# LEAF AND DISEASE DETECTION

**A MINI PROJECT REPORT**

***Submitted by***


**SOUMYA RATHORE[RA2011029010033]**
**ANKITA DAS[RA2011029010023]**
**HARSHIT PRAKASH[RA2011029010022]**

*Under the guidance of*

**Dr.K. A. Varun Kumar**
(*Professor*)


***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE & ENGINEERING**

of

**FACULTY OF ENGINEERING AND TECHNOLOGY**


SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

S.R.M.Nagar, Kattankulathur, Chengalpattu District


**APRIL 2023**

# ABSTRACT

There is developing Indian population, which is increasingly dependent on the agricultural yield. The end goal is kept in mind to develop progressively the diseases need to be examined in earlier. Diseases are investigated utilizing different image processing techniques and diagnosed so that Farmers can overcome from yeild and financial loss.One of the important and tedious task in agricultural practices is detection of disease on crops. It requires huge time as well as skilled labor. This paper proposes a smart and efficient technique for detection of crop disease which uses computer vision and machine learning techniques. The proposed system is able to detect 20 different diseases of 5 common plants with 93% accuracy. Keywords: Digital image processing, Foreground detection, Machine learning, Plant disease detection

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ABBREVIATIONS

| | |
|---|---|
| **AES** | Advanced Encryption Standard |
| **ANN** | Artificial Neural Network |
| **CSS** | Cascading Style Sheet |
| **CV** | Computer Vision |
| **DB** | Data Base |
| **DNA** | Deoxyribo Neucleic Acid |
| **SQL** | Structured Query Language |
| **SVM** | Support Vector Machine |
| **UI** | User Interface |
| **GLCM** | Gray-Level Co-Occurrence Matrix |

# CHAPTER 1

# INTRODUCTION

In India about 70% of the populace relies on agriculture. Identification of the plant diseases is important in order to prevent the losses within the yield. It's terribly troublesome to observe the plant diseases manually. It needs tremendous quantity of labor, expertize within the plant diseases, and conjointly need the excessive time interval. Hence, image processing and machine learning models can be employed for the detection of plant diseases. In this project, we have described the technique for the detection of plant diseases with the help of their leaves pictures. Image processing is a branch of signal processing which can extract the image properties or useful information from the image. Machine learning is a sub part of artificial intelligence which works automatically or give instructions to do a particular task. The main aim of machine learning is to understand the training data and fit that training data into models that should be useful to the people. So it can assist in good decisions making and predicting the correct output using the large amount of training data. The color of leaves, amount of damage to leaves, area of the leaf, texture parameters are used for classification. In this project we have analyzed different image parameters or features to identifying different plant leaves diseases to achieve the best accuracy. Previously plant disease detection is done by visual inspection of the leaves or some chemical processes by experts. For doing so, a large team of experts as well as continuous observation of plant is needed, which costs high when we do with large farms. In such conditions, the recommended system proves to be helpful in monitoring large fields of crops. Automatic detection of the diseases by simply seeing the symptoms on the plant leaves makes it easier as well as cheaper. The proposed solution for plant disease detection is computationally less expensive and requires less time for prediction than other deep learning based approaches since it uses statistical machine learning and image processing algorithm

# Literature Review

In 2015, S. Khirade et Al. tackled the problem of plant disease detection using digital image processing techniques and back propagation neural network (BPNN) [1]. Authors have elaborated different techniques for the detection of plant disease using the images of leaves. They have implemented Otsu's thresholding followed by boundary detection and spot detection algorithm to segment the infected part in leaf. After that they have extracted the features such as color, texture, morphology, edges etc. for classification of plant disease. BPNN is used for classification i.e. to detect the plant disease. Shiroop Madiwalar and Medha Wyawahare analyzed different image processing approaches for plant disease detection in their research [2].

Authors analyzed the color and texture features for the detection of plant disease. They have experimented their algorithms on the dataset of 110 RGB images. The features extracted for classification were mean and standard deviation of RGB and YCbCr channels, grey level cooccurrence matrix (GLCM) features, the mean and standard deviation of the image convolved with Gabor filter. Support vector machine classifier was used for classification. Authors concluded that GCLM features are effective to detect normal leaves. Whereas color features and Gabor filter features are considered as best for detecting anthracnose affected leaves and leaf spot respectively. They have achieved highest accuracy of 83.34% using all the extracted features. Peyman Moghadam et Al. demonstrated the application of hyperspectral imaging in plant disease detection task [3]. visible and near-infrared (VNIR) and short-wave infrared (SWIR) spectrums were used in this research. Authors have used k-means clustering algorithm in spectral domain for the segmentation of leaf. They have proposed a novel grid removal algorithm to remove the grid from hyperspectral images.

Authors have achieved the accuracy of 83% with vegetation indices in VNIR spectral range and 93% accuracy with full spectrum. Though the proposed method achieved higher accuracy, it requires the hyperspectral camera with 324 spectral bands so the solution becomes too costly. Sharath D. M. et Al. developed the Bacterial Blight detection system for Pomegranate plant by using features such as color, mean, homogeneity, SD, variance, correlation, entropy, edges etc. Authors have implemented grab cut segmentation for segmenting the region of interest in the image [4]. Canny edge detector was used to extract the edges from the images. Authors have successfully developed a system which can predict the infection level in the fruit.

# Methodology

Dataset For this project we have used public dataset for plant leaf disease detection called PlantVillage curated by Sharada P. Mohanty et Al.
The dataset consists of 87000 RGB images of healthy and unhealthy plant leaves having 38 classes out of which We have selected only 25 classes for experimentation of our algorithm These classes are shown in Table

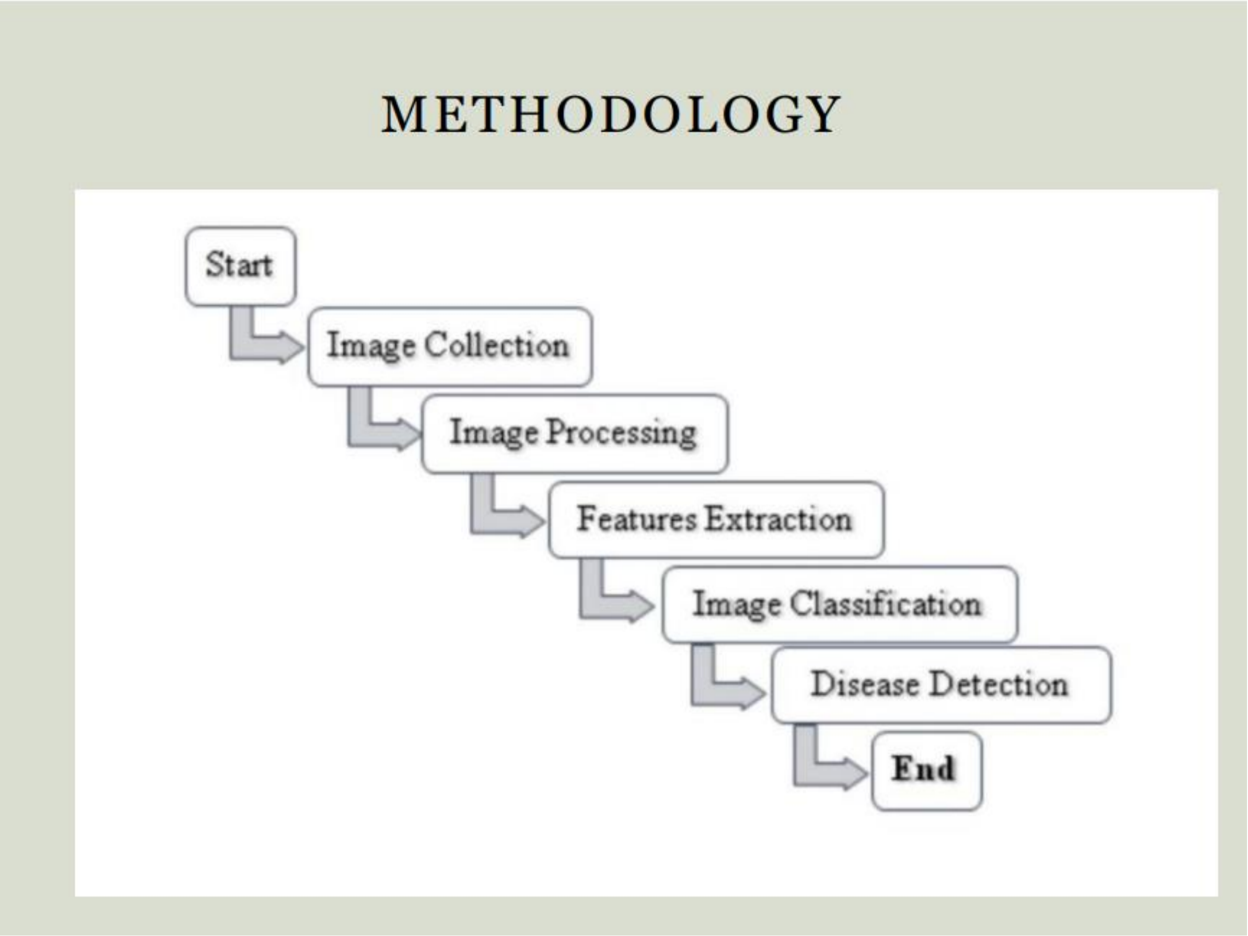


Table 1. Dataset Specifications.

| Plant | Disease Name | No. of Images |
|---|---|---|
| Apple | Healthy | 2008 |
| | Diseased Scab | 2016 |
| | Diseased: Black rot | 1987 |
| | Diseased: Cedar apple rust | 1760 |
| Corn | Healthy | 1859 |
| | Diseased: Cercospora leaf spot | 1642 |
| | Diseased: Common rust | 1907 |
| | Diseased: Northern Leaf Blight | 1908 |
| Grapes | Healthy | 1692 |
| | Diseased: Black rot | 1888 |
| | Diseased: Esca (Black Measles) | 1920 |
| | Diseased: Leaf blight (Isariopsis) | 1722 |
| Potato | Healthy | 1824 |
| | Diseased: Early blight | 1939 |
| | Diseased: Late blight | 1939 |
| Tomato | Healthy | 1926 |
| | Diseased: Bacterial spot | 1702 |
| | Diseased: Early blight | 1920 |
| | Diseased: Late blight | 1851 |
| | Diseased: Leaf Mold | 1882 |
| | Diseased: Septoria leaf spot | 1745 |
| | Diseased: Two-spotted spider mite | 1741 |
| | Diseased: Target Spot | 1827 |
| | Diseased: Yellow Leaf Curl Virus | 1961 |
| | Diseased: Tomato mosaic virus | 1790 |

LEAVES AND IT’S DIFFERENT WAYS

Data preprocessing and feature extraction

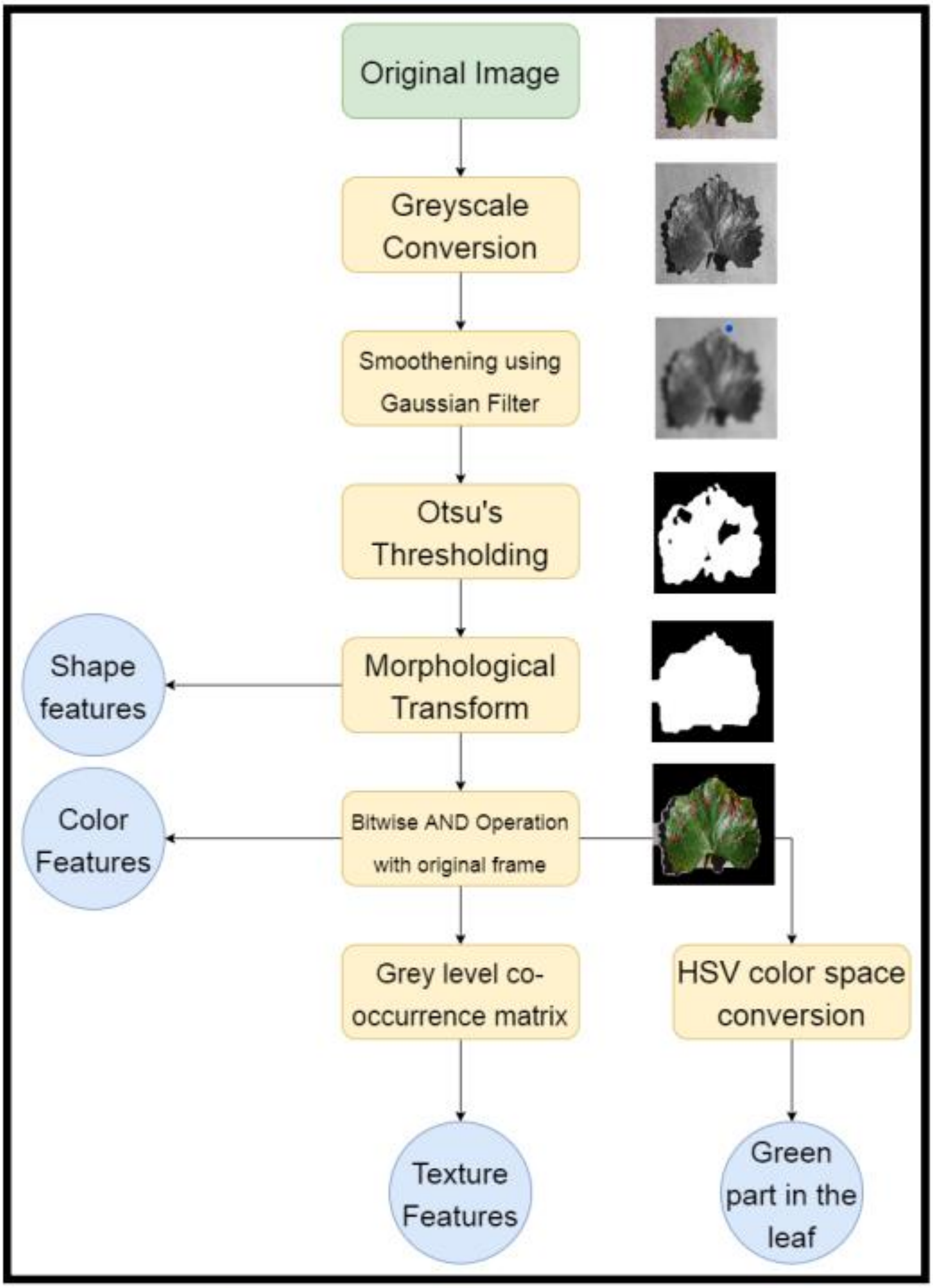



**Fig. 2.** Steps for data preprocessing and feature extraction.

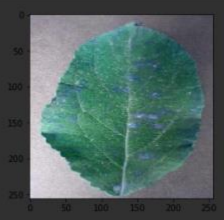
GLCM is the spacial relationship of pixels in the image. Extracting texture features from GCLM is one of the tradition method in computer vision. We have extracted following features from GCLM: • Contrast • Dissimilarity • Homogeneity • Energy • Correlation After extracting all the features from all the images in the dataset, feature selection task is performed.

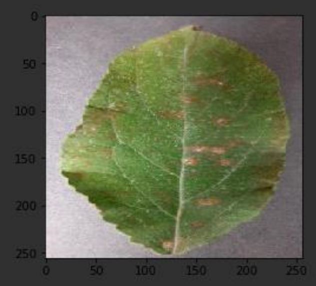
# Coding and testing

```
Requirement already satisfied: numpy in /opt/anaconda3/lib/python3.8/site-packages (1.16.5)
Collecting numpy
  Downloading numpy-1.24.2-cp38-cp38-macosx_10_9_x86_64.whl (19.8 MB)
K     |████████████████████████████████| 19.8 MB 3.9 MB/s eta 0:00:01     |████████████████          | 13.3 MB 4.2 MB/s eta 0:00:02
?25hInstalling collected packages: numpy
  Attempting uninstall: numpy
    Found existing installation: numpy 1.16.5
    Uninstalling numpy-1.16.5:
      Successfully uninstalled numpy-1.16.5
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.
tensorflow 2.12.0 requires numpy<1.24,>=1.22, but you have numpy 1.24.2 which is incompatible.
Successfully installed numpy-1.24.2
Note: you may need to restart the kernel to use updated packages.
```

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt
import numpy as np
import cv2
import matplotlib.pyplot as plt
from scipy import ndimage
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler
from matplotlib.colors import hsv_to_rgb
```

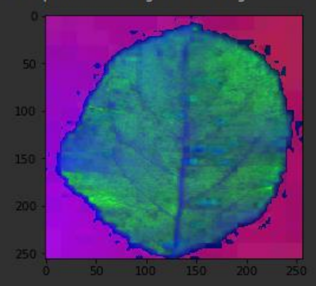
```python
img = cv2.imread('./image.jpg')
plt.imshow(img)
plt.show()
```



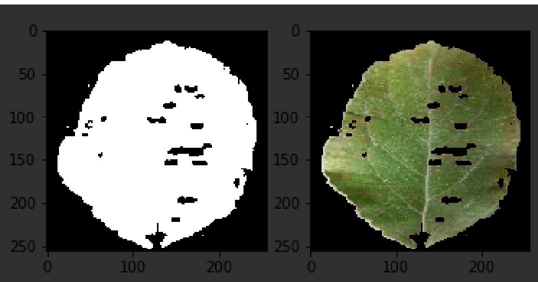


```python
hsv_img = cv2.cvtColor(img, cv2.COLOR_RGB2HSV)
plt.imshow(hsv_img)
```

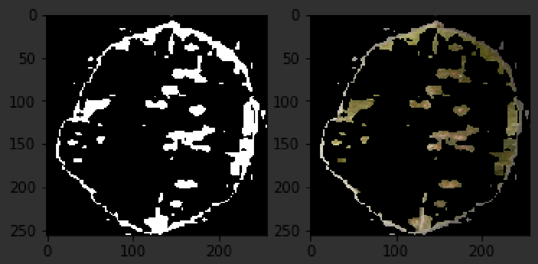
```
<matplotlib.image.AxesImage at 0x7f85f82aad90>
```



```python
lower_green = np.array([25,0,20])
upper_green = np.array([100,255,255])
mask = cv2.inRange(hsv_img, lower_green, upper_green)
result = cv2.bitwise_and(img, img, mask=mask)
plt.subplot(1, 2, 1)
plt.imshow(mask, cmap="gray")
plt.subplot(1, 2, 2)
plt.imshow(result)
plt.show()
```

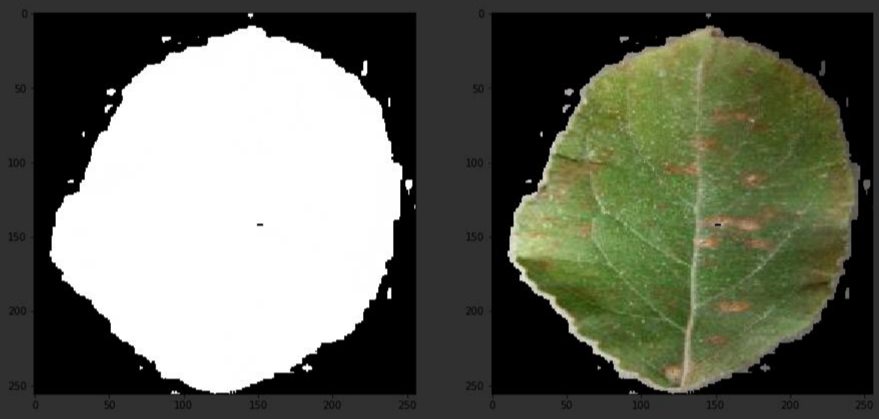



```
lower_brown = np.array([10,0,10])
upper_brown = np.array([30,255,255])
disease_mask = cv2.inRange(hsv_img, lower_brown, upper_brown)
disease_result = cv2.bitwise_and(img, img, mask=disease_mask)
plt.subplot(1, 2, 1)
plt.imshow(disease_mask, cmap="gray")
plt.subplot(1, 2, 2)
plt.imshow(disease_result)
plt.show()
```




```
final_mask = mask + disease_mask
final_result = cv2.bitwise_and(img, img, mask=final_mask)
plt.figure(figsize=(15,15))
plt.subplot(1, 2, 1)
plt.imshow(final_mask, cmap="gray")
plt.subplot(1, 2, 2)
plt.imshow(final_result)
plt.show()
```

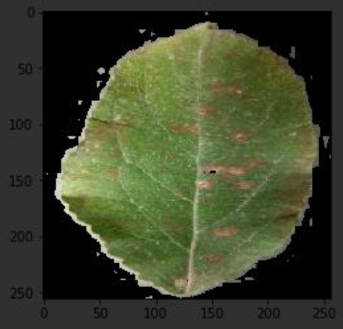
```
final_mask = mask + disease_mask
final_result = cv2.bitwise_and(img, img, mask=final_mask)
plt.figure(figsize=(15,15))
plt.subplot(1, 2, 1)
plt.imshow(final_mask, cmap="gray")
plt.subplot(1, 2, 2)
plt.imshow(final_result)
plt.show()
```




```
pip install opencv-contrib-python
```

```
Requirement already satisfied: opencv-contrib-python in /opt/anaconda3/lib/python3.8/site-packages (4.7.0.72)
Requirement already satisfied: numpy>=1.17.3 in /opt/anaconda3/lib/python3.8/site-packages (from opencv-contrib-python) (1.24.2)
Note: you may need to restart the kernel to use updated packages.
```

```
<matplotlib.image.AxesImage at 0x23acd6dff28>
```




```python
# global Feature
global_feature = des
```

```python
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
rescaled_features = scaler.fit_transform(global_feature)
```

```python
global_feature.shape
```

```
(217, 64)
```

```python
surf.descriptorSize()
```

```
64
```

```python
print("[STATUS] feature vector size {}".format(np.array(global_feature).shape))
```

```
[STATUS] feature vector size (217, 64)
```

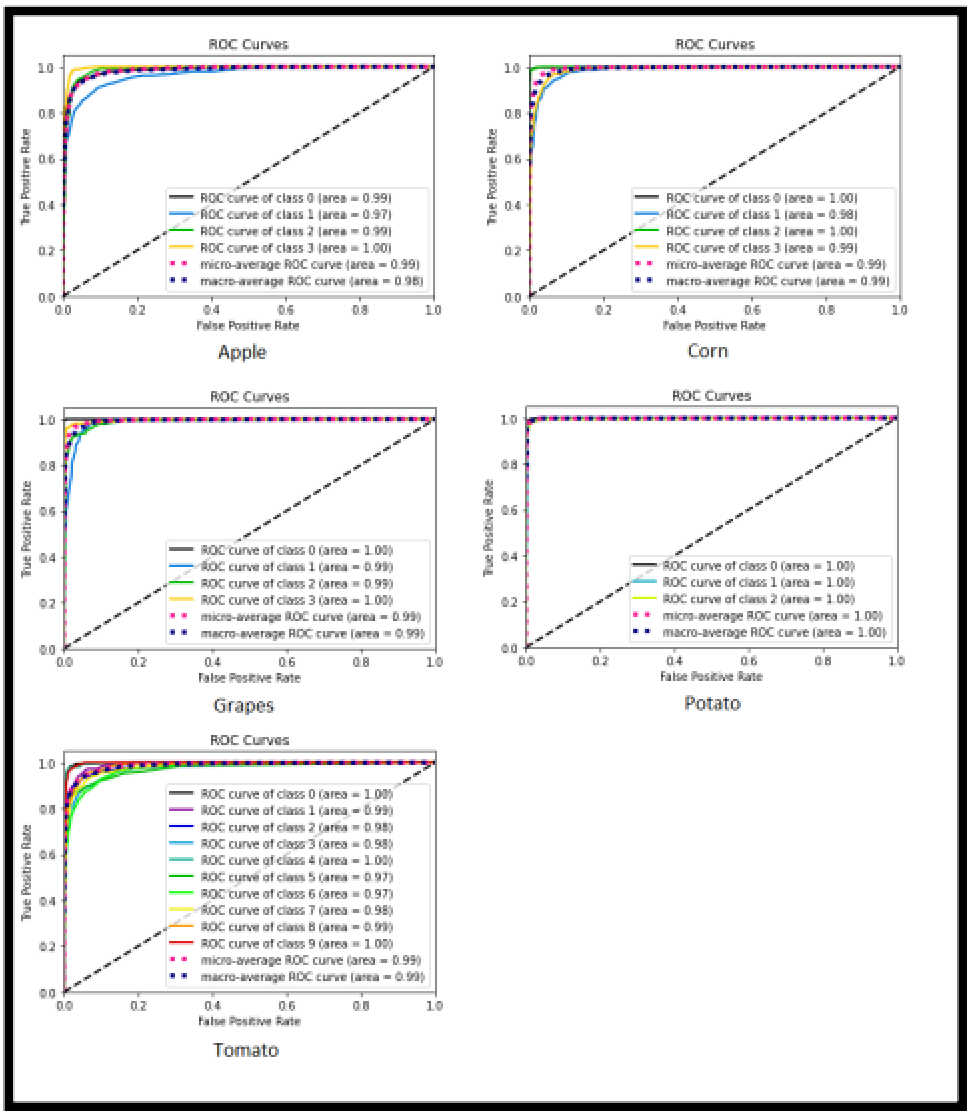
```python
a = global_feature
```

# Results and discussion

performance matrices for each model developed for each of the plant. We can observe that the accuracy scores are nearly equal to f1 scores. This is because of balanced number of false negative and false positive predictions. This is considered as best case for any machine learning algorithm. The average accuracy was 93%.

**Table 2.** Performance matric for all models.

| Plant | Accuracy | F1 Score |
|---|---|---|
| Apple | 0.91 | 0.91 |
| Corn | 0.94 | 0.94 |
| Grapes | 0.95 | 0.95 |
| Potato | 0.98 | 0.98 |
| Tomato | 0.87 | 0.87 |



# Conclusion

We have successfully developed a computer vision based system for plant disease detection with average 93% accuracy and 0.93 F1 score. Also the proposed system is computationally efficient because of the use of statistical image processing and machine learning model. Table 3 illustrates the overall benefits of our system over the other approaches.

# FUTURE SCOPE

As we are developing rest full webservices for development, In future same API can be used for mobile application and web application developement. Currently in proposed solution "disease detection and its possible remidies" will be provided. In future development of the project with use of IoT suggested fertiliser can be automatically supplied to plant and soil.

# REFERENCES

1. S. D. Khirade and A. B. Patil, "Plant Disease Detection Using Image Processing," 2015 International Conference on Computing Communication Control and Automation, 2015, pp. 768-771, doi: 10.1109/ICCUBEA.2015.153. 2. S. C. Madiwalar and M. V. Wyawahare, "Plant disease identification: A comparative study," 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), 2017, pp. 13-18, doi: 10.1109/ICDMAI.2017.8073478. 3. P. Moghadam, D. Ward, E. Goan, S. Jayawardena, P. Sikka and E. Hernandez, "Plant Disease Detection Using Hyperspectral Imaging," 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2017, pp. 1-8, doi: 10.1109/DICTA.2017.8227476. 4. S. D.M., Akhilesh, S. A. Kumar, R. M.G. and P. C., "Image based Plant Disease Detection in Pomegranate Plant for Bacterial Blight," 2019 International Conference on Communication and Signal Processing (ICCSP), 2019, pp. 0645-0649, doi: 10.1109/ICCSP.2019.8698007. 5. G. Shrestha, Deepsikha, M. Das and N. Dey, "Plant Disease Detection Using CNN," 2020 IEEE Applied Signal Processing Conference (ASPCON), 2020, pp. 109-113, doi: 10.1109/ASPCON49795.2020.9276722. 6. Mohanty SP, Hughes DP and Salathé M (2016) Using Deep Learning for Image-Based Plant Disease Detection. Front. Plant Sci. 7:1419. doi: 10.3389/fpls.2016.01419 7. R. M. Haralick, K. Shanmugam and I. Dinstein, "Textural Features for Image Classification," in IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-3, no. 6, pp. 610-621, Nov. 1973, doi: 10.1109/TSMC.1973.4309314. 8. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:10109334043\