# “PLANT DISEASE DETECTION”

Mini- Project (OPEN SOURCE TECH LAB)

Submitted in partial fulfillment of the requirement of University of Mumbai
For the Degree of

**(Computer Engineering)**

**By**


| | |
|---|---|
| **1) NAHUSH KULKARNI** | **“ID No: TU3F1819026”** |
| **2) JANMEJAY PATIL** | **“ID No: TU3F1819039”** |
| **3) ANIKET THORAT** | **“ID No: TU3F1819038”** |

**Under the Guidance of**

**Prof. “SONALI DHAMELE”**

**Department of Computer Engineering**
**TERNA ENGINEERING COLLEGE**
**Plot no.12, Sector-22, Opp. Nerul Railway station,**
**Phase-11, Nerul (w), Navi Mumbai 400706**
**UNIVERSITY OF MUMBAI**

# Terna Engineering College

**NERUL, NAVI MUMBAI**

# CERTIFICATE

*This is to certify that*

| | |
|---|---|
| **1) NAHUSH KULKARNI** | **“ID No: TU3F1819026”** |
| **2) JANMEJAY PATIL** | **“ID No: TU3F1819039”** |
| **3) ANIKET THORAT** | **“ID No: TU3F1819038”** |

*Has satisfactorily completed the requirements of the **Mini Project***

*Of subject*

**“OPEN SOURCE TECH LAB”**

*As prescribed by the **University of Mumbai** Under the guidance of*

Prof. “**SONALI DHAMELE**”

**Subject In charge** **APC** **HOD**

# INDEX

# Introduction

Modern technologies have given human society the ability to produce enough food to meet the demand of more than 7 billion people. However, food security remains threatened by a number of factors including climate change), the decline in pollinators (Report of the Plenary of the Intergovernmental Science-Policy Platform on Biodiversity Ecosystem and Services on the work of its fourth session, 2016), plant diseases (Strange and Scott, 2005), and others. Plant diseases are not only a threat to food security at the global scale, but can also have disastrous consequences for smallholder farmers whose livelihoods depend on healthy crops. In the developing world, more than 80 percent of the agricultural production is generated by smallholder farmers (UNEP, 2013), and reports of yield loss of more than 50% due to pests and diseases are common (Harvey et al., 2014). Furthermore, the largest fraction of hungry people (50%) live in smallholder farming households (Sanchez and Swaminathan, 2005), making smallholder farmers a group that's particularly vulnerable to pathogen-derived disruptions in food supply.

Various efforts have been developed to prevent crop loss due to diseases. Historical approaches of widespread application of pesticides have in the past decade increasingly been supplemented by integrated pest management (IPM) approaches (Ehler, 2006). Independent of the approach, identifying a disease correctly when it first appears is a crucial step for efficient disease management. Historically, disease identification has been supported by agricultural extension organizations or other institutions, such as local plant clinics. In more recent times, such efforts have additionally been supported by providing information for disease diagnosis online, leveraging the increasing Internet penetration worldwide. Even more recently, tools based on mobile phones have proliferated, taking advantage of the historically unparalleled rapid uptake of mobile phone technology in all parts of the world.

Computer vision, and object recognition in particular, has made tremendous advances in the past few years. The PASCAL VOC Challenge (Everingham et al., 2010), and more recently the Large Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al., 2015) based on the ImageNet dataset (Deng et al., 2009) have been widely used as benchmarks for numerous visualization-related problems in computer vision, including object classification. In 2012, a large, deep convolutional neural network achieved a top-5 error of 16.4% for the classification of images into 1000 possible categories (Krizhevsky et al., 2012). In the following 3 years, various advances in deep convolutional neural networks lowered the error rate to 3.57%

# PROBLEM STATEMENT

Plant Disease Detection is developed in Python. This project meets the results by the means of Image Segmentation and Channel Separation. In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse.
Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

**Existing System:**

The existing method for plant disease detection is simply naked eye observation by experts through which identification and detection of plant diseases is done. For doing so, a large team of experts as well as continuous monitoring of plant is required, which costs very high when we do with large farms.
At the same time, in some countries, farmers do not have proper facilities or even idea that they can contact to experts. Due to which consulting experts even cost high as well as time consuming too. In such conditions, the suggested technique proves to be beneficial in monitoring large fields of crops.

**Proposed System:**

The proposed system includes Automatic detection of the diseases by just seeing the symptoms on the plant leaves. Therefore making it easier as well as cheaper. This uses machine vision to provide image based automatic process control, inspection, and robot guidance.

We use OpenCV, a free to use yet powerful Computer Vision Library which supports Python. The given image of a plant leaf kept on a white background is channel separated to get Red, Green, and Blue channel images. Blue channel being throughout the leaf, seems to be less useful. Hence, Red and Green Channel Images are used to get an image that contains less of the leaf and more diseased part. A variable called the "Processing Factor" is used to fine tune the disease image. The processing factor simply removes the pixels with higher white colour intensity than its value. The white background from the original image is separated from the leaf by checking if a pixel's value in all three channels(R-G-B) goes above 200. If so, the pixel is turned to a value of 255 in Alpha channel image. If not, the pixel is turned to value of 0. This Alpha channel clearly tells us which part to consider while calculating the percentage disease. The Disease Image and the Alpha Channel is used to calculate the Percentage Disease by using the following formula:

$$Percentage\ Disease = \frac{\text{Number of black pixels in Disease Image}}{\text{Number of black pixels in Alpha Channel}} \times 100$$

Tkinter is used to create a window to get the Processing Factor input via a slider with values ranging from 0 to 255 with the default being set to 150(works just fine for many images). Also, a label displays the Percentage Disease on this window.

## Recommended System Requirements:

- Operating System:
  - Windows 10
  - Mac OS 10.14 and above
  - Ubuntu 18 and above
- Intel Core i5 or higher Processor,
- 8 GB of RAM or higher,
- Python compiler with needed libraries,
- Text Editor.

## Technologies and Tools Used:

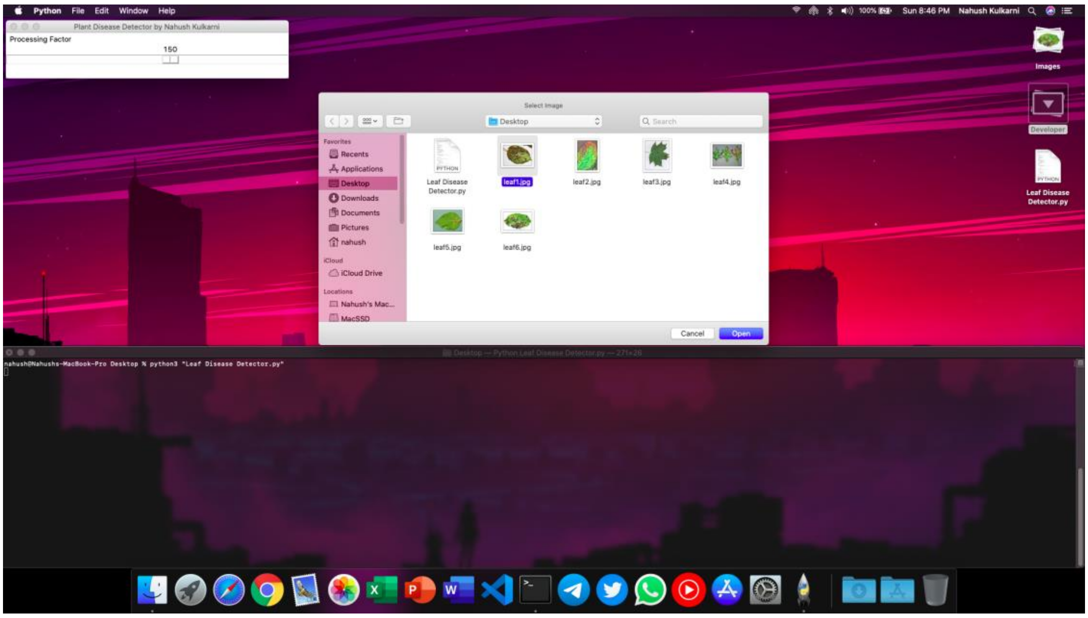
- PyCharm,
- Tkinter,
- OpenCV.

# IMPLEMENTATION

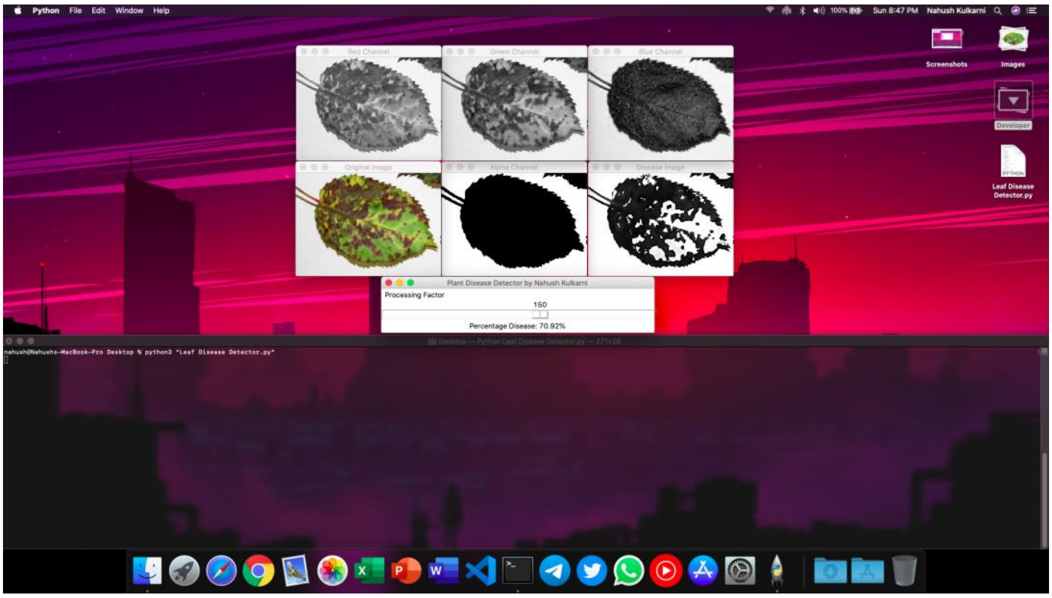
```python
from tkinter import filedialog

import cv2
import sys
import tkinter


def ProcessImage(self):
    OriginalImage = cv2.imread(filename, 1)
    cv2.imshow("Original Image", OriginalImage)
    b = OriginalImage[:, :, 0]
    g = OriginalImage[:, :, 1]
    r = OriginalImage[:, :, 2]
    cv2.imshow("Red Channel", r)
    cv2.imshow("Green Channel", g)
    cv2.imshow("Blue Channel", b)
    Disease = r - g
    global Alpha
    Alpha = b
    GetAlpha(OriginalImage)
    cv2.imshow("Alpha Channel", Alpha)
    ProcessingFactor = S.get()
    for i in range(0, OriginalImage.shape[0]):
        for j in range(0, OriginalImage.shape[1]):
            if int(g[i, j]) > ProcessingFactor:
                Disease[i, j] = 255
    cv2.imshow("Disease Image", Disease)
    DisplayDiseasePercentage(Disease)
    S.bind('<ButtonRelease-1>', ProcessImage)
    MainWindow.mainloop()


def GetAlpha(OriginalImage):
    global Alpha
    for i in range(0, OriginalImage.shape[0]):
        for j in range(0, OriginalImage.shape[1]):
            if OriginalImage[i, j, 0] > 200 and OriginalImage[i, j, 1] > 200 and
OriginalImage[i, j, 2] > 200:
                Alpha[i, j] = 255
            else:
                Alpha[i, j] = 0


def GetFile():
    if len(sys.argv) > 1:
        return sys.argv[1]
    else:
        return filedialog.askopenfilename(title="Select Image")


def DisplayDiseasePercentage(Disease):
    Count = 0
    Res = 0
```

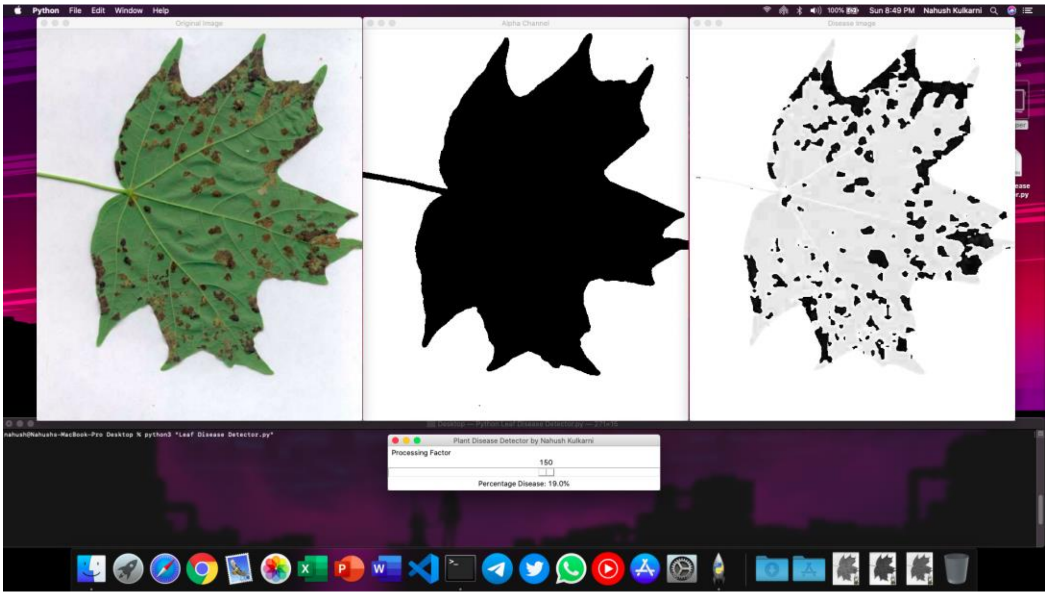
```
    for i in range(0, Disease.shape[0]):
        for j in range(0, Disease.shape[1]):
            if Alpha[i, j] == 0:
                Res += 1
            if Disease[i, j] < S.get():
                Count += 1
    Percent = (Count / Res) * 100
    DiseasePercent.set("Percentage Disease: " + str(round(Percent, 2)) + "%")


Alpha = None
MainWindow = tkinter.Tk()
MainWindow.title("Plant Disease Detector by Nahush Kulkarni")

S = tkinter.Scale(MainWindow, from_=0, to=255, length=500,
orient=tkinter.HORIZONTAL,
                  background='white', fg='black', troughcolor='white',
label="Processing Factor")
S.pack()
S.set(150)

DiseasePercent = tkinter.StringVar()
L = tkinter.Label(MainWindow, textvariable=DiseasePercent)
L.pack()

filename = GetFile()
if filename != "":
    ProcessImage(None)
else:
    print("No File!")
    exit(0)
```

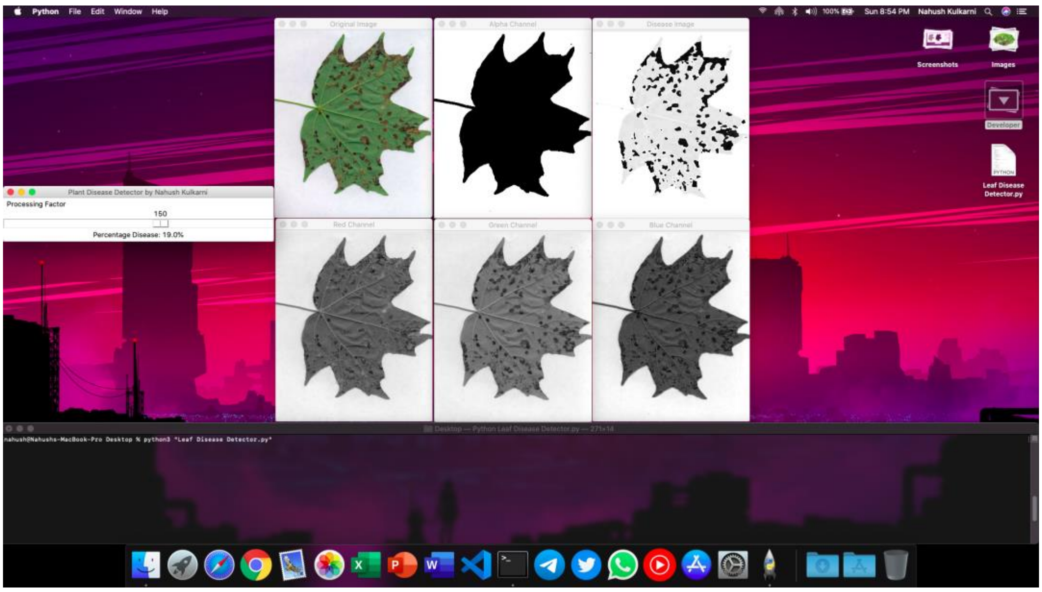
# Snapshots



Python File Edit Window Help
Sun 8:46 PM Nahush Kulkarni
Plant Disease Detector by Nahush Kulkarni
Processing Factor
150
Select Image
Desktop
Search
Favorites
Recents
Applications
Desktop
Downloads
Documents
Pictures
nahush
iCloud
iCloud Drive
Locations
Nahush's Mac...
MacSSD
Leaf Disease Detector.py
leaf1.jpg
leaf2.jpg
leaf3.jpg
leaf4.jpg
leaf5.jpg
leaf6.jpg
Cancel
Open
Images
Developer
Leaf Disease Detector.py
nahush@Nahushs-MacBook-Pro Desktop % python3 "Leaf Disease Detector.py"


Plant Disease Detector by Nahush Kulkarni
Processing Factor
150
Percentage Disease: 70.92%


Plant Disease Detector by Nahush Kulkarni
Processing Factor
150
Percentage Disease: 19.0%


Plant Disease Detector by Nahush Kulkarni
Processing Factor
150
Percentage Disease: 19.0%

# Conclusion

Our project fully meets the objectives specified of the user for which it has been developed. The application operates at a high level of efficiency and the users associated with the system understand its advantage.

In order to get completed, finding what were the key concepts and what was the roles of them on the project's different parts was necessary. Outline of the major milestones that would be involved in completing the project, and worked backwards to break down the work that would need to be done at each stage was the procedure followed
After work and research, the project is now completely debugged and runnable.

By delivering this project, diseases classification techniques used for plant leaf disease detection and an algorithm for image segmentation technique that can be used for automatic detection as well as classification of plant leaf diseases later.

In doing so we used Python as it combines elements of highly portable, syntax-understandability, safety and security. Although the actual focus lies on the learning from the project. We expect that in the future there could be many other image detection programs could be developed using the knowledge of this project. Extending this project by linking it to other resources or creating new features and functions will maximize the added value of the hard work and time given in development, which will increase sustainable future aspects.

# Reference


[1] *"Detection of Potato Diseases Using Image Segmentation and Multiclass Support Vector Machine" by Monzurul Islam, Anh Dinh, Khan Wahid (Department of Electrical and Computer Engineering, University of Saskatchewan, Saskatoon, Canada) and Pankaj Bhowmik (National Research Council Canada Saskatoon, Saskatchewan, Canada).*

[2] *"Detection of Unhealthy Plant Leaves Using Image Processing and Genetic Algorithm with Arduino" by Arya M S (ECE Department, Vidya Academy of Science & Technology Thrissur, India), Anjali K (ECE Department, Vidya Academy of Science & Technology Thrissur, India) and Mrs.Divya Unni (Asst. Professor, ECE Department Vidya Academy of Science & Technology, Thrissur, India).*

[3] *"Detection of unhealthy region of plant leaves using Image Processing and Genetic Algorithm" by Vijai Singh (Asst Professor, IMS Engg. college Ghaziabad, UP, India), Varsha (Asst Professor JSSATE Noida, Uttar Pradesh, India), and Prof. A K Misra (Professor, MNNIT Allahabad, Uttar Pradesh, India).*