FRUIT OR VEGETABLE DISEASE DETECTION (USING K-MEANS CLUSTERNIG)

A PROJECT REPORT

Submitted in partial fulfillment for the award of the course

DIGITAL IMAGE PROCESSING (SWE1010)

Submitted by

STUDENT NAME : G KARTHIKEYAN 16MIS0387

School of Information Technology & Engineering

CONTENT:

ABSTRACT:-

Identification of the fruits or vegetable diseases is the key providing the losses in the yield and quantity of the agricultural product. The studies of the fruits or vegetable diseases mean the studies of visually observable patterns seen on the plant. Health monitoring and diseases detection on plant is very critical for sustainable agricultural. It is very difficult to monitor the fruits or vegetable diseases manually. It requires tremendous amount of work, expertize in the fruits or vegetable diseases, and also require the excessive processing time. In this project we discussed the methods used for detection of fruits or vegetable diseases using their fruits or vegetable images.

I. INTRODUCTION:-

Fruit disease detection is an art as well as science. The detection process is inherently visual and requires intuitive judgment as well as the use of scientific method.

Fig 1: Samples of diseased fruits or vegetable

Photographic images of symptoms and signs of fruit diseases as shown in Figure 1 are extensively used to enhance description of diseases on fruits or vegetable which are invaluable in research, teaching and diagnostics etc.

Farmers are very much concerned about the huge costs involved in these activities. Automatic identification and classification of diseases based on their particular symptoms are very useful to farmers and also to agriculture scientists.

Early detection of diseases is a major challenge in agriculture science. Fruit diseases have turned into a dilemma as it can cause significant reduction in both quality and quantity of agricultural products. The naked eye observation of experts is the main approach adopted in practice for detection and identification of fruit diseases. Further, in some developing countries, farmers may have to go long distances to contact experts, this makes consulting experts too expensive and time consuming.

II. RELATED WORKS OR LITERATURE SURVEY

In this section describes various approaches for detecting the disease in fruits or vegetable using image processing technique

[1] prof. Sanjay, B. Dhaygude& et al…The application of texture statistics for detecting the fruits or vegetable disease has been explained Firstly by color transformation structure RGB is converted into HSV (herpes simplex virus) space because HSV is a good color descriptor. Masking and removing of green pixels with pre-computed threshold level. Then in the next step segmentation is performed using 32X32 patch size and obtained useful segments. These segments are used for texture analysis by color co-occurrence matrix. Finally if texture parameters are compared to texture parameters of normal fruit.

[2] Amandeep Singh, Maninder Lal Singh& et al. .The most significant challenge faced during the work was capturing the quality images with maximum detail of the fruit color. It is very typical task to get the image with all the details within a procesable memory. Such images are formed a through high resolution and thus are of 6-10MB of size. Second challenge faced was to get illumination conditions as from the start to the end of paddy crop season, illumination varies a lot even when the image acquiring time is fixed.

[3] Malvika Ranjan, Manasi Rajiv Weginwar& et al…Image of diseased fruits or vegetable is captured .As the result of segmentation Color HSV features are extracted. Artificial neural network (ANN) is then trained to stinguish the healthy and diseased samples. ANN classification performance is 80% better in accuracy.

[4] P.Revathi, M.Hemalatha& et al…This proposed work is based on Image Edge detection Segmentation techniques in which, the captured images are processed for enrichment first. Then R, G, B color Feature image segmentation is carried out to get target regions image features such as boundary, shape, color and texture are extracted for the disease spots to recognize diseases and control the pest recommendation. Edge detection based Image segmentation, analysis and classification of disease.

[5] Heeb Al Bashish, Malik Braik & et al…In this project an image-processing-based approach is proposed and used for disease detection in fruits or vegetable. We test our program on five diseases which effect on the fruits or vegetable; they are bacterial, fungal, viral, nematode and abiotic.

[6] Mr. Pramod S. landge, Sushil A. Patil& et al… In this propose and experimentally evaluate a software solution for automatic detection and classification of fruits or vegetable diseases through Image Processing. This paper addresses this problem with the objective of developing image processing algorithms that can recognize problems in crops from images, based on color, texture and shape to automatically detect diseases or other conditions that might affect crops and give the fast and accurate solutions to the farmer with the help of SMS. The design and implementation of these technologies will greatly in selective chemical application, reducing costs and thus leading to improved productivity, as well as improved produce.

[7] Sachin D. Khirade & et al… Identification of the fruits or vegetable diseases is the key to preventing the losses in the yield and quantity of the agricultural product. It requires high amount of work, expertize in the fruits or vegetable diseases, and also require the excessive processing time. Hence, image processing is used for the detection of fruits or vegetable diseases. Disease detection involves the steps like image acquisition, image pre-processing, image segmentation, feature extraction and classification. This project discussed the methods used for the detection of fruits or vegetable diseases. This paper also discussed some Feature extraction and classification techniques to extract the features of infected diseases and the classification of fruits or vegetable diseases. The accurately detection and classification of the plant disease is very important for the successful cultivation of crop and this can be done using image processing. This project discussed various techniques to segment the disease part of the fruits or vegetable.

[8] M.Malathi, K.Aruli & et al… They provides survey on fruits or vegetable disease detection using image processing techniques. Disease in crops causes to reduction in quantity and quality of the agricultural product. Identification of symptoms of disease is difficult for farmer. Depending on the applications, many image processing technique has been introduced to solve the problems by pattern recognition and some automatic classification tools. There are many methods in automated or computer vision for disease detection and classification but still there is lack in this research topic. All the disease cannot be identified using single method. Crop protection especially in large farms is done by using computerized image processing technique. That can be detect diseased fruits or vegetable using color information of fruits or vegetable.

[9] Y.Sanjana, AshwathSivasamy& et al… it describes the uploaded pictures captured by the mobile phones are processed in the remote server and presented to an expert group for their opinion. A simple color difference based approach is followed for segmentation of the disease affected lesions. The goal of this research is to develop an image recognition system that can recognize crop diseases. Image processing starts with the digitized color image of disease fruits or vegetable. A method of mathematics morphology is used to segment these images. The system allows the expert to evaluate the analysis results and provide feedbacks to the farmers

through a notification to their mobile phones. Then texture, shape and color features of color image of disease spot on fruits or vegetable were extracted, and a classification method of membership function was used to discriminate between the three types of diseases.

[10] Nitya subramanium et al. says the use of polarization information for surface inspection. The authors have considered the problem of detecting regions of damage and disease in the skins of different type of fruits or vegetable. The authors have used moments to estimates the components of the polarization image (mean intensity polarization and phase) from images obtained with multiple polarizer angles. The authors have used the normalized cut method to segment surface into different regions depending on their surface reflection properties.

[11] kaiyi wang, et al it designed and developed a novel method for identification fruits or vegetable disease and this method was based on the current computer vision and image processing methods. A new extraction and classification algorithm was used to recognize fruits or vegetable from images. To deal with the area of adhesion, a mathematical morphology algorithm was used for separating the objects. The proposed method was implemented on mobile smart devices and tested with field experiments. The experimental results showed good recognition performance with high efficiency.

[12] k. dang et al ,developed a method for detection of disease in fruits or vegetable. This author was achieve this goal, the techniques used was image processing through the deployment of WSMN in the cultivated field. They proposed a fruits or vegetable disease detection approach which was designed to run on the resource constrained WMSN nodes. This new approach was able to make a preliminary local decision on the health condition of the fruits or vegetable and determine the necessity of sending back images to the control center for further inspection, thus improving the efficiency of the monitoring network. The complete method included image segmentation based on both color and shape and used 2D histogram as the feature for classification. Experiments on the fruits or vegetable image with nutrient efficiency symptoms showed the classification accuracy of 87.5%.

[13] H kalkan et al. proposed and developed a method an LDM-based feature extraction and selection and selection algorithm for the analysis of hyper spectral data along the spectral and spatial frequency the algorithm was implemented on consecutive multispectral image. The developed algorithm extracted the relevant features dimension and the corresponding data by pruning in feature space. The developed algorithm was tested on detection of contaminated red chili papers.

[14] Satish Madhgoria, MarekSchikora& et al… Proposed automatic pixel based classification method for detecting unhealthy regions in fruits or vegetable images is presented. The algorithms have been tested extensively. Linear SVM has been used to classify each pixel. We have also shown hoe the results from SVM could be improved remarkably using the neighborhood check technique. The presented algorithm could well extended for other detection tasks which also mainly rely on color information, but extension to other features is easily possible. The task is performed in three steps. First, we perform segmentation to divide the image into foreground and background. In the second step, support vector machines are applied to predict the class of each pixel belonging to the foreground. And finally, we do further refinement by neighborhood-check to omit all falsely-classified pixels from second step.

[15] Bhumika S.Prajapati, Vipul K.Dabhi& et al… The survey on background removal and segmentation techniques was discussed. Through this survey, we concluded that for background removal color space conversion from RGB to HSV (HERPES SIMPLEX VIRUS) is useful. We also found that thresholding technique gives good result compared to other background removal techniques. We performed color segmentation by masking green pixels in the background removed image and then applying thresholding on the obtained masked image to get binary image. This is useful to extract accurate features of disease. We found that

SVM (support vector machine) gives good results, in terms of accuracy, for classification of diseases. There are five major steps in our proposed work, out of which three steps have been implemented: Image Acquisition, Image pre-processing, and Image segmentation.

[16] The behavior of classifiers for identification and classification of fungal disease symptoms affected on fruits or vegetable. Fungal disease symptoms are early blight, and late blight affected on specific type of fruits or vegetable crop are considered for recognition classification.

[17] The author shiv ram dubey suggested an image processing based way for detection and identification of fruit disease. The fruits or vegetable selected is tomato and disease considered are namely tomato rot, tomato blotch for conducting the experiments. For image segmentation, k-means clustering is used. Color coherence vector, histogram, local binary patterns, complete local binary patterns are used to extracting the features. for fruits or vegetable disease detection, multiclass support vector machine is used.

[18] monica jhuria –provided an approach for fruits or vegetable disease detection based on image processing. The purpose of research work is to detect disease on fruit. Grapes, tomato and mangoes are selected for conducting experiments. Morphology, color and texture features vectors are chosen for feature extraction. Morphology feature gives 90% accurate results than other feature vectors. For disease detection and weight calculation of fruit image processing techniques are used. Back propagation is used for weight adjustment of images that are stored in learning database. On the basis of disease spreading, the grading of fruits or vegetable has been decided.

[19] Sandeep Kumar k s, shiva Kumar g-has discussed about the image segmentation process can be used to detect the fruits or vegetable. Image segmentation is a process that partitions an image into its objects. Effective segmentation of complex images is one of the most difficult tasks in image processing. Various image segmentation algorithm have been proposed to archive efficient and accurate results.

[20] juan Ignacio explained about detecting disease in fruits or vegetable. They said four main stages. Segmentation based on RGB color space is performed. From segmented image, features are detected and extracted. Discriminable set of features are selected. Posterior probability model selection (PPMS) algorithm is also used for complexity selection of fruits or vegetable in an image.

[21] Rob.J. Mullen focus on edge pattern extraction to image feature extraction. This techniques help to increase performance of the algorithm and also eliminate the requirement for a user set threshold. The algorithm allows to adapt autonomically an appropriate threshold for a given image or data set. This approach is extended for simultaneously multiple feature extraction and dynamic adaptation to changing imagery. fruit Disease Detection using Color, Texture Analysis and ANN . K-means clustering, ANN . Accuracy of this technique is 85% to detect the fruit disease.

.

[22] Machine learning based detection and recognition of fruit diseases can provide clues to identify and treat the diseases in its early stages. Comparatively, visually identification of diseases is expensive, inefficient, and difficult. One of the main health benefits of fruits or vegetable is their high nutrient content. Fruits or vegetable contain vitamins and minerals that contribute to growth and the maintenance of good health. Fruits or vegetable contain a variety of nutrients including vitamins, minerals and antioxidants. Eating the recommended amount of fruits or vegetable each day can reduce the risk of chronic diseases. There are different kinds of fruits or vegetable. Fruit quality is frequently attributed to size, shape, mass, firmness, color and bruises. Based on such parameters, fruits or vegetable can be classified and sorted. The diseases need to be controlled in the primary stage of the infection .Today’s farmers are not able to identify the diseases at the early stages due to lack of knowledge of infections and diseases that can attack the crop.

[23] Rohit ranjan – the disease detection system operates on the images. Image acquisition is to collection the data from the user. Image pre-processing is used to remove the noise from the image, some techniques are used to remove the noise from the image such as filter, crop, resize. Image segmentation is used to dividing an image into many meaningful parts. The techniques used to segmentation is k-means clustering, Otsu threshold method. Feature extraction gives very good result for identification of disease from fruits or vegetable. And the final step is classification is used to identify and it is done by using support vector machine.Classification of Anthracnose Fungal Disease of fruit based on Statistical Texture Features . Thresholding algorithm is used to segmentation, run length matrix is used to extract feature, ANN is used to classification. The average accuracy of 84.56% for normal type and 76.67% for anthracnose affected type.

[24] R. vidhya , S sabita-in this paper the detection of fruits or vegetable disease is proposed and evaluated by using complete local binary pattern. This approach involve three methods k-means clustering techniques which is used for image segmentation, in which image features are extracted from the segmented image , finally the images are trained and classified by using multi-class support vector machine. This proposed system is achieved 93% accuracy.

[25] Image processing technique for tomato fruit disease detection-in paper , the image acquisition is getting the input image, pre-processing convert the original image into binary image also display color threshold ranges by computing histogram of image. The segmentation is done by watershed algorithm, feature are extracted from segmented images using blob extraction method, finally the disease is classified using normalized correlation method, at last fruit grading is performed for detecting the disease fruits or vegetable. Smart Farming: tomato Disease Detection Using Image Processing
K-means clustering algorithm, ANN is used for classification. The test image is obtained from the user.
The overall accuracy of this method is 82%

III. PRELIMINARY CONCEPTS

(1)IMAGE ACQUISITION:-image

Image acquisition is collecting data set of the disease infected images. The data set collection is collected from the internet. The input image of the fruit is taken from the digital camera having high resolution and keeping specific distance between digital camera and required fruits or vegetable

(2)IMAGE PREPROCESSING:-

The image preprocessing is used to remove the unwanted noise from the image of fruits or vegetable. It also perform histogram equalization on image to distribute the intensities to increase the quality of the image. Some techniques are used to pre-processing such as filter image, crop image, resize image**.**

(3)IMAGE SEGMENTATION:-

It is the method of dividing an image into many meaningful parts. Segmentation of images is done to change the representation or simplify the image in easier form. Image segmentation can be done by some techniques such as k-means clustering algorithm, otsu threshold method and watershed and etc…

K – means clustering:

k – means clustering used to segment the disease fruit image. In this project we are need to segment the image so we convert the RGB image into L*a*b color image. After converting the image in to L*a*b color space image we apply the k – means clustering with 3 clusters. And we can choose suitable segmented image for feature extraction

$$\text{Accuracy} = \frac{\text{No. of correct output}}{\text{Total no. of image tried}} \times 100$$

K – means clustering algorithm:

1. Define number of clusters (k -> value).

2. Choose centroid. In function it will choose the centroid itself.

3. Cluster the image points based on the distance of their intensity values from the centroid intensity values.

$$c^{(i)} = \arg\min_j \| x^{(i)} - \mu_j \|^2$$

4. Compute new centroid for each cluster.

$$\mu_i = \frac{\sum_{i=1}^{m} 1\{c_{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c_{(i)} = j\}}$$

5. Repeat the step 3 and 4, stop when no more new assignment.

.

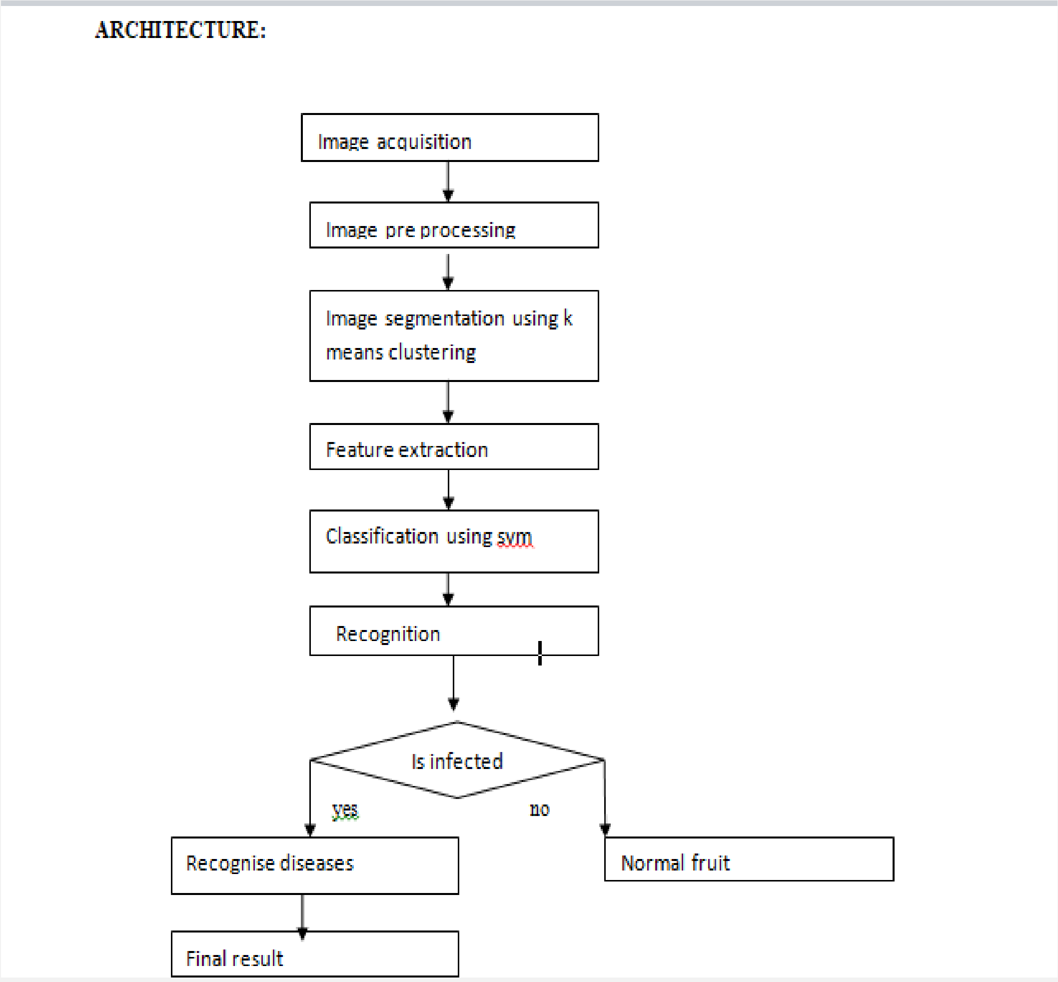
(4)FEATURE EXTRACTION:-

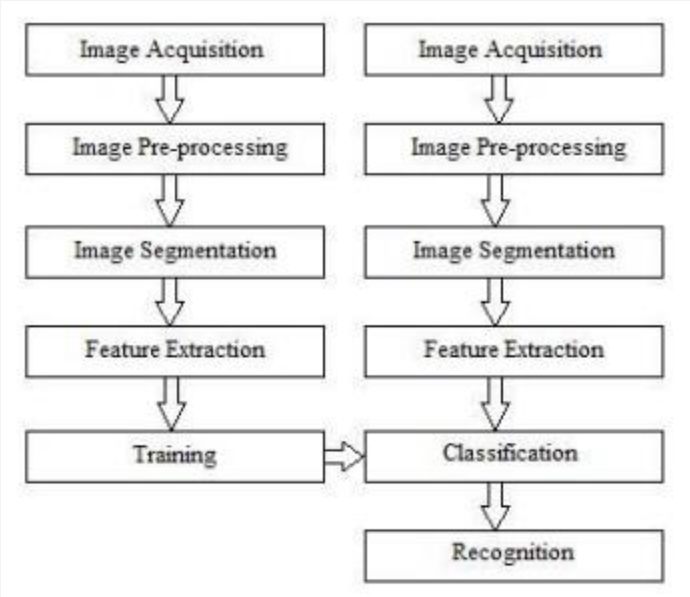
The feature extraction gives very good result for identification of disease from image. The feature extraction is used in many application of image processing. The feature extraction is used to reduce the large input data to small data so that it will take less time to process data but in extracted the feature must have important data to be process. The feature extraction can be done by using morphology, color, edges, texture and etc…

(5)CLASSIFICATION:-

This phases used to classify the disease of the image into one of the class whether it is infected or not. It is done by using support vector machine (SVM).svm perform supervised learning for classification. Marked the images to one of the two categories. A svm training algorithm makes a model that assigns new example into one categories or other. Some other techniques also used to classify the image defect such as artificial neural network (ANN).

IV. PROPOSED ARCHITECTURE:-





V.CODE:

Veg.m

```
function varargout = veg(varargin)
% VEG MATLAB code for veg.fig
%      VEG, by itself, creates a new VEG or raises the existing
%      singleton*.
%
%      H = VEG returns the handle to a new VEG or the handle to
%      the existing singleton*.
%
%      VEG('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in VEG.M with the given input arguments.
%
%      VEG('Property','Value',...) creates a new VEG or raises the
%      existing singleton*.  Starting from the left, property value pairs
are
%      applied to the GUI before veg_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to veg_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help veg

% Last Modified by GUIDE v2.5 03-Nov-2017 10:57:34

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
```

```
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @veg_OpeningFcn, ...
                   'gui_OutputFcn',  @veg_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State,varargin{:});
end

% End initialization code - DO NOT EDIT


% --- Executes just before veg is made visible.
function veg_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to veg (see VARARGIN)

% Choose default command line output for veg

handles.output = hObject;
handles.q=1;
ss = ones(300,400);
axes(handles.axes1);
imshow(ss);
axes(handles.axes2);
imshow(ss);
axes(handles.axes3);
imshow(ss);


% Update handles structure
guidata(hObject, handles);

% UIWAIT makes veg wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = veg_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
```

```matlab
% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc
[filename, pathname] = uigetfile({'*.*';'*.bmp';'*.jpg';'*.gif'}, 'Pick a
Fruit Image File');
I = imread([pathname,filename]);
pI = imresize(I,[256,256]);
I2 = imresize(I,[300,400]);
axes(handles.axes1);
imshow(I2);
title('Query Image');
ss = ones(300,400);
axes(handles.axes2);
imshow(ss);
axes(handles.axes3);
imshow(ss);
handles.ImgData1 = I;
guidata(hObject,handles);



% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
I3 = handles.ImgData1;
I4 = imadjust(I3,stretchlim(I3));
I5 = imresize(I4,[300,400]);
axes(handles.axes2);
imshow(I5);title(' Contrast Enhanced ');
handles.ImgData2 = I4;
guidata(hObject,handles);


% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
I6 = handles.ImgData2;
I = I6;
%% Extract Features

% Color Image Segmentation
% Use of K Means clustering for segmentation
% Convert Image from RGB Color Space to L*a*b* Color Space
% The L*a*b* space consists of a luminosity layer 'L*', chromaticity-layer
'a*' and 'b*'.
% All of the color information is in the 'a*' and 'b*' layers.
cform = makecform('srgb2lab');
% Apply the colorform
lab_he = applycform(I,cform);

% Classify the colors in a*b* colorspace using K means clustering.
% Since the image has 3 colors create 3 clusters.
% Measure the distance using Euclidean Distance Metric.
```

```
ab = double(lab_he(:,:,2:3));
nrows = size(ab,1);
ncols = size(ab,2);
ab = reshape(ab,nrows*ncols,2);
nColors = 3;
[cluster_idx cluster_center] = kmeans(ab,nColors,'distance','sqEuclidean',
...
                                      'Replicates',3);
%[cluster_idx cluster_center] =
kmeans(ab,nColors,'distance','sqEuclidean','Replicates',3);
% Label every pixel in tha image using results from K means
pixel_labels = reshape(cluster_idx,nrows,ncols);
%figure,imshow(pixel_labels,[]), title('Image Labeled by Cluster Index');

% Create a blank cell array to store the results of clustering
segmented_images = cell(1,3);
% Create RGB label using pixel_labels
rgb_label = repmat(pixel_labels,[1,1,3]);

for k = 1:nColors
    colors = I;
    colors(rgb_label ~= k) = 0;
    segmented_images{k} = colors;
end



figure,subplot(2,3,2);imshow(I);title('Original Image');
subplot(2,3,4);imshow(segmented_images{1});title('Cluster 1');
subplot(2,3,5);imshow(segmented_images{2});title('Cluster 2');
subplot(2,3,6);imshow(segmented_images{3});title('Cluster 3');
% Feature Extraction
pause(2)
x = inputdlg('Enter the cluster no. to show in GUI:');
i = str2double(x);
% Extract the features from the segmented image
seg_img = segmented_images{i};
% Convert to grayscale if image is RGB
if ndims(seg_img) == 3
   img = rgb2gray(seg_img);
end
%figure, imshow(img); title('Gray Scale Image');

% Create the Gray Level Cooccurance Matrices (GLCMs)
glcms = graycomatrix(img);

% Derive Statistics from GLCM
stats = graycoprops(glcms,'Contrast Correlation Energy Homogeneity');
Contrast = stats.Contrast;
Correlation = stats.Correlation;
Energy = stats.Energy;
Homogeneity = stats.Homogeneity;
Mean = mean2(seg_img);
Standard_Deviation = std2(seg_img);
Entropy = entropy(seg_img);
RMS = mean2(rms(seg_img));
%Skewness = skewness(img)
Variance = mean2(var(double(seg_img)));
a = sum(double(seg_img(:)));
Smoothness = 1-(1/(1+a));
```

```
Kurtosis = kurtosis(double(seg_img(:)));
Skewness = skewness(double(seg_img(:)));
% Inverse Difference Movement
m = size(seg_img,1);
n = size(seg_img,2);
in_diff = 0;
for i = 1:m
    for j = 1:n
        temp = seg_img(i,j)./(1+(i-j).^2);
        in_diff = in_diff+temp;
    end
end
IDM = double(in_diff);

feat_disease = [Contrast,Correlation,Energy,Homogeneity, Mean,
Standard_Deviation, Entropy, RMS, Variance, Smoothness, Kurtosis, Skewness,
IDM];
disp(feat_disease);
%fn='F:\Train_data.xls';
%t=strcat('A',int2str(handles.q));
%xlswrite(fn,feat_disease,1,t);
%handles.q=handles.q+1;
I7 = imresize(seg_img,[300,400]);
axes(handles.axes3);
imshow(I7);title('Segmented Image');
%set(handles.edit3,'string',Affect);
% Update GUI
handles.ImgData3 = feat_disease;
guidata(hObject,handles);



function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
test = handles.ImgData3;
% Load All The Features
load('TrainData.mat')

% Put the test features into variable 'test'

result = multisvm(Veg_Feat,Veg_Label,test);
%disp(result);

% Visualize Results

if result == 0
    R1 = 'Alternaria Rot';
    set(handles.edit1,'string',R1);
elseif result == 1
    R2 = 'Aspergillus rot';
    set(handles.edit1,'string',R2);
elseif result == 2
    R3 = 'Botrytris';
    set(handles.edit1,'string',R3);
elseif result == 4
    R5 = 'Normal';
    set(handles.edit1,'string',R5);
end
```

```
% Update GUI
guidata(hObject,handles);



function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of edit1 as a
double


% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

multisem.m

```
function [itrfin] = multisvm( T,C,test )
%Inputs: T=Training Matrix, C=Group, test=Testing matrix
%Outputs: itrfin=Resultant class

itrind=size(test,1);
itrfin=[];
Cb=C;
Tb=T;
for tempind=1:itrind
    tst=test(tempind,:);
    C=Cb;
    T=Tb;
    u=unique(C);
    N=length(u);
    c4=[];
    c3=[];
    j=1;
    k=1;
    if(N>2)
        itr=1;
        classes=0;
        cond=max(C)-min(C);
        while((classes~=1)&&(itr<=length(u))&& size(C,2)>1 && cond>0)
        %This while loop is the multiclass SVM Trick
            c1=(C==u(itr));
            newClass=c1;
            %svmStruct = svmtrain(T,newClass,'kernel_function','rbf'); % I
am using rbf kernel function, you must change it also
```

```
            svmStruct = svmtrain(T,newClass);
            classes = svmclassify(svmStruct,tst);

            % This is the loop for Reduction of Training Set
            for i=1:size(newClass,2)
                if newClass(1,i)==0;
                    c3(k,:)=T(i,:);
                    k=k+1;
                end
            end
        T=c3;
        c3=[];
        k=1;

            % This is the loop for reduction of group
            for i=1:size(newClass,2)
                if newClass(1,i)==0;
                    c4(1,j)=C(1,i);
                    j=j+1;
                end
            end
        C=c4;
        c4=[];
        j=1;

        cond=max(C)-min(C); % Condition for avoiding group
                            %to contain similar type of values
                            %and the reduce them to process

            % This condition can select the particular value of iteration
            % base on classes
            if classes~=1
                itr=itr+1;
            end
        end
    end

valt=Cb==u(itr);        % This logic is used to allow classification
val=Cb(valt==1);        % of multiple rows testing matrix
val=unique(val);
itrfin(tempind,:)=val;
end

end
```

V. DATA SETS:

0.0858112	0.9601727	0.8188536	0.9915323	10.8629	40.70899	1.1786783	2.997166	1325.9276	0.9999999	19.664233	4.1216452	255
0.0958725	0.9779125	0.313652	0.9719482	39.217174	51.224734	4.6962769	11.015713	2217.8831	1	4.0925518	1.2780023	255
0.1669047	0.9852833	0.4610181	0.9742708	52.508812	82.957405	4.0253773	9.530421	6179.6521	1	2.9569355	1.2637462	255
0.0741523	0.9757856	0.6604509	0.9874401	20.495041	54.782972	2.1557661	4.8395016	2140.8954	0.9999999	10.906131	2.952966	255
0.403958	0.9153045	0.2723833	0.9327907	43.674906	53.980862	5.393523	12.080985	2685.611	1	5.7133013	1.6048094	255
0.2621218	0.9512823	0.6448252	0.9701366	23.080549	53.350457	2.1573588	6.6362563	2643.6622	1	7.0855377	2.2722447	255
0.0416812	0.9755723	0.7420253	0.9941137	12.431189	44.729573	1.4308906	3.8971481	1636.5969	0.9999999	17.978909	3.9683836	255
0.2065676	0.9763094	0.1936828	0.9459489	84.983384	72.765933	6.0923573	12.858797	4092.4793	1	1.8266968	0.1794333	255
0.1246192	0.9125523	0.7396627	0.9878948	10.445084	35.227324	1.6133393	4.3769852	1086.5364	0.9999999	19.308175	3.914738	255
0.1997788	0.9825142	0.2369443	0.9526584	124.78586	89.7443	6.0532671	13.048056	2296.6145	1	1.5769744	-0.3174754	255
0.1102563	0.9714835	0.6844433	0.9875931	21.720685	58.71312	1.928829	4.2338946	2271.6456	1	10.69008	2.9345662	255
0.2083692	0.9577582	0.8261017	0.9830479	15.834071	54.441552	1.1603699	2.972895	2286.9253	0.9999999	13.653104	3.4563478	255
0.0417947	0.9980253	0.3749511	0.9891345	145.86254	116.90472	3.2087018	12.508542	7083.9385	1	1.2223713	-0.3073665	255
0.01197	0.947756	0.8016777	0.9952164	5.8465486	20.915932	1.4964145	3.6072403	329.48686	0.9999998	20.39885	4.181105	255
0.0330217	0.9536572	0.711883	0.9866352	8.7666528	22.162132	1.8063234	4.1731777	323.4523	0.9999998	10.135467	2.7292557	255
0.2093016	0.9838022	0.3411204	0.9755866	76.168146	89.077999	4.9006666	10.75097	6098.9274	1	1.6039236	0.5430594	255
0.2609481	0.9621629	0.5444286	0.9799875	31.608508	63.373879	2.900512	6.6777732	3079.8088	1	5.2608088	1.8912027	255
0.0858438	0.9792702	0.6381558	0.9893945	22.404599	57.0868	2.2367095	5.0741766	1991.7132	1	8.6508824	2.6186299	255
0.1193327	0.9774366	0.5503221	0.9899578	27.756395	58.242068	2.8674511	6.8026568	2587.283	1	5.6954891	2.0142572	255
0.1352407	0.9486209	0.569191	0.9952619	22.502167	42.381746	3.4138052	9.0685374	1528.1768	1	17.784599	3.4867233	255
0.2572433	0.9836657	0.530898	0.9832476	58.736663	95.229136	3.207141	7.2704865	6408.5513	1	2.3231596	1.0987781	255
0.2109148	0.9509424	0.7506957	0.983409	17.57854	50.391746	1.8817847	4.3353752	2146.6763	0.9999999	11.077591	3.0075567	255
0.2367648	0.9744769	0.1944113	0.9560791	88.135328	74.414438	6.4154549	13.178615	4649.0264	1	2.2754914	0.352305	255
0.0444234	0.970084	0.85339	0.9951604	7.6393736	34.192862	0.9668941	2.6153835	993.95059	0.9999999	29.320076	5.0947017	255
0.1985176	0.959106	0.6171207	0.9739774	25.258246	58.912124	2.3905626	7.2731415	3225.6696	0.9999999	7.4353157	2.3812764	255
0.7577318	0.9343334	0.4157165	0.9404948	54.695203	81.799697	3.6572793	9.2088328	5941.2217	1	2.5360436	1.0753296	255
0.6476141	0.9478537	0.5102197	0.9466839	47.642042	84.981064	4.0561525	9.527942	6097.6749	1	3.8153936	1.5963664	255
0.2974553	0.9516418	0.6622979	0.9633619	25.570405	60.128465	2.1711323	6.3624538	3143.6463	1	6.520582	2.2227301	255
0.2685712	0.9822742	0.2787727	0.9521439	86.480179	94.683685	4.9076233	11.307436	7313.0333	1	1.6345811	0.486717	255
0.136799	0.9496322	0.531966	0.9717466	22.334894	52.194307	2.7286822	6.7307089	2107.805	1	7.3431539	2.3683662	255
0.3391165	0.9753245	0.5428844	0.9714056	51.660619	90.695314	3.1484767	7.7740689	6022.2582	0.9999999	3.0261199	1.3494027	255
0.7807229	0.953825	0.4119289	0.9517082	69.028283	97.486086	3.6199325	9.3146104	8139.1489	0.9999999	2.0881254	0.9012713	255
0.4637108	0.9074391	0.4789813	0.9616977	35.84104	68.807214	3.2743239	8.602012	4107.4997	0.9999999	5.6092244	1.9615154	255
0.2254173	0.9805771	0.3193514	0.9672193	67.399108	81.927205	4.7586622	10.646164	3723.3522	1	2.1397309	0.7565899	255
0.1623864	0.9594072	0.6314566	0.980541	23.097525	56.003848	2.2592837	5.1225299	1845.0465	0.9999999	8.8520579	2.6047686	255
0.3013284	0.8068366	0.5487309	0.9728873	17.706629	36.618755	2.9105767	7.2193219	1062.8661	0.9999999	10.11197	2.5643563	255
0.1876155	0.9526835	0.6591637	0.9767434	21.988098	53.473307	2.2889624	6.0942857	2390.9842	1	8.6714534	2.5562998	255
0.0933638	0.9572731	0.6284277	0.9823416	17.624039	45.275419	2.3301461	5.146107	1432.7542	0.9999999	9.9742019	2.7738901	255
0.1274653	0.9880962	0.2759476	0.9794648	78.261918	80.285226	5.1849594	11.524374	4705.8162	1	1.5365123	0.3238519	255
0.1517606	0.9596483	0.4304801	0.9832224	31.175327	57.54752	3.5936987	7.758375	2313.9038	1	5.7129341	1.915074	255
0.1714919	0.985952	0.3254093	0.984854	77.104555	90.120367	4.4543215	10.337092	4344.588	1	1.6262841	0.5512961	255
0.2877869	0.9454199	0.7170329	0.9734635	20.440246	54.04067	1.7070259	4.8501694	2339.5858	0.9999999	8.734465	2.6169515	255
0.1326042	0.95304	0.6000207	0.974651	21.298827	55.372237	2.4740604	4.9160765	1911.3876	0.9999999	10.327887	2.8738101	255
0.4983302	0.9378797	0.3027563	0.9333731	49.090686	66.722401	4.6029963	10.544886	3617.8852	1	3.0601705	1.1304603	255
0.2752618	0.9734963	0.5051564	0.9756601	49.041913	83.205236	3.2213131	7.9758779	5185.3124	1	3.2390967	1.3588662	255
0.7804571	0.7121103	0.5935468	0.9245333	19.13327	43.030784	2.7945385	6.9725625	1639.6265	0.9999997	10.149338	2.6799015	255
0.8580381	0.9254303	0.2610406	0.8773987	72.154063	81.655268	5.5559351	11.789998	5083.919	0.9999999	1.8414494	0.6316472	255
0.7728381	0.7114549	0.5954615	0.9251725	18.989553	42.856113	2.7792569	6.9469337	1626.0741	0.9999997	10.207146	2.6890415	255
0.3206426	0.9457398	0.6496875	0.976014	27.78837	63.396373	2.1168229	6.3988733	3501.7185	1	6.7649735	2.2227759	255
0.1683143	0.9584171	0.3981296	0.980097	41.512858	72.298932	3.8643853	9.0790759	3512.512	1	4.4997927	1.7280341	255
0.3206426	0.9457398	0.6496875	0.976014	27.78837	63.396373	2.1168229	6.3988733	3501.7185	1	6.7649735	2.2227759	255
0.6449321	0.9428238	0.4698141	0.9408716	49.053016	81.81678	3.6843678	9.0646279	5697.7307	0.9999999	3.304452	1.3762382	255
0.2235371	0.9484541	0.5176133	0.9806659	27.933329	62.585198	3.0131788	6.5301539	2695.1568	0.9999998	6.9079378	2.2771575	255
0.5251567	0.9248838	0.4249958	0.9415904	37.617617	61.678159	3.6427325	8.5548588	2930.0342	0.9999999	3.7578353	1.4440754	255
0.3634127	0.8918357	0.6224472	0.9755615	20.622117	50.740421	2.3457803	6.3458052	2278.7097	0.9999997	9.7362058	2.7088658	255
0.1222619	0.9381255	0.6475753	0.987413	17.232033	51.243861	2.0208739	5.4370036	2147.0636	0.9999996	12.693475	3.2737206	255

VI. EXPERIMENTAL RESULTS or SIMULATION RESULTS:

DATASET IMAGE:

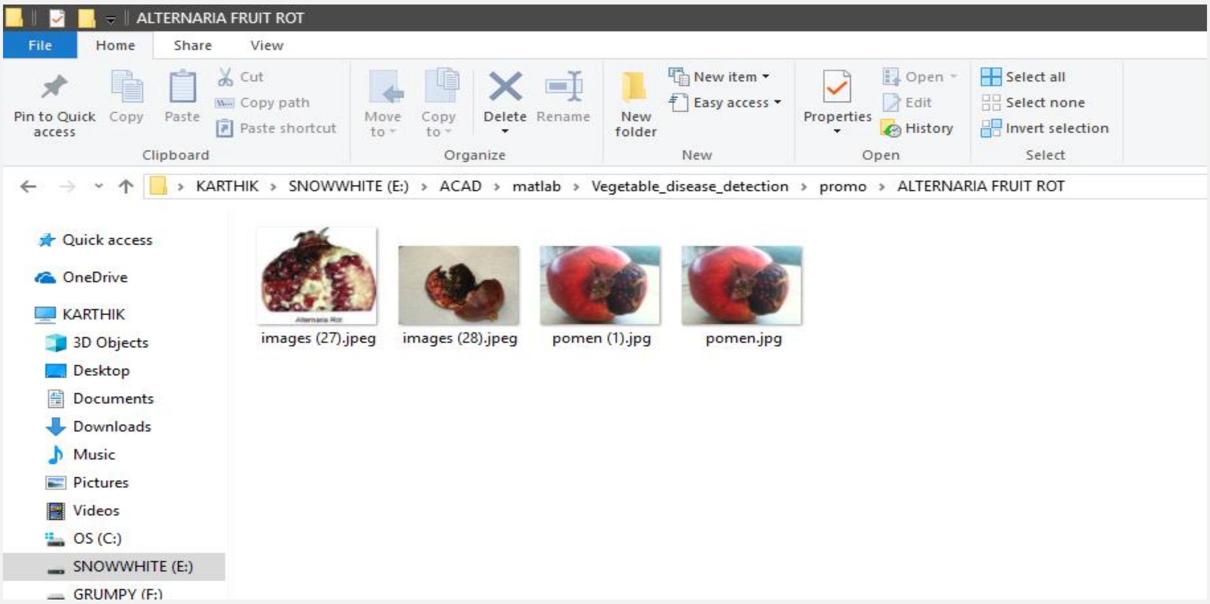


Fig 1: dataset image

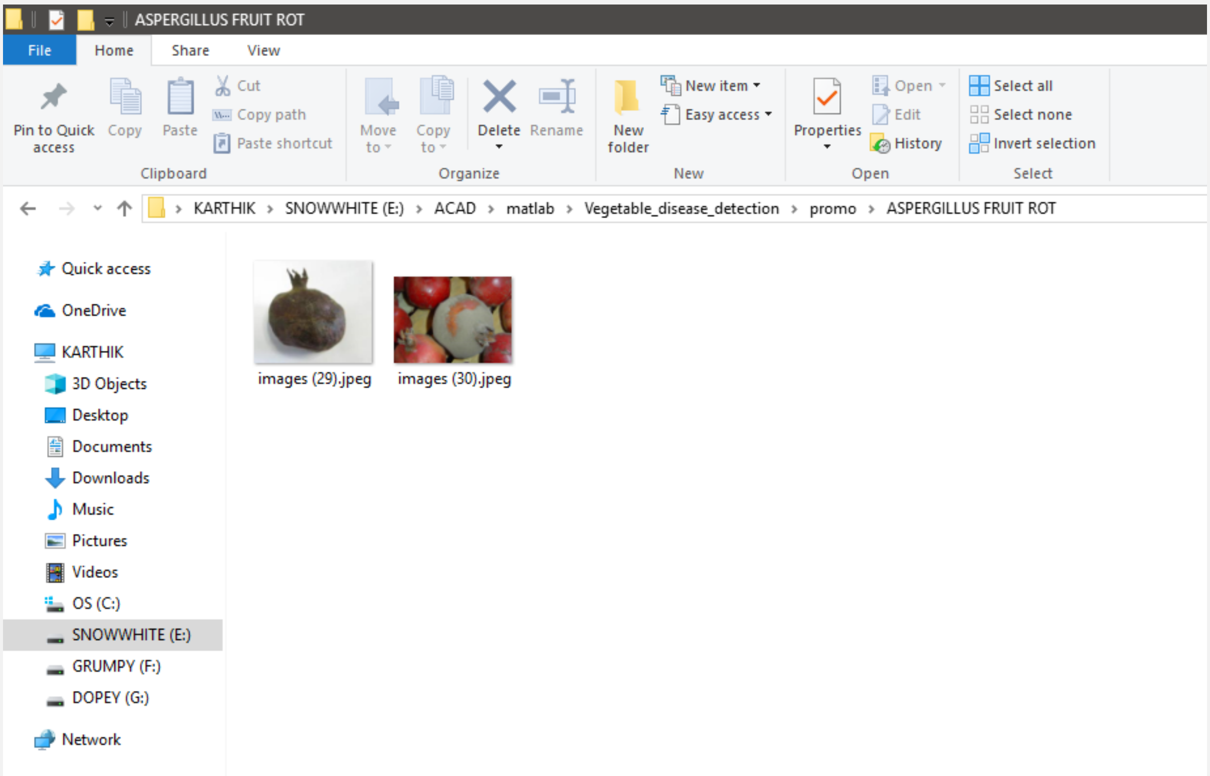


Fig2. Dataset image

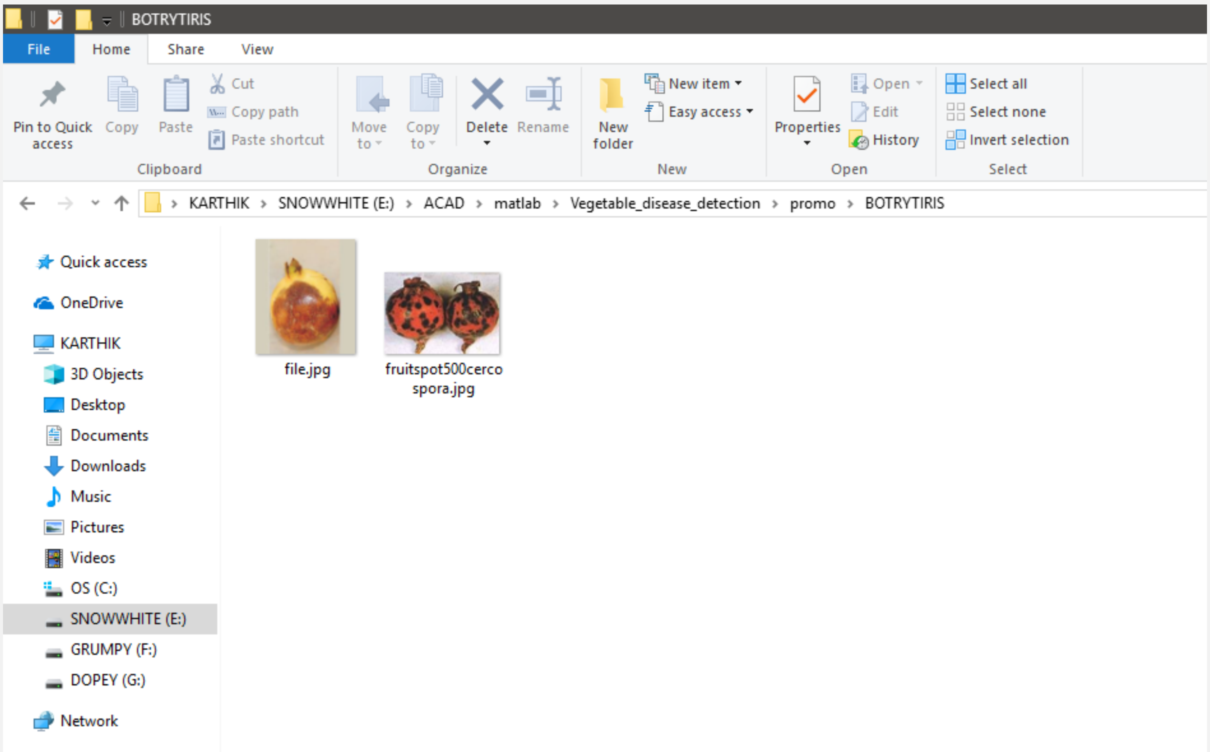


Fig3. Dataset image

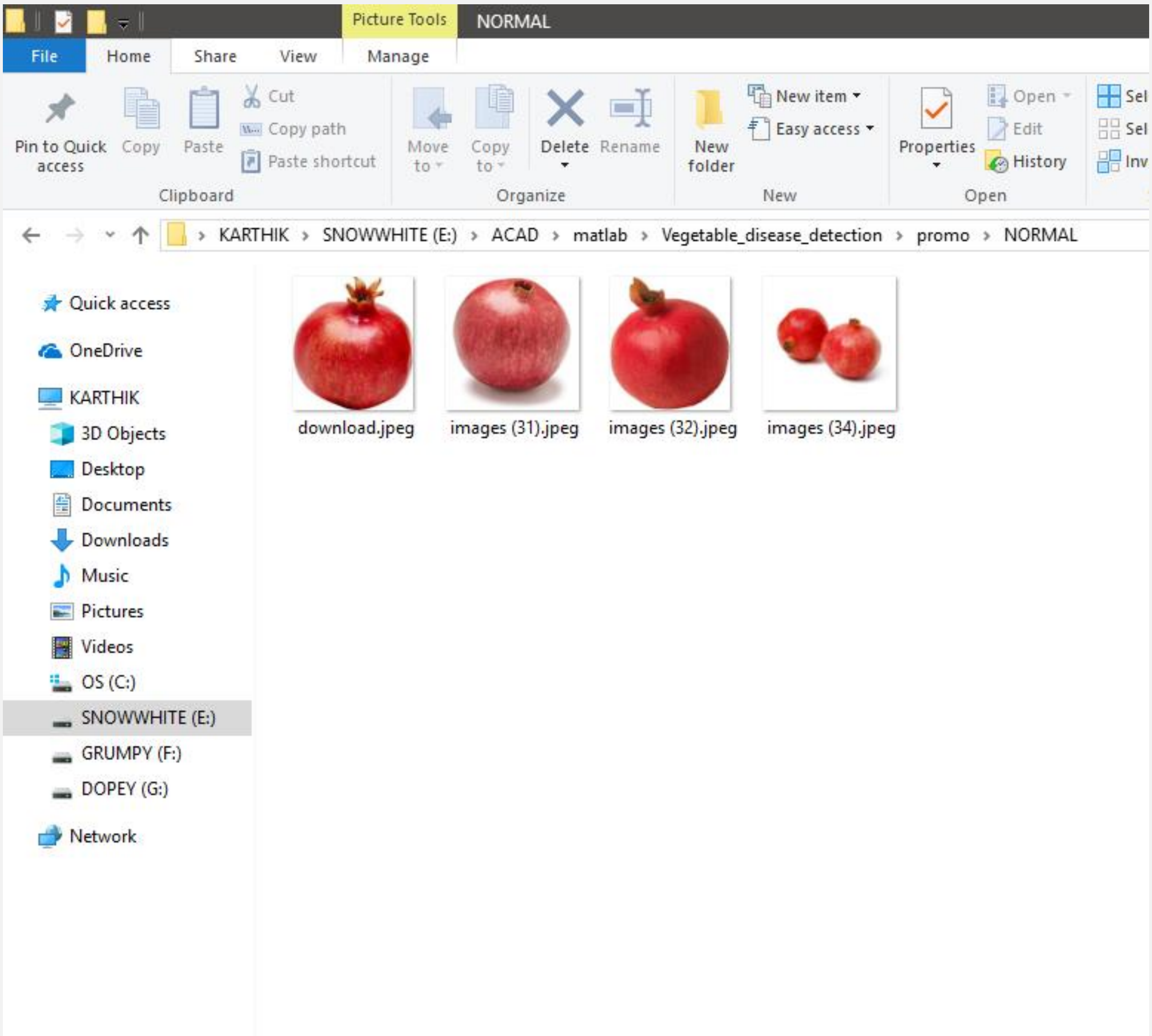


Fig4. Dataset image

OUTPUT IMAGE:

Fig.1

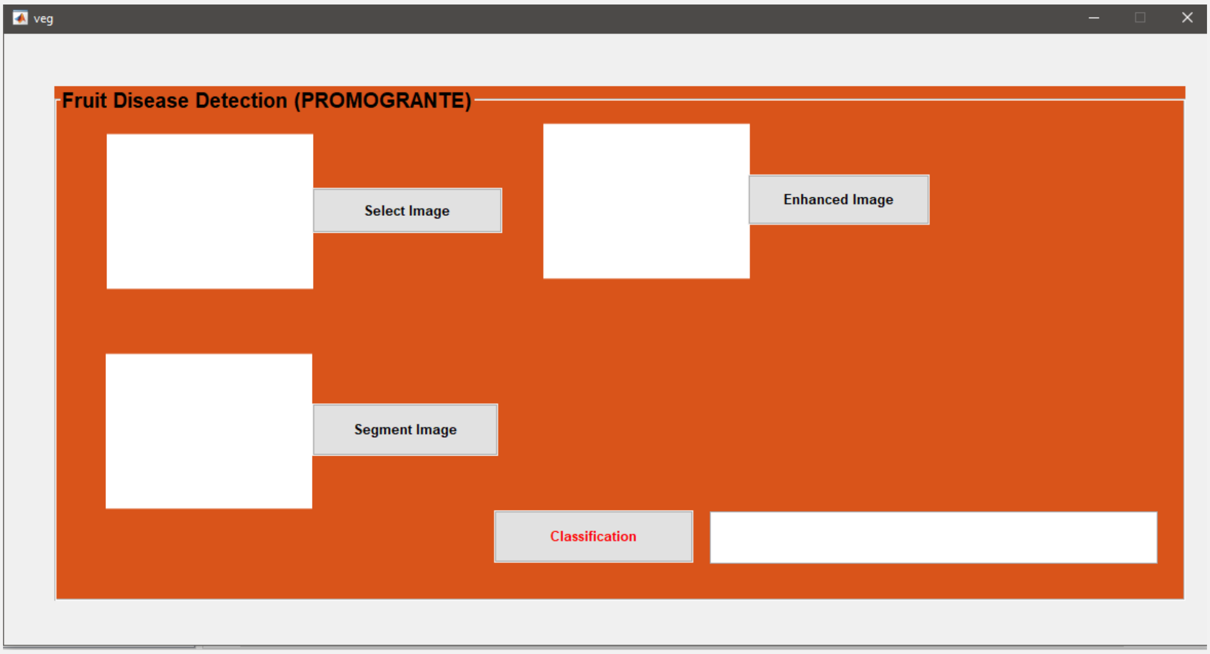


Fig.2 **Test image is selected for disease detection**

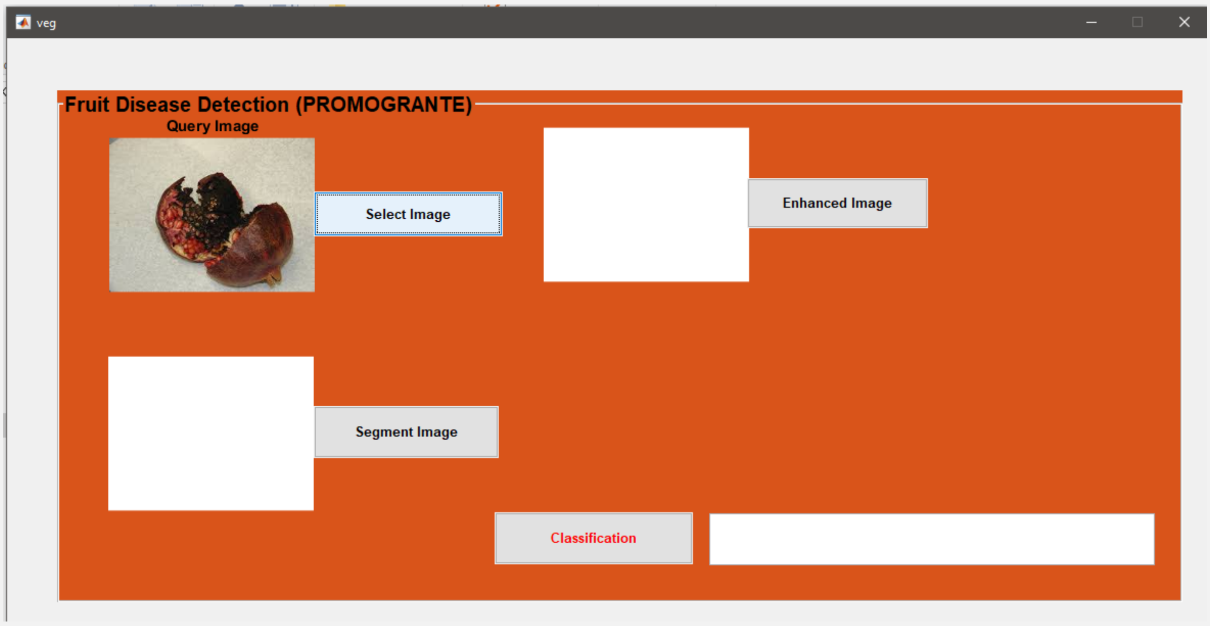


Fig.3 **Enhanced image**

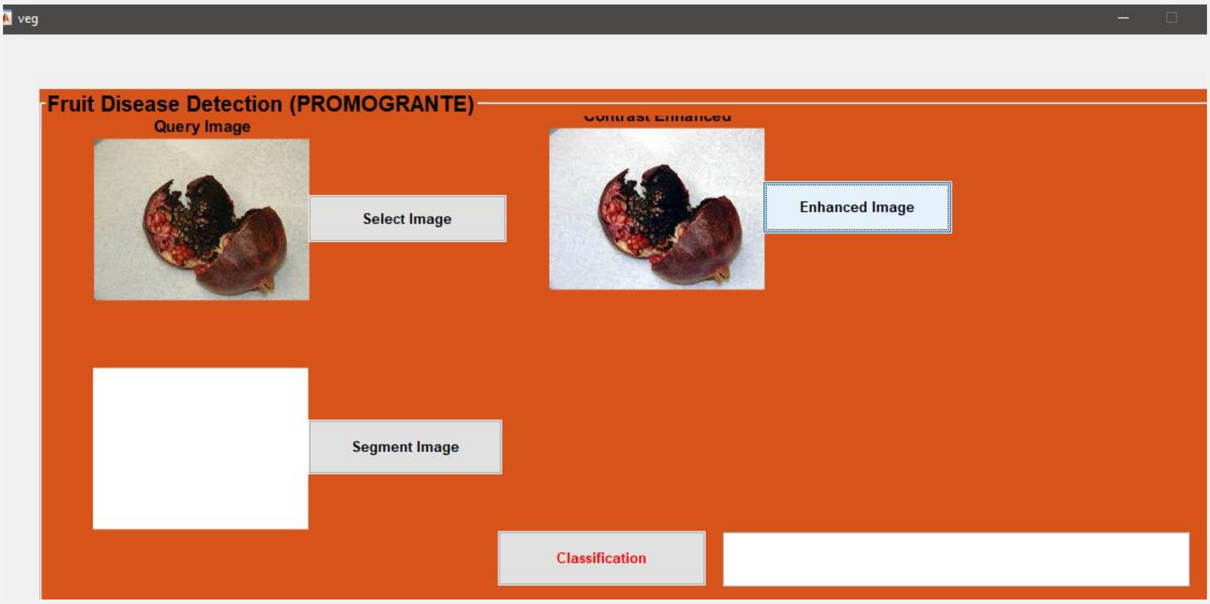
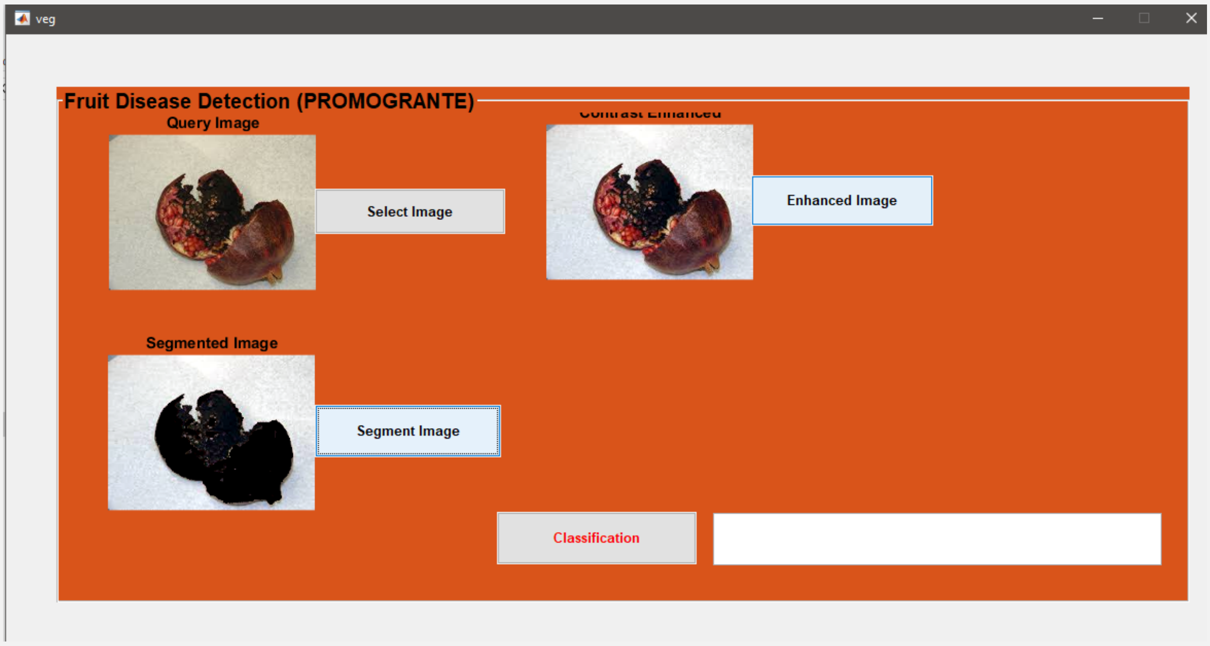


Fig.4 **Selecting cluster**



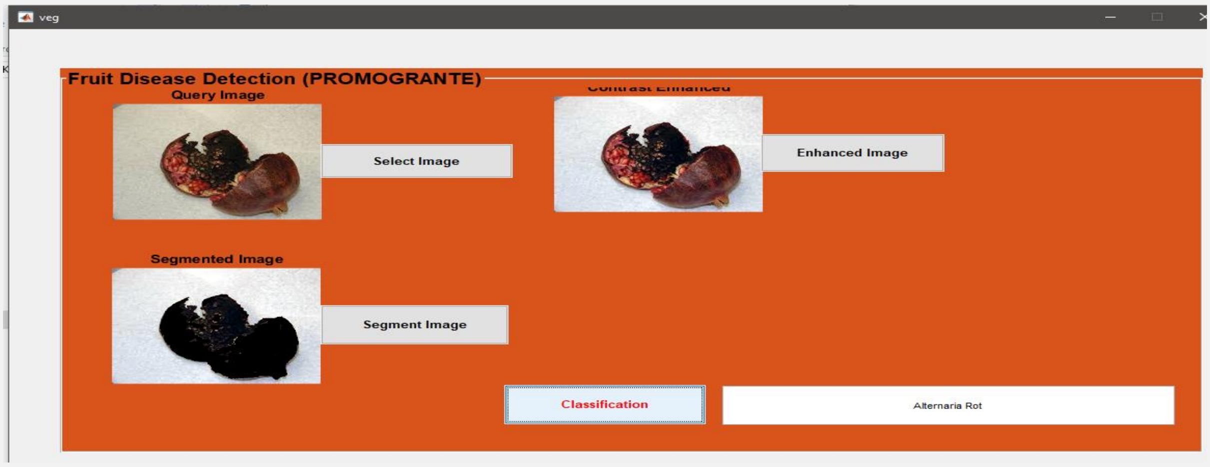


fig.5 **Result of selected image**

COMPARATIVE STUDY OF PROPOSED SYSTEM WITH RELATED EXISTING SYSTEM (includes tabulated results or graphs)

Paper	**Techniques used**	**Accuracy and result**
Classification of Anthracnose Fungal Disease of fruit based on Statistical Texture Features	Thresholding algorithm is used to segmentation, run length matrix is used to extract feature, ANN is used to classification.	The average accuracy of 84.56% for normal type and 76.67% for anthracnose affected type.
fruit Disease Detection using Color, Texture Analysis and ANN	K-means clustering, ANN	Accuracy of this technique is 85% to detect the fruit disease.
Infected fruit Part Detection Using Clustering	Gaussian low pass filter to remove the noise, K – means clustering	The algorithms are able to segment the defect 93% accuracy. The major drawback of k-means is that, there may be a skewed clustering result if the cluster number estimate is incorrect.
Smart Farming: tomato Disease Detection Using Image Processing	K-means clustering algorithm, ANN is used for classification. The test image is obtained from the user.	The overall accuracy of this method is 82%
Grading and Classification of Anthracnose Fungal Disease of Fruits or vegetable based on Statistical Texture Features	Thresholding algorithm is used to segmentation, runlength matrix is used to extract feature, ANN is used to classification.	The average accuracy of 84.65% for normal type and 76.6% for anthracnose affected type.
Fruit Disease Detection using Color, Texture Analysis and ANN	K-means clustering, SURF algorithm, ANN	Accuracy of this technique is 85% to detect the fruit disease.
Infected Fruit Part Detection Using Clustering	Gaussian low pass filter to remove the noise, K – means clustering	The algorithms are able to segment the defect 93% accuracy. The major drawback of k-means is that, there may be a skewed clustering result if the cluster number estimate is incorrect.
Grading and Classification of Anthracnose Fungal Disease of Fruits or vegetable based on Statistical Texture Features	K-means clustering algorithm, ANN is used for classification. The test image is obtained from the user.	The overall accuracy of this method is 82%
Fruit Disease Detection using Color, Texture Analysis and ANN Infected Fruit Part Detection Using Clustering	K – means clustering, CLBP (Complete Local Binary Patter) used to extract feature, Multiclass SVM for classifier.	The CLBP feature shows more accurate result for the identification of apple fruit diseases and achieved more than 93% classification accuracy.

VI. CONCLUSION:-

This paper provides efficient and accurate plant disease detection and classification technique by using MATLAB image processing. The proposed methodology in this paper depends on K-means and Multi SVM techniques which are configured for both leaf & fruit disease detection. The MATLAB software is ideal for digital image processing. K-means clustering and SVM algorithm provides high accuracy and consumes very less time for entire processing. In future work, we will extend our database for more plant disease identification.

VII. FUTURE WORK:-

The accurate detection and classification of the plant disease is very important for the successful cultivation of the crops, this can be done using digital image processing. In this project, the detection as wells the remedy for curing it is achieved. This project utilizes GSM so as to send the message to every kind of mobile handset. This project utilizes various image processing techniques which provide accurate results.

VIII. REFERENCES:-

[1] Di Cui , Qin Zhang , Minzan Li , Youfu Zhao ,Glen L. Hartman, “Detection of soybean rust using a multispectral Image sensor”, Springer Science+ Business Media, LLC 2009.

[2] Brendan A. Niemira, William W. Kirk, and Jeffrey M. Stein, “Screenng For Late Blight Susceptibiliy In Potato Tubers By Digital Analysis Of Cut Tuber Surfaces”, Department of Botany and Plant Pathology, Michigan State University East Lansing48824.

[3] Yan Li Chanlei Xia, Jangmyung Lee, “Vision Based Pest Detection and Automatic Spray of Greenhouse Plant”, IEEE International Symposium on Industrial Electronics (ISIE 2009) Seoul Olympic Parktel, Seoul, Korea July 5-8, 2009.

[4] Amina Khatra and Maninder Kaur, “A Novel Machine Vision System for Radish Crop Quality Monitoring based on Leaf Inspection”, IOSR Journal of Engineering (IOSRJEN) Vol. 2 Issue 2, Feb.2012, pp. 372-375.

[5] Nitya Subramanium, Gule Saman, Edwin Hancock, “DETECTION OF SKIN LESIONS USING DIFFUSE POLARIZATION”, Proceedings of 2010 IEEE 17th International Conference on Image Processing, September 26-29, 2010, Hong Kong.

[6] Kuo-Yi Huang, “Application of artificial neural network for detecting Phalaenopsis seedling diseases using color and texture features”,http://www.sciencedirect.com, Computers and Electronics in Agriculture 57 (2007) 3–11.

[7] Kaiyi Wang,Shuifa Zhang , Zhibin Wang, Zhong Qiang Liu and Feng Yang, “Mobile smart device-based fruit disease and insect pest recognition method”, National Engineering Research Center for Information Technology in Agriculture , Beijing , 100097 , China 26 Jul 2013.

[8] K.Dang,H.Sun,J.P.Chanet,J.Garcia-Vidal,J.M.Barcelo-Ordinas,H.L.Shi and K.M.Hou , “ Wireless Multimedia Sensor Network for Plant Disease Detections”, NICST'2103 New Information Communication Science and Technology for Sustainable Development: France-China International Workshop, 18-20 Septembre, Clermont-Ferrand, France.

[9] Jose J.Lopez, Maximo Cobos, Emanuel Aguilera,”Computer-based detection and classification of flaws in citrus fruits or vegetable”, Neural Comput & Applic (2011) 20:975–981, DOI 10.1007/s00521-010-0396-2, Springer-Verlag London Limited 2010.

[10] Jayme Garcia Arnal Barbedo, “Digital image processing techniques for detecting, quantifying and classifying plant diseases",BarbedoSpringerPlus2013,2:660,http://www.springerplus.co m/content/2/1/660. [11]

Guanlin Li, Zhanhang Ma, Haiguang Wang, "Development of a Single-Leaf Disease Severity Automatic Grading System Based on Image Processing", Proceedings of the 2012 International Conference on Information Technology and Software Engineering, Springer-Verlag Berlin Heidelberg 2013. [12] H.Kalkan,P. Berait,Y. Yardimci,T.C.Pearson "Detection of contaminated hazelnuts and ground red chilli pepper flakes by multispectral imaging", Computers and Electronics in Agriculture 77 (2011) 28–34, www.elsevier.com/locate/compag.

[13] Zou Xiao-bo,Zhao jie-wen,Li yanxiao,Mel Holmes "In-line detection of apple defects using three color cameras system Logic", Computers and Electronics in Agriculture 70 (2010) 129–134 , www.elsevier.com/locate/compag.

[14] Zulkifli Bin Husin, Abdul Hallis Bin Abdul Aziz, Ali Yeon Bin Md Shakaff, Rohani Binti S Mohamed Farook, "Feasibility Study on Plant Chili Disease Detection Using Image Processing techniques", 2012 Third International Conference on Intelligent Systems Modelling and Simulation, 978-0-7695-4668-1/12 $26.00 © 2012 IEEE.

[15] A.Meunkaewjinda,P.Kumsawat,K.Attakitmongcol,A.Srikaew "Grape leaf diseases detection from color imagery using hybrid intelligent system", Proceedings of ECTI-CON 2008 978-1-4244-2101-5/08/$25.00 c2008 IEEE.

[16] Arun Kumar R, Vijay S Rajpurohit, "Research Avenues in Fruit Characterization using Machine Vision: A Review", International Journal of Emerging Technology and Advanced Engineering, www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 12, December 2014.

[17] Wang Zhi-jun, Liu Xin, Jiang Meng and Cheng Shu-han "The Application of Image Retrieval technology in the Prevention of Diseases and Pests in Fruit Trees", D. Li and Y. Chen (Eds.): CCTA 2013, Part I, IFIP AICT 419, pp. 160–167, 2014. IFIP International Federation for Information Processing 2014.

[18] Hongfei Lu,Hong Zheng,Ya Hu, Heqiang Lou, Xuecheng Kong , "Bruise detection on red bayberry (Myrica rubra Sieb. & Zucc.) Using fractal analysis and support vector machine", Journal of Food Engineering 104 (2011) 149–153 www.elsevier.com/locate/jfoodeng.

[19] Bhavini J. Samajpati and Sheshang D. Degadwala "Hybrid Approach for Apple Fruit Diseases Detection and Classification Using Random Forest Classifier" International Conference on Communication and Signal Processing, April 6-8, 2016, India.

[20] Jun Lu, Pengfei Wu, Jiwei Xue "Detecting Defects on Citrus Surface Based on Circularity Threshold Segmentation" 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD).

[21] Monika Jhuria, Rushikesh Borse "Image Processing for Smart Farming: Detection of Disease and fruit grading" 2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013).

[22] Mrunmayee Dhakate and Ingole A. B. "Diagnosis of Pomegranate Plant Diseases using Neural Network" IEEE.

[23] Ashwini Awate, Damini Deshmankar, Gayatri Amrutkar, Utkarsha Bagul, Prof. Samadhan Sonavane "Fruit Disease Detection using Color, Texture Analysis and ANN" 2015 International Conference on Green Computing and Internet of Things (ICGCIoT).

[24] Sudhir Rao Rupanagudi, Ranjani B.S., Prathik, Nagaraj ,Varsha G. Bhat "A Cost Effective Tomato Maturity Grading System using Image Processing for Farmers" International Conference on Contemporary Computing and Information,2014.

[25] Pradnya Ravindra Narvekar, Mahesh Manik Kumbhar2, S. N. Patil "Grape Leaf Diseases Detection & Analysis using SGDM Matrix Method" International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3237:2007 certified organization) Vol.2,Issue 3, March 2014.