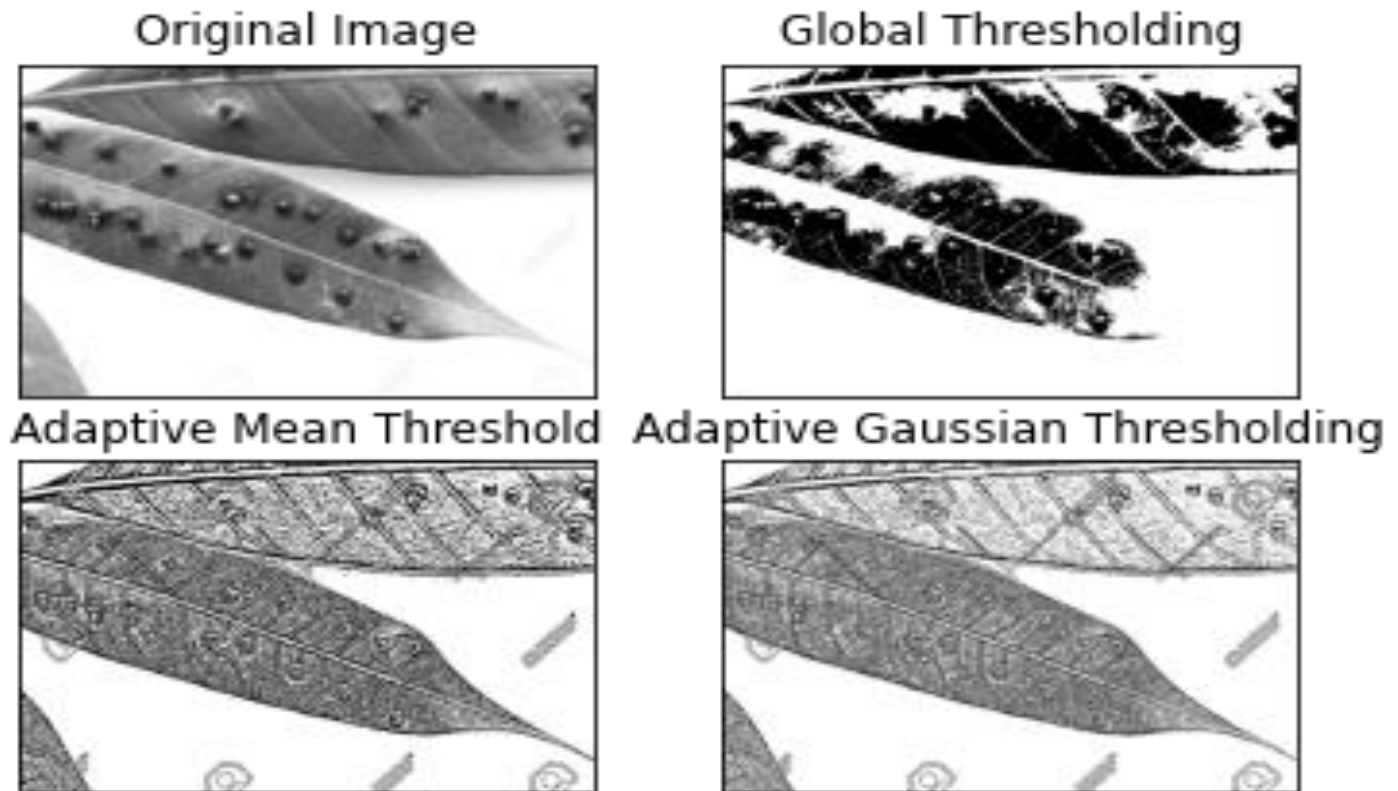


Fig 7. segmentation using a binary threshold algorithm

Leaf Dataset type:

1. Leaf Color, grayscale and Leaf segmented

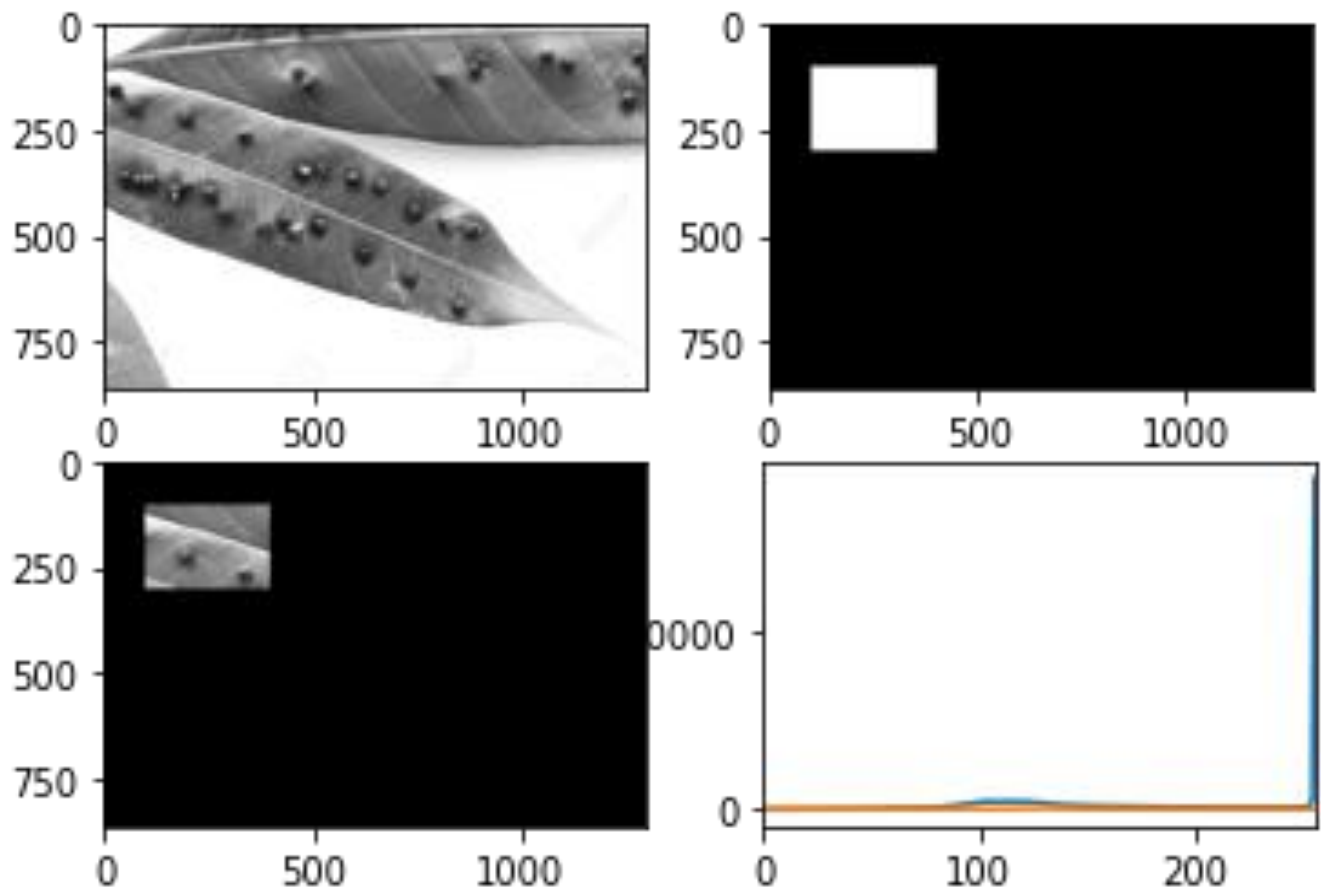


Fig 8. Recognition of the Diseases

CONCLUSION

With very less computational efforts the optimum results were obtained, which also shows the efficiency of proposed algorithm in recognition and classification of the leaf diseases. Another advantage of using this method is that the plant diseases can be identified at early stage or the initial stage. Banana, beans, jackfruit, lemon, mango, potato, tomato, and sapota are some of those ten species on which proposed algorithm is tested. Therefore, related diseases for these plants were taken for identification.

The model not only adapts to complex environments, but also increases the accuracy of identification. Compared with the traditional model, the model proposed in this paper not only guarantees the robustness of the convolutional neural network, but also reduces the number and quality requirements of the convolutional neural network on the data set and obtains better results.

The model which overcomes the problem of environment complexity can get an accurate identification result in practical application. Furthermore, this study enriches the existing theory and helps to improve the accuracy. At the same time, it is of great significance for the study of plant disease identification in the field of environmental complexity and helps researchers pay attention to the important role of environmental complexity in plant disease identification.

Therefore, the model applies information technology to agricultural production and is favorable to sustainable development of smart agriculture.

REFERENCES

1. Savita N. Ghaiwat, Parul Arora Detection and classification of plant leaf diseases using image processing techniques: a review Int J Recent Adv Eng Technol, 2 (3) (2014), pp. 2347-2812 ISSN (Online)
2. Sanjay B. Dhaygude, Nitin P. Kumbhar Agricultural plant leaf disease detection using image processing Int J Adv Res Electr Electron Instrum Eng, 2 (1) (2013)
3. R. Badnakhe Mrunalini, Prashant R. Deshmukh An application of K-means clustering and artificial intelligence in pattern recognition for crop diseases Int Conf Adv Inf Technol, 20 (2011) 2011 IPCSIT
4. S. Arivazhagan, R. Newlin Shebiah, S. Ananthi, S. Vishnu Varthini Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features Agric Eng Int CIGR, 15 (1) (2013), pp. 211-217
5. Anand H. Kulkarni, R.K. Ashwin Patil Applying image processing technique to detect plant diseases Int J Mod Eng Res, 2 (5) (2012), pp. 3661-3664
6. J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosystems Engineering*, vol. 172, pp. 84–91, 2018.
7. G. Geetharamani and J. Arun Pandian, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," *Computers & Electrical Engineering*, vol. 76, pp. 323–338, 2019.
8. P. F. Konstantinos, "Deep learning models for plant disease detection and diagnosis," *Computers & Electrical Engineering*, vol. 145, pp. 311–318, 2018.

9. Sabah Bashir, Navdeep Sharma Remote area plant disease detection using image processing IOSR J Electron Commun Eng, 2 (6) (2012), pp. 31-34 ISSN: 2278-2834
10. Smita Naikwadi, Niket Amoda Advances in image processing for detection of plant diseases Int J Appl Innov Eng Manage, 2 (11) (2013)
11. Sanjay B. Patil, et al. Leaf disease severity measurement using image processing Int J Eng Technol, 3 (5) (2011), pp. 297-301.
12. V. Singh and A. K. Misra, "Detection of plant leaf diseases using image segmentation and soft computing techniques," *Information Processing in Agriculture*, vol. 4, no. 1, pp. 41–49, 2017.
13. S. P. Mohanty, D. P. Hughes, and S. Marcel, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, p. 1419, 2016.
14. Y. Guo, X. Hu, Y. Zou et al., "Maximizing E-tailers' sales volume through the shipping-free discount and product recommendation system," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–14, 2020.

APPENDIX

A.SOURCE CODE

```
import numpy as np
import pickle
import cv2
from os import listdir
from sklearn.preprocessing import LabelBinarization
from keras.models import Sequential
from keras.layers.normalization import BatchNormalization

from keras.layers.convolutional import Conv2D
from keras.layers.convolutional import MaxPooling2D
from keras.layers.core import Activation, Flatten, Dropout, Dense
from keras import backend as K
from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import Adam
from keras.preprocessing import image
```

```
from keras.preprocessing.image import img_to_array
from sklearn.preprocessing import MultiLabelBinarizer
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

```
EPOCH = 25
```

```
INIT_LR = 1e-3
```

```
BS = 32
```

```
Default_image_size = tuple((256, 256))
```

```
Image_size = 0
```

```
Directory_root = 'c:\\Users\\Downloads\\Pictures\\diseasedleaf'
```

```
Width = 256
```

```
Height = 256
```

```
Depth = 3
```

```
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
```

```
#import the required libraries
import numpy as np
import matplotlib.pyplot as plt
import cv2
%matplotlib inline
image = cv2.imread('diseased leaf.jpg')
#converting image to Gray scale
    gray_image = cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
#plotting the grayscale image
plt.imshow(gray_image)
#converting image to HSV format
    hsv_image = cv2.cvtColor(image,cv2.COLOR_BGR2HSV)
#plotting the HSV image
plt.imshow(hsv_image)
```

```

#import the libraries

import numpy as np

import matplotlib.pyplot as plt

import cv2 %matplotlib inline


#ADAPTIVE THRESHOLDING

gray_image = cv2.imread('diseased leaf.jpg',0)

ret,thresh_global = cv2.threshold(gray_image,127,255,cv2.THRESH_BINARY) #here
11 is the pixel neighbourhood that is used to calculate the threshold value
thresh_mean =
cv2.adaptiveThreshold(gray_image,255,cv2.ADAPTIVE_THRESH_MEAN_C,cv2.T
HRESH_BINARY,11,2)

thresh_gaussian =
cv2.adaptiveThreshold(gray_image,255,cv2.ADAPTIVE_THRESH_GAUSSIAN
_C,cv2.THRESH_BINARY,11,2)

```

```
names = ['Original Image','Global Thresholding','Adaptive Mean Threshold','Adaptive  
Gaussian Thresholding']  
images = [gray_image,thresh_global,thresh_mean,thresh_gaussian]
```

```
for i in range(4):  
    plt.subplot(2,2,i+1),plt.imshow(images[i], 'gray')  
    plt.title(names[i])  
    plt.xticks([],plt.yticks([]))  
  
plt.show()
```

```
img = cv2.imread('diseased leaf.jpg',0)

# create a mask
mask = np.zeros(img.shape[:2], np.uint8)
mask[100:300, 100:400] = 255
masked_img = cv2.bitwise_and(img,img,mask = mask)

# Calculate histogram with mask and without mask
# Check third argument for mask
hist_full = cv2.calcHist([img],[0],None,[256],[0,256])
hist_mask = cv2.calcHist([img],[0],mask,[256],[0,256])
```

```
plt.subplot(221), plt.imshow(img, 'gray')  
plt.subplot(222), plt.imshow(mask, 'gray')  
plt.subplot(223), plt.imshow(masked_img, 'gray')  
plt.subplot(224), plt.plot(hist_full), plt.plot(hist_mask)  
plt.xlim([0,256])
```

```
plt.show()
```

```
import cv2
import numpy as np
from matplotlib import pyplot as plt

img = cv2.imread('diseased leaf.jpg')
color = ('b','g','r')
for i,col in enumerate(color):
    histr = cv2.calcHist([img],[i],None,[256],[0,256])    plt.plot(histr,color = col)
    plt.xlim([0,256])
plt.show()
```

A) Plant Disease Detection Using Transfer Learning

ORIGINAL IMAGE

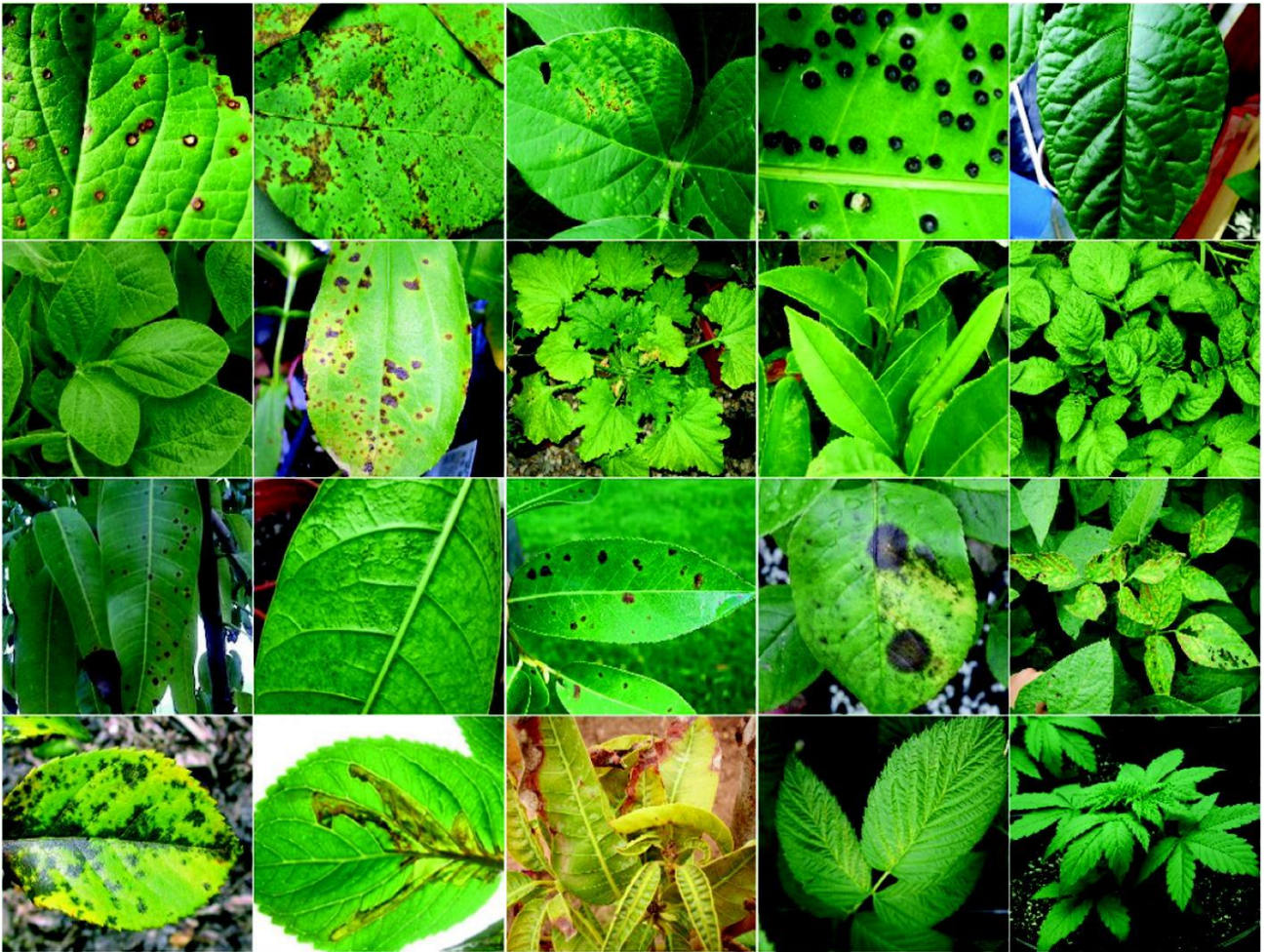


Fig 1. The Original Images of Leaf was Taken

B) Image Thresholding in Python Using OpenCV

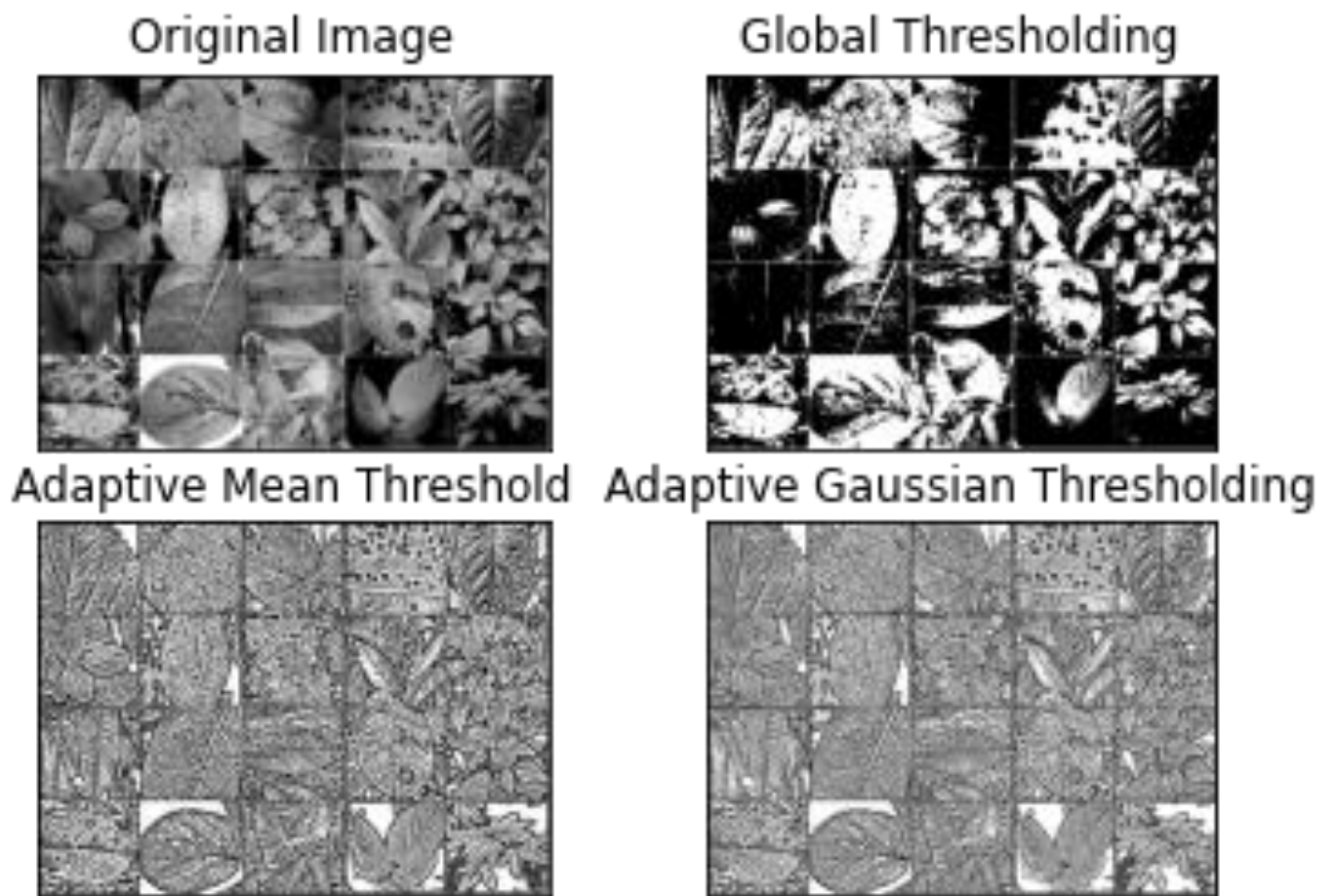
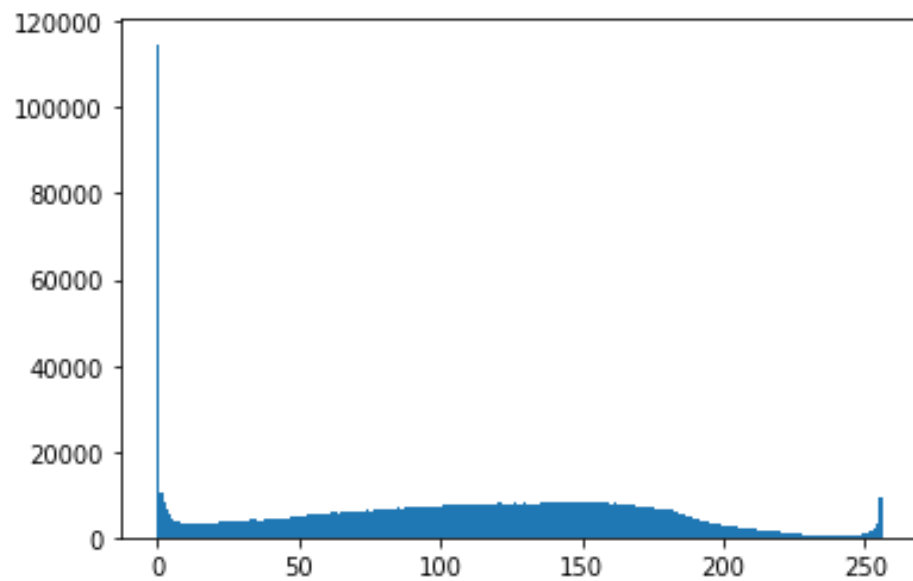


Fig 2. Plant disease leaf image segmentation. Original leaf image, grayscale image, grayscale histogram and histogram clustering.

C) Quantative analysis of Deep Leaf. Plant Disease Detector on the smart edge

Training and Validation Accuracy



Training and Validation Loss

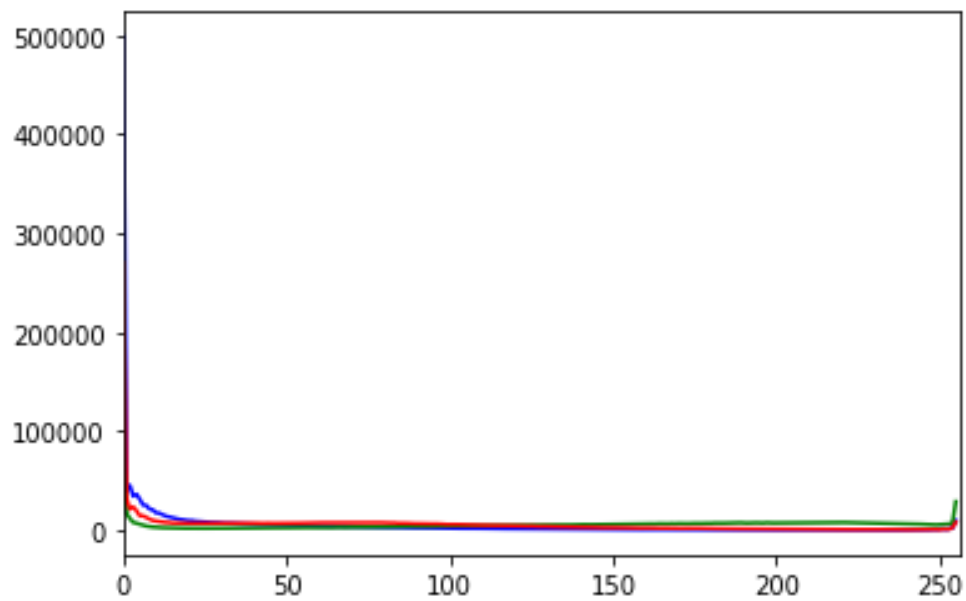


Fig 3. Plant disease detection using Deep Neural Network (DNN)

WEBSITES:

<http://www.slideshare.com>

<http://www.oldplantlife.com>

<http://www.w3schools.com>