

PLANT DISEASES DETECTION

A COURSE PROJECT REPORT

By

APARNA SURESH [RA2011026010099]

Under the guidance of

Dr.M.Maheswari

(Assistant Professor, CINTEL)

In partial fulfilment for the Course

of

**18CSE353T – DIGITAL IMAGE
PROCESSING**

in Computational Intelligence



FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR-603203

MAY 2023

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this mini project report "**Plant Disease Detection** " is the bonafide work of “**APARNA SURESH(RA2011026010099)**” who carried out the project work under my supervision.

SIGNATURE

**Dr. M Maheswari
Assistant Professor
Computational Intelligence**

SIGNATURE

**Dr.R.Annie Uthra
Head and Professor
Computational Intelligence**

ABSTRACT

To promote sustainable development, the smart city implies a global vision that merges artificial intelligence, big data, decision making, information and communication technology (ICT), and the Internet-of-Things (IOT). These processes above are related for solving real life problems. Food is one of the basic needs of human being. World population is increasing day by day. So, it has become important to grow enough crops to feed such a huge population. But with the time passing by, plants are affected with various kinds of diseases, which cause great harm to the agricultural plant productions.

Beside that many countries economy greatly depends on agricultural productivity and it's also a need for a country to attain agricultural productivity of basic agricultural product for the people of that particular country. Detection of plant disease through some automatic technique is beneficial as it requires a large amount of work of monitoring in big farm of crops, and at very early stage itself it detects symptoms of diseases means where they appear on plant leaves. In this paper surveys on different disease classification techniques that can be used for plant leaf disease detection.

INTRODUCTION

The agricultural land mass is more than just being a feeding sourcing in today's world. Indian economy is highly dependent of agricultural productivity. Therefore, in the field of agriculture, detection of disease in plants plays an important role. To detect a plant disease in the very initial stage, use of automatic disease detection techniques is beneficial. For instance a disease named little leaf disease is a hazardous disease found in pine trees in United States. The affected tree has stunted growth and dies within 6 years. Its impact is found in Alabama, Georgia parts of Southern US. In such scenarios early detection could have been fruitful. 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 the idea that they can contact 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. Automatic detection of diseases by just seeing the symptoms on the plant leaves makes it easier as well as cheaper. This also supports machine vision to provide image based automatic process control, inspection, and robot guidance.

Plant disease identification by 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 less effort, less time and become more accurate. In plants, some general diseases seen are brown and yellow spots, early and late scorch, and others are fungal, viral, and bacterial diseases. Image processing is used for measuring affected areas of disease and to determine the difference in the color of the affected area.

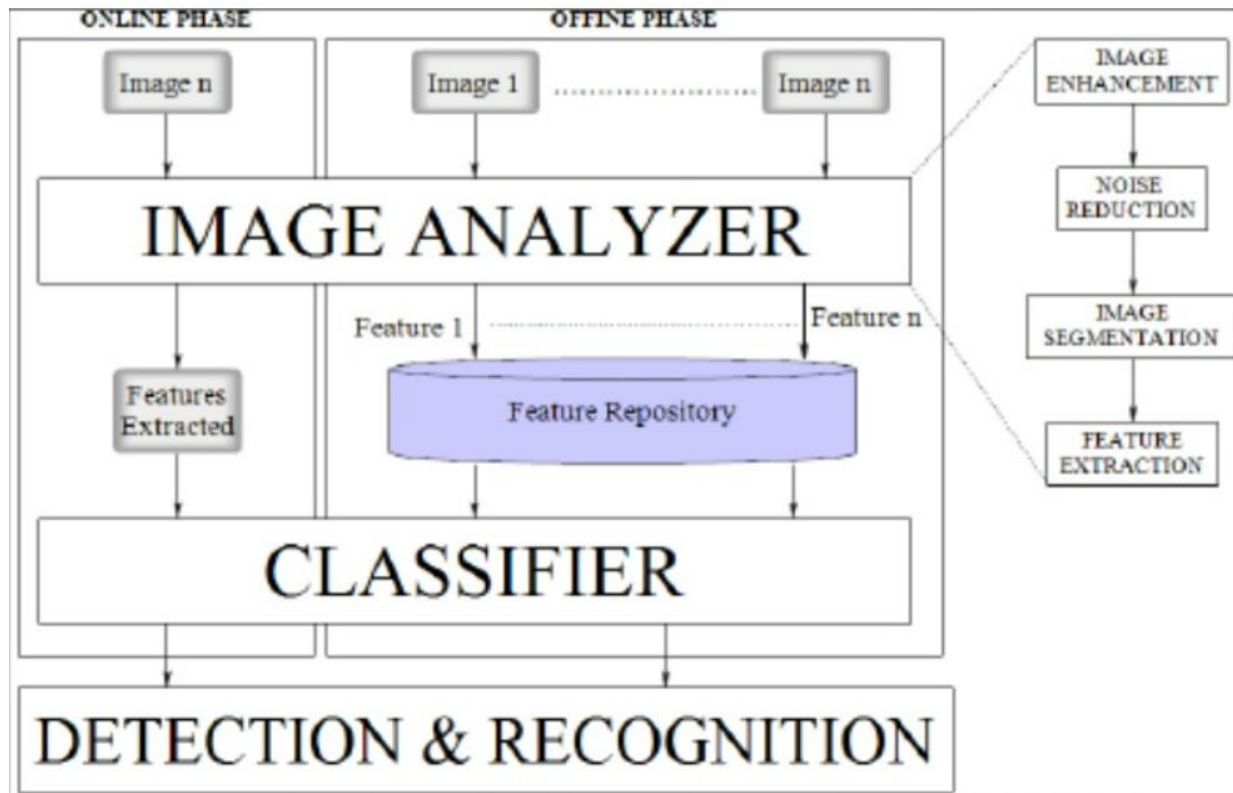
Image segmentation is the process of separating or grouping an image into different parts. There are currently many ways of performing image segmentation, ranging from the simple thresholding method to advanced color image segmentation methods. These parts normally correspond to something that humans can easily separate and view as individual objects. Computers have no means of intelligently recognizing objects, and so many different methods have been developed to segment images. The segmentation process is based on various features found in the image. This might be color information, boundaries, or segment of an image. We use Genetic algorithm for color image segmentation.

LITERATURE SURVEY

In paper [1] there are four steps. Out of them the first one is gathering image from several part of the country for training and testing. Second part is applying Gaussian filter is used to remove all the noise and thresholding is done to get all green color component. K-means clustering is used for segmentation. All RGB images are converted into HSV for extracting feature.

The paper [2] presents the technique of detecting jute plant disease using image processing. Image is captured and then it is realized to match the size of the image to be stored in the database. Then the image is enhanced in quality and noises are removed. Hue based segmentation is applied on the image with customized thresholding formula. Then the image is converted into HSV from RGB as it helps extracting region of interest. This approach proposed can significantly support detecting stem oriented diseases for jute plant.

ARCHITECTURE DIAGRAM



TECHNIQUES IMPLEMENTED IN DETAIL

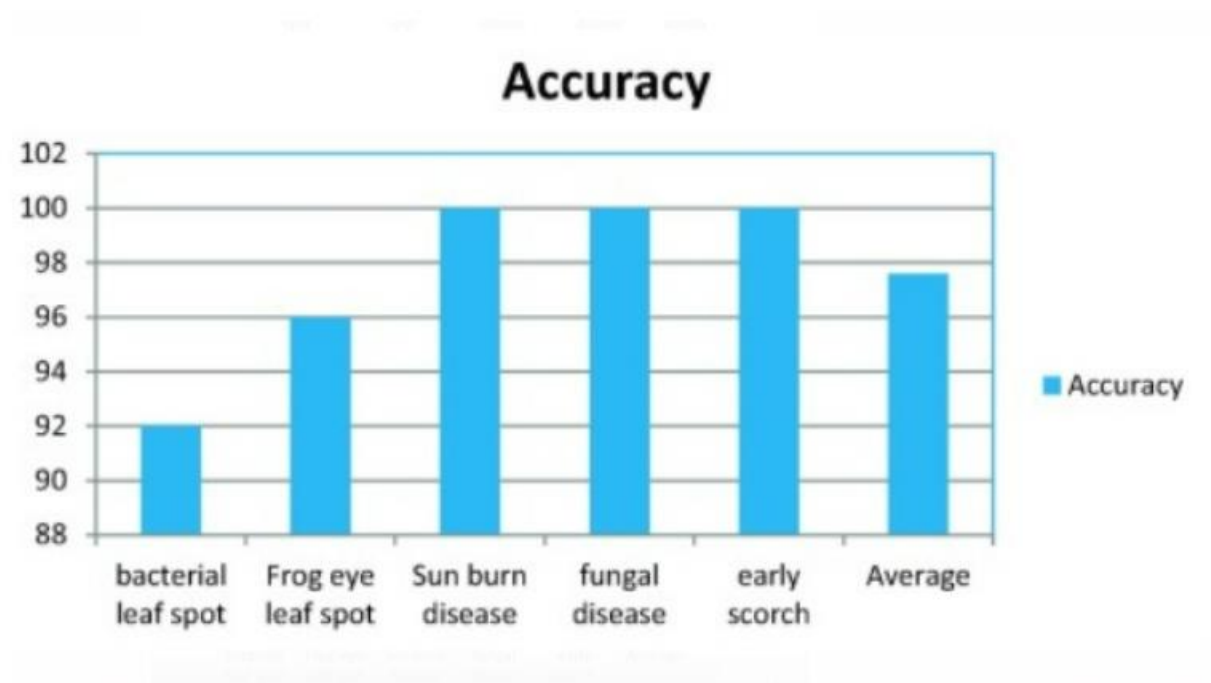
Genetic algorithms belong to the evolutionary algorithms which generate solutions for optimization problems. Algorithm begins with a set of solutions called population. Solutions from one population are chosen and then used to form a new population. This is done with the anticipation, that the new population will be enhanced than the old one. Solutions which are selected to form new solutions (offspring) are chosen according to their fitness – the more appropriate they are, the more probability they have to reproduce.

The basic steps of genetic algorithm are as follows:

- (1) [Start] Generate random population of n chromosomes (suitable solutions for the problem).
- (2) [Fitness] Evaluate the fitness $f(x)$ of each chromosome x in the population.
- (3) [New population] Create a new population by repeating following steps until the new population is complete.
 - (a) [Selection] Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected).
 - (b) [Crossover] With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
 - (c) [Mutation] With a mutation probability mutate new offspring at each locus (position in chromosome).
 - (d) [Accepting] Place new offspring in a new population.
- (4) [Replace] Use new generated population for a further run of algorithm.
- (5) [Test] If the end condition is satisfied, stop, and return the best solution in current population.
- (6) [Loop] Go to step 2.

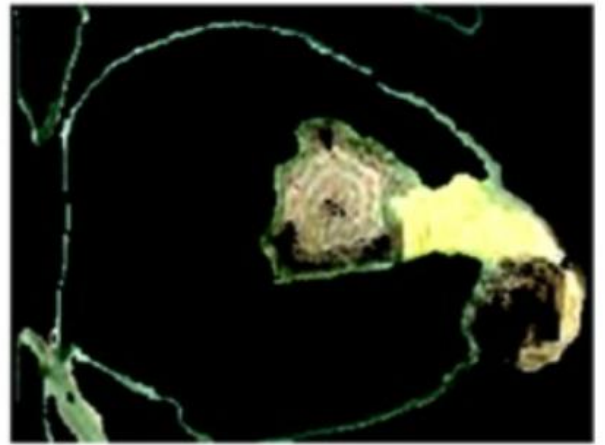
RESULT ACCURACY

Leaf disease	Bacterial leaf spot	Frog eye leaf spot	Sun burn disease	Fungal disease	Early scorch	Accuracy
Bacterial leaf spot	23	2	0	0	0	92
Frog eye leaf spot	1	24	0	0	0	96
Sun burn disease	0	0	25	0	0	100
Fungal disease	0	0	0	25	0	100
Early scorch	0	0	0	0	25	100
Average						97.6

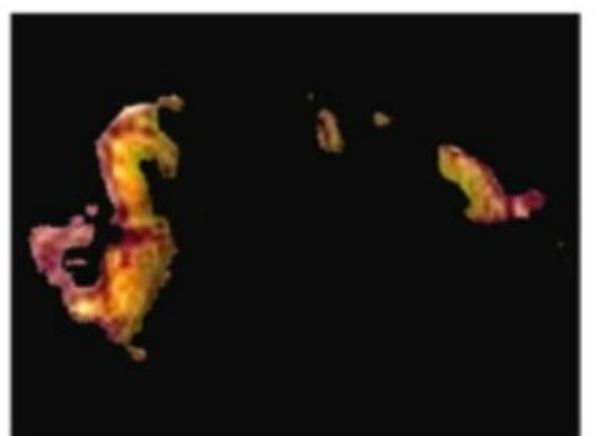
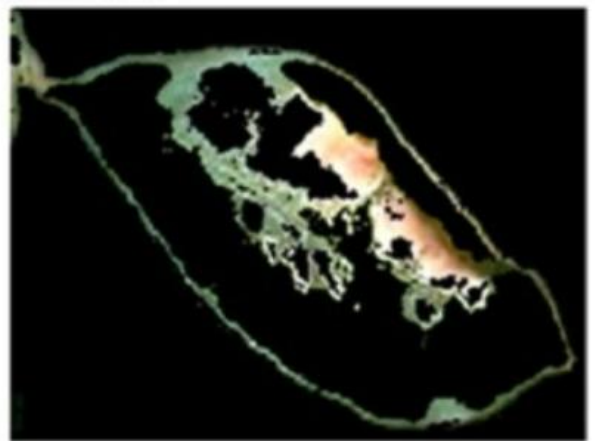


Images: Input image before applying and after applying, histogram.

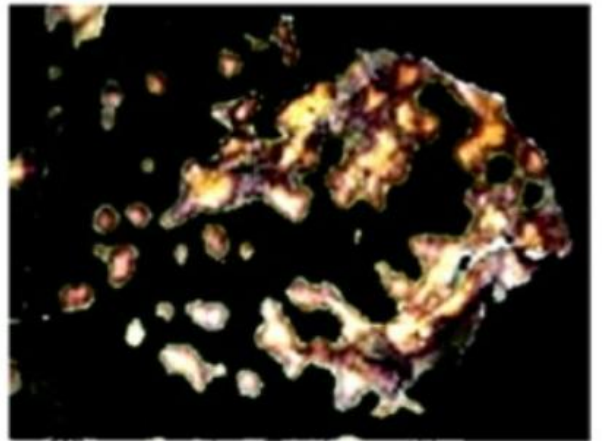
1. Fungal disease:



2.burn disease in lemon leaf:



2. Bacterial leaf disease:



SOURCE CODE

```
import pandas as pd
import cv2 as cv
import numpy as np
import matplotlib.pyplot as plt
import os
import seaborn as sns

DATASET="train"
DATASET2="valid"

CATEGORIES=["Tomato__Bacterial_spot","Tomato_Early_blight","Tomato_healthy","Tomato_Late_blight","Tomato_Leaf_Mold","Tomato_Septoria_leaf_spot","Tomato_Spider_mites",
            "Tomato_spotted_spider_mite","Tomato_Target_Spot","Tomato_Tomato_mosaic_virus","Tomato__Tomato_Yellow_Leaf_Curl_Virus"]

train_data=[]

for category in CATEGORIES:
    label=CATEGORIES.index(category)
    path=os.path.join(DATASET,category)
    for img_file in os.listdir(path):
        img=cv.imread(os.path.join(path,img_file),1)
        img=cv.cvtColor(img,cv.COLOR_BGR2RGB)
        img=cv.resize(img,(64,64))
        train_data.append([img,label])

test_data=[]

for category in CATEGORIES:
    label=CATEGORIES.index(category)
    path=os.path.join(DATASET2,category)
    for img_file in os.listdir(path):
        img=cv.imread(os.path.join(path,img_file),1)
        img=cv.cvtColor(img,cv.COLOR_BGR2RGB)
        img=cv.resize(img,(64,64))
        test_data.append([img,label])

print(len(train_data))
print(len(test_data))

import random
```

```

random.shuffle(train_data)
random.shuffle(test_data)

for lbl in train_data[:10]:
    print(lbl[1])

X_train=[]
y_train=[]

for features,label in train_data:
    X_train.append(features)
    y_train.append(label)

Y=[]
for i in y_train:
    if i==0:
        Y.append("BACTERIAL SPOT")
    elif i==1:
        Y.append("EARLY BLIGHT")
    elif i==2:
        Y.append("HEALTHY")
    elif i==3:
        Y.append("LATE BLIGHT")
    elif i==4:
        Y.append("LEAF MOLD")
    elif i==5:
        Y.append("SEPTORIA LEAF SPOT")
    elif i==6:
        Y.append("SPIDER MITE")
    elif i==7:
        Y.append("TARGET SPOT")
    elif i==8:
        Y.append("MOSAIC VIRUS")
    else:
        Y.append("YELLOW LEAF CURL VIRUS")

len(X_train),len(y_train)

X_test=[]
y_test=[]

for features,label in test_data:

```

```

X_test.append(features)
y_test.append(label)

Z=[]
for i in y_test:
    if i==0:
        Z.append("BACTERIAL SPOT")
    elif i==1:
        Z.append("EARLY BLIGHT")
    elif i==2:
        Z.append("HEALTHY")
    elif i==3:
        Z.append("LATE BLIGHT")
    elif i==4:
        Z.append("LEAF MOLD")
    elif i==5:
        Z.append("SEPTORIA LEAF SPOT")
    elif i==6:
        Z.append("SPIDER MITE")
    elif i==7:
        Z.append("TARGET SPOT")
    elif i==8:
        Z.append("MOSAIC VIRUS")
    else:
        Z.append("YELLOW LEAF CURL VIRUS")

len(X_test),len(y_test)

X_train=np.array(X_train).reshape(-1,64,64,3)
X_train=X_train/255.0
X_train.shape

X_test=np.array(X_test).reshape(-1,64,64,3)
X_test=X_test/255.0
X_test.shape

order=['BACTERIAL SPOT','EARLY BLIGHT','HEALTHY','LATE BLIGHT','LEAF
MOLD','SEPTORIA LEAF SPOT','SPIDER MITE','TARGET SPOT','MOSAIC
VIRUS','YELLOW LEAF CURL VIRUS']

ax=sns.countplot(Y, order=order)
ax.set_xlabel("Leaf Diseases")
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha='right')

```

```
ax.set_ylabel("Image Count")
```

```
ax=sns.countplot(Z, order=order)
ax.set_xlabel("Leaf Diseases")
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha='right')
ax.set_ylabel("Image Count")
```

```
from keras.utils import to_categorical
```

```
one_hot_train=to_categorical(y_train)
one_hot_train
```

```
one_hot_test=to_categorical(y_test)
one_hot_test
```

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D,Dense,Flatten,MaxPooling2D,Dropout
```

```
classifier=Sequential()
```

```
classifier.add(Conv2D(32,(3,3), input_shape=(64,64,3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))
classifier.add(Dropout(0.2))
```

```
classifier.add(Conv2D(64,(3,3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))
classifier.add(Dropout(0.2))
```

```
classifier.add(Conv2D(128,(3,3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))
classifier.add(Dropout(0.4))
```

```
classifier.add(Flatten())
```

```
classifier.add(Dense(activation='relu', units=64))
classifier.add(Dense(activation='relu', units=128))
classifier.add(Dense(activation='relu', units=64))
classifier.add(Dense(activation='softmax', units=10))
```

```
classifier.compile(optimizer='adam', loss='categorical_crossentropy',
metrics=['accuracy'])
```

```
classifier.summary()
```

```
hist=classifier.fit(X_train,one_hot_train,epochs=75,batch_size=128,validation_split=0.2)
```

```
test_loss,test_acc=classifier.evaluate(X_test,one_hot_test)
test_loss,test_acc
```

```
plt.plot(hist.history['loss'])
plt.plot(hist.history['val_loss'])
plt.title('Classifier Loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train','Validation'],loc='upper right')
plt.show()
```

```
plt.plot(hist.history['accuracy'])
plt.plot(hist.history['val_accuracy'])
plt.title('Classifier Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train','Validation'],loc='upper left')
plt.show()
```

```
y_pred=classifier.predict_classes(X_test)
y_pred
```

```
y_prob=classifier.predict_proba(X_test)
y_prob
```

```
from sklearn.metrics import roc_curve, auc
```

```
fpr = {}
tpr = {}
thresh = {}
roc_auc = {}
```

```
n_class = 10
```

```
for i in range(n_class):
    fpr[i], tpr[i], thresh[i] = roc_curve(y_test, y_prob[:,i], pos_label=i)
    roc_auc[i] = auc(fpr[i], tpr[i])
```

```
plt.plot(fpr[0], tpr[0], color='orange',label='Bacterial Spot AUC = %0.3f % roc_auc[0])
plt.plot(fpr[1], tpr[1], color='green',label='Early Blight AUC = %0.3f % roc_auc[1])
```



```
plt.plot(fpr[2], tpr[2], color='blue',label='Healthy AUC = %0.3f % roc_auc[2])
plt.plot(fpr[3], tpr[3], color='red',label='Late Blight AUC = %0.3f % roc_auc[3])
plt.plot(fpr[4], tpr[4], color='pink',label='Leaf Mold AUC = %0.3f % roc_auc[4])
plt.plot(fpr[5], tpr[5], color='purple',label='Septoria Leaf Spot AUC = %0.3f %
roc_auc[5])
plt.plot(fpr[6], tpr[6], color='brown',label='Spider Mites AUC = %0.3f % roc_auc[6])
plt.plot(fpr[7], tpr[7], color='cyan',label='Target Spot AUC = %0.3f % roc_auc[7])
plt.plot(fpr[8], tpr[8], color='yellow',label='Mosaic Virus AUC = %0.3f % roc_auc[8])
plt.plot(fpr[9], tpr[9], color='black',label='Yellow Leaf Curl Virus AUC = %0.3f %
roc_auc[9])
plt.title('Tomato Leaves Diseases ROC curve')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive rate')
plt.legend(loc='best')
```

```
from sklearn.metrics import confusion_matrix
```

```
sns.heatmap(confusion_matrix(y_test,y_pred))
cm=confusion_matrix(y_test,y_pred)
```

CONCLUSION

This paper presents the survey on different 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. 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. 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 plant diseases can be identified at early stage or the initial stage. To improve recognition rate in classification process Artificial Neural Network, Bayes classifier, Fuzzy Logic and hybrid algorithms can also be used.

REFERENCE

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

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)