SEED GRADING SYSTEM

A PROJECT REPORT

Submitted by

SUBASHREE D (311117205057)

RAKSHINI J (311117205046)

CURIE CHRISTAMIN R (311117205021)

In partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

LOYOLA-ICAM

COLLEGE OF ENGINEERING AND

TECHNOLOGY CHENNAI-600034

ANNA UNIVERSITY: CHENNAI - 600025

APRIL 2021

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project **“SEED GRADING SYSTEM”** is the bonafide work of **SUBASHREE D (311117205057), RAKSHINI J (311117205046), CURIE CHRISTAMIN R(311117205021)** who carried out the project work under my supervision.

SIGNATURE

Dr. R JULIANA, M.E, Ph.D

HEAD OF THE DEPARTMENT
Professor
Information Technology
Loyola-ICAM College of
Engineering and Technology
Loyola campus,
Nungambakkam, Chennai-34

SIGNATURE

Ms. ANITHA E, M.E

SUPERVISOR
Assistant Professor
Information Technology
Loyola-ICAM College of
Engineering and Technology
Loyola campus,
Nungambakkam, Chennai-34

Submitted for the Project Viva voce examination held on ______________ .

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First of all, I am grateful to God, for granting me this opportunity and the capability to proceed successfully. Working on this project has been a rewarding experience.

I would like to express my sincere thanks and gratitude to our Director, **Rev. Dr Maria Wenish SJ**, the Dean of Students **Dr Caleb Chanthi Raj**, and the Dean of Engineering Education, **Mr. Nicholas Juhel** for their constant support and encouragement. I thank them for making the right resources available at the right time.

I would like to extend my gratitude to our Principal, **Dr L Antony Michael Raj, M.E., Ph.D.,** for his motivating, constructive criticism and valuable guidance during the course of the project. I would like to express my sincere thanks to the Head of the Department **Dr R Juliana, M.E., Ph.D.**, for her comments and suggestions.

I also would like to thank the project coordinator Dr. Janani, the faculty and technical staff of our department for their critical advice and guidance which was helpful in completion of the project.

I also extend my sincere gratitude to my project guide Ms E Anitha , for providing valuable insights and resources leading to the successful completion of our project. I would also like to thank the faculty members of our department for their critical advice and guidance which was indispensable for our work.

Last but not the least; I place a deep sense of gratitude to my friends and family who have been a constant source of inspiration during the preparation of this project.

ABSTRACT

The quality of seeds are directly proportional to the quality and quantity of the crops yielded. That is, the use of good quality seed plays an important role in the satisfactory production of a good quality crop, which is in turn an essential requirement for exporting them to markets and consumers. Thus, this grading of grains forms as one of the major steps in quality analysis of grains. This initial and crucial step is mostly ignored in most cases or is done using large, loud and expensive electronic machinery. The main aim of this paper is to design and implement a system that mainly uses image processing technique to detect and classify the healthy seeds that are suitable for cultivation. Thereby, eliminating the need for such machines at an affordable cost that can also be set up easily. As seeds are the main part of any cultivation, healthy seeds will yield healthy crops, it becomes necessary to provide the farmers with healthy seeds. Digital image processing is being used for quality evaluation and grading of agricultural food in industry. Here we are going to extend the complexity and bring it to macro level and implement it to seed. The system is based on Region growing algorithm which is an approach to image segmentation which examines neighboring pixels of initial seed points and determines whether the pixel neighbors should be added to the region and classifies the seeds based on the values of the parameters like area, diameter, perimeter etc. Thus providing a methodological solution to detect and classify the healthy seeds.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. Introduction to Domain

India is an agricultural based country. The demand for the quality of food products we consume is increasing day by day. As the literacy rate is increasing in India, people are becoming more aware of what they consume, creating an increasing need for the quality of food products. This puts a high pressure on the farmers to grow better quality crops. They are finding it difficult and are facing a lot of problems to reach demands mainly because the seeds exported to them are not upto the quality. Now that we know how highly the quality of grains is important for today's market, We also need to know that some traders adulterate it with poor quality products to gain illicit profits. This malpractice has motivated production of low-grade quality grains. Adulteration of grains consist of damaged seeds, breaking, cracked ones, etc. To resolve this issue, quality testing of food grains play an important role. But to test the quality of seeds manually is a very tedious task. Therefore providing farmers with an automated system will provide us with better results and successful cultivation. This can be achieved by using image processing techniques. In some developing countries cultivation of best quality crops is becoming competitive and crop monitoring plays an important role in it. To solve these problems we are using the Region Grow algorithm to digitally process the captured image of seeds for testing and grading .

1.2 Purpose of the Project

As the technology is growing wider people are adopting new technologies. Abandoning the old manual and tedious work techniques. This project aims to automate the system for grading of seeds by extracting their morphological features from their images, as the main attribute for classifying them. This classification is done using image processing techniques using MATLAB C and Region Growing algorithm. This method requires minimum time as well as low in cost at the same time, providing much more efficient results.

1.3 Project Plan

The project completion is planned to be done in a 12 week span, with the time allocation for analyzing, integrating, development and testing the system for 8 weeks (Development from 3rd week and Testing from 7th week). The Documentation works are planned to be progressed from 2nd week onwards to record simultaneously the work being carried out.

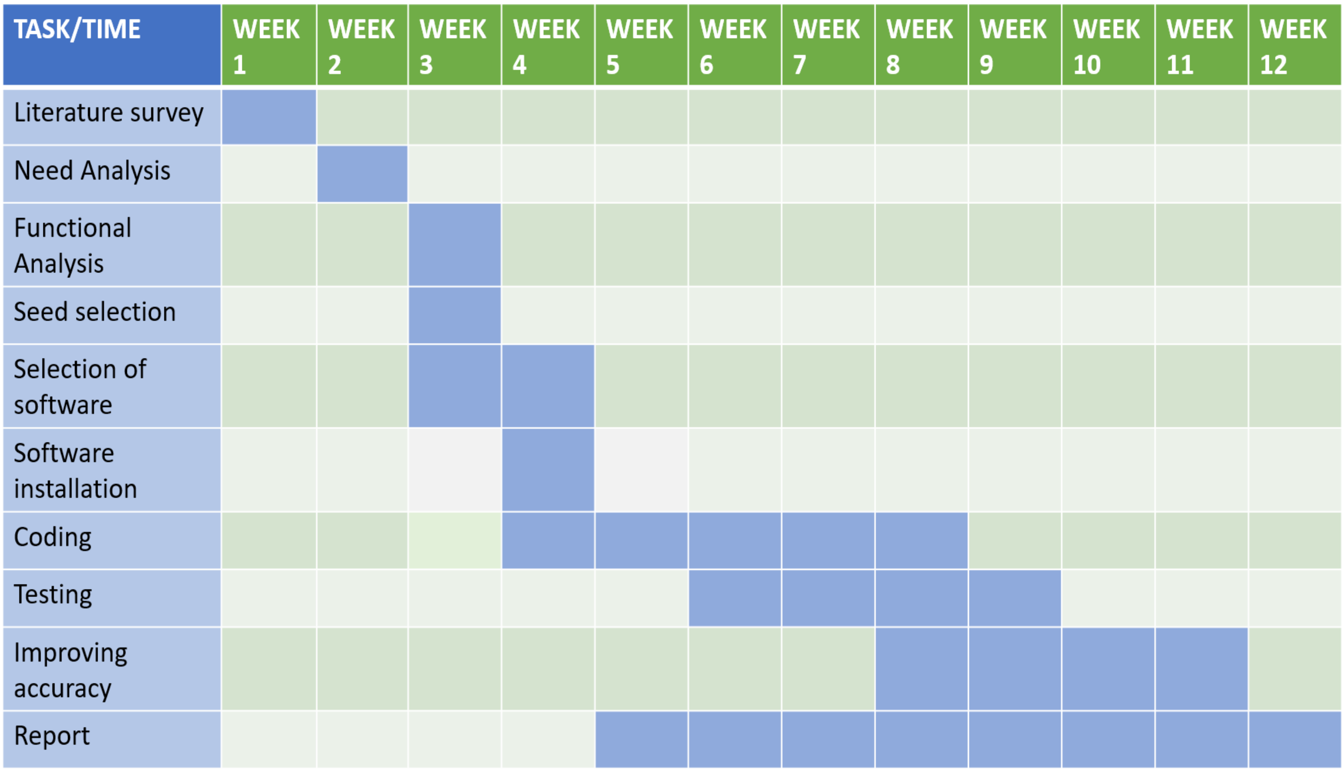


Fig 1.1 Project plan

1.4 Scope of the Project

The scope of this project is at a global level as food forms as the most important requirement for survival of any species and providing improvements in that field will never go out of trend. Testing and qualifying the seeds is one of the major parts of agriculture. If the best quality seeds are provided then the usage of fertilizers can be reduced to some extent because the diseases are sometimes seed borne diseases.This application would automate seed grading equipment and provide us with good quality seeds for cultivation. Image processing technique we use will help in detecting the quality of seeds based on the various parameters like Area, Perimeter, Centroid etc. which would result in best yield as the basic requirement is being satisfied undoubtedly, that is being provided with the best seeds .

1.5 Summary

This application aims to automate the seed grading equipment and manual grading technique used stereotypically by using MATLABin combination with Region Grow image segmentation method and Edge Masking to identify the good quality seeds from the damaged ones which would increase the crop yield rate.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

The seed testing method is more than 100 years old. It was implemented in the parts of Australia and Africa .But only simple and easiest methods were used. Later on seed testing laboratories were developed all over the world .

2.2 Surveyed Papers

- In this Literature survey a number of authors proposed their work and is what follows. Kaur and Verma, [1] have proposed computer vision techniques for grading of rice based on their sizes. They have done grading on the basis of medium ,long and short length rice. The images are acquired using a digital camera having high pixel resolution. The camera is located at a position normal to the object.The grading formula & standards were acquired from the analysis procedure for grading rice followed in India.
- Salome Hema Chitra,S.Suguna and S.Naganandini Sujatha[2] have proposed A Survey on Image Analysis Techniques in Agricultural Product In this work, they implemented a five processing module for seed identification. Seed sample images been taken for the basis of image acquisition and then a seed image is pre-processed by removal of noise and image enhancement technique. edge detection is used for Enhanced image and segmentation was done. From the segmented image extract the features like colour, shape and texture for normal and defected seed which may help to identify the seed by image analysis techniques.
- Ajay [2] have proposed a quality evaluation of rice grains using morphological methods. Physical features play a vital role in

classification and grading of rice grains. Features like length of seed, width, and perimeter are specially considered.

2.3 Summary

From the literature survey it is evident that major work is done on identification of different types of seeds but very less amount of work has been reported on the grading of seeds. Hence it is important that an automated process is essential to replace manual process of grading with "Digital image processing technique”. Here we have used region growing algorithms inside MATLAB to grade the seeds.

CHAPTER 3
SYSTEM ANALYSIS AND DESIGN

3.1 Problem Definition

To sort out the best quality seeds from a group of seeds, as sometimes there are problems that occur, such as the damaged, cracked or broken seeds are sold to the farmers along with other good ones, which thereby results in low yield and loss of both time and effort for the farmers. It is a very tedious task to sort out the damaged seeds manually hence it is mostly ignored and at few places they are using grading equipment which are very expensive. To overcome this problem we move towards the Digital Image Processing techniques for finding out the best quality of seeds.

3.2 Need Analysis

This application aims at reducing the impact of the nature of seeds on crop yield. It bridges the gap between farmers and seed retailers. By finding the best one's simply by capturing their images and receiving back the same image with seeds being labeled as damaged or not. Our motive is to simplify the process of grading and segregating the seeds of whatever the type may be, and this would also help people without prior agricultural knowledge by reducing the time and amount spent for finding the best ones.

3.2.1 Bull Diagram

The Bull Diagram describes the purpose of the System and about all the Actors who are involved in the system. As many vital actors are dependent upon this system.

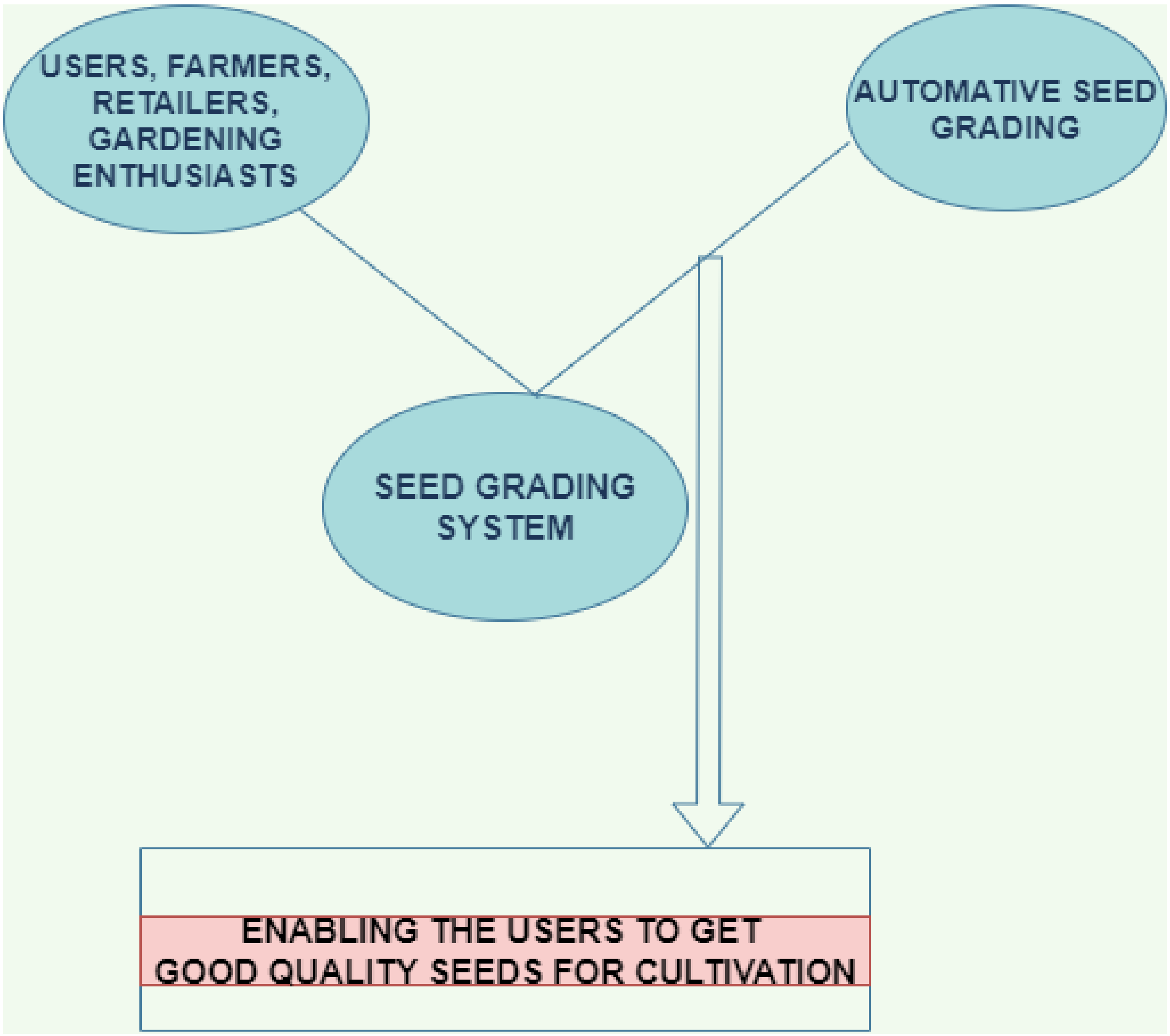


Fig 3.1 Bull Diagram

3.3 Functional Analysis

As farmers mostly don't find time to pursue top quality seeds, we have designed this system to reduce the time and cost spent on grading the seeds. People may buy seed packets as a whole and out of interest they may wish that they could also yield successful cultivation. But may not find sufficient time for figuring the damaged ones, so at times like that; our system would help its users with providing the facilities like, labeling the damaged ones without a lag and also with utmost accuracy.

3.3.1 Octopus Diagram

The Octopus Diagram describes the functionalities involved in a system. Those functions are classified into two, they are Principle functions and Constraint functions. The principle functions are those, which specify the primary functionalities that must be provided by the system. The constraint functions are those, which on satisfying the users will provide additional credits to the system or product developed.

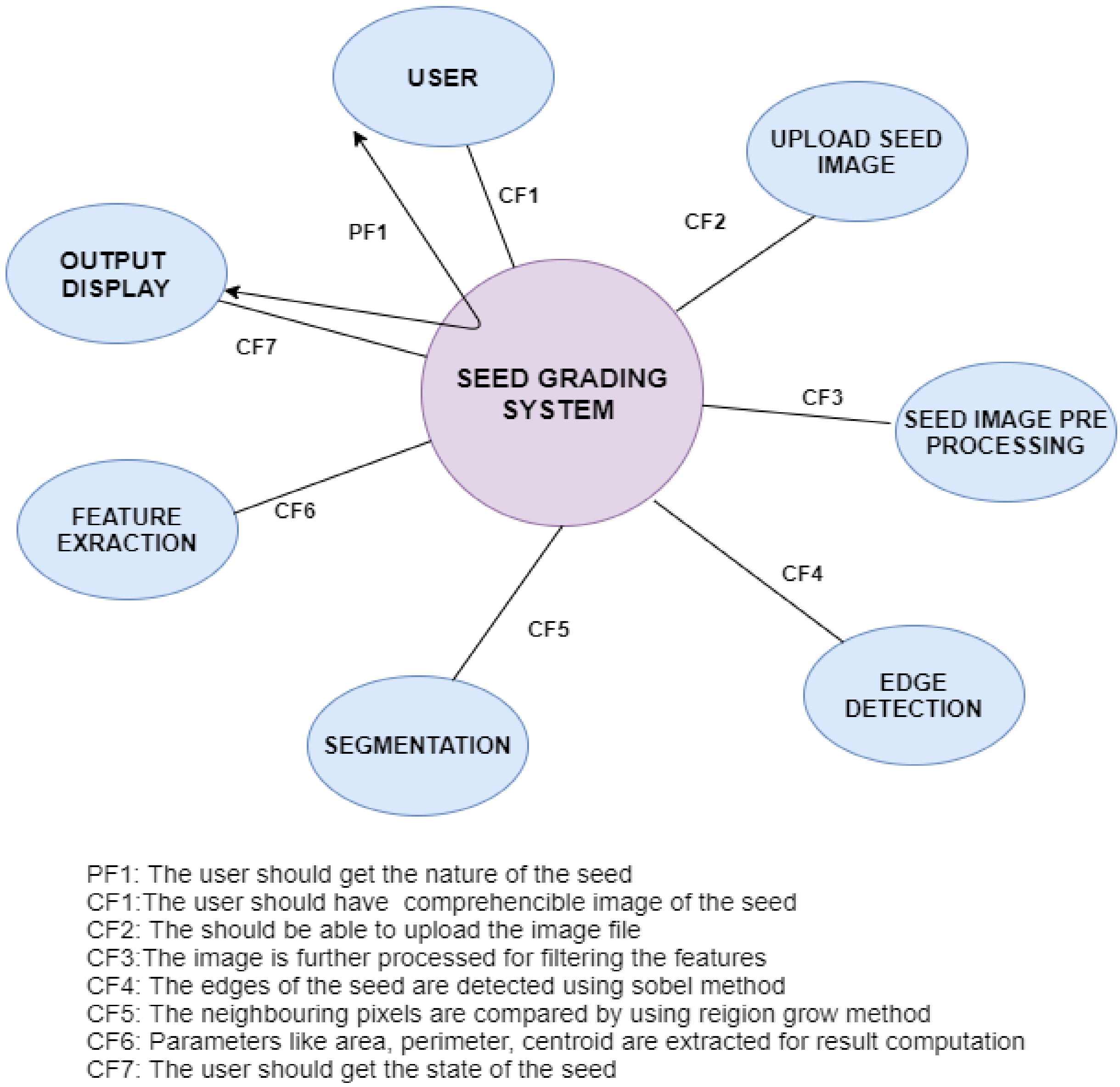


Fig 3.2 Octopus Diagram

3.4 Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. In our face mask detection and alert system the process flow is represented in level 0, level 1, level 2 diagrams.

3.4.1 DFD-0

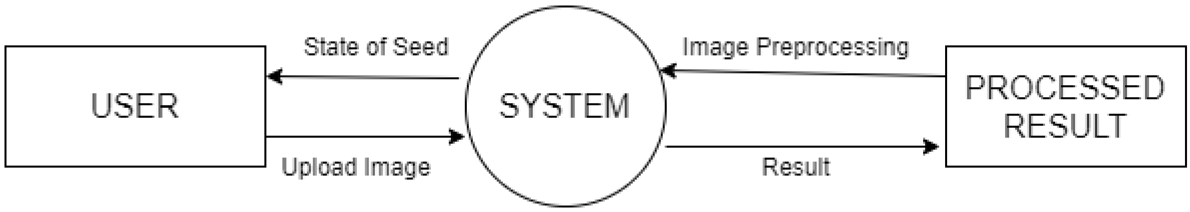


Fig 3.3 DFD - 0

3.4.2 DFD-1

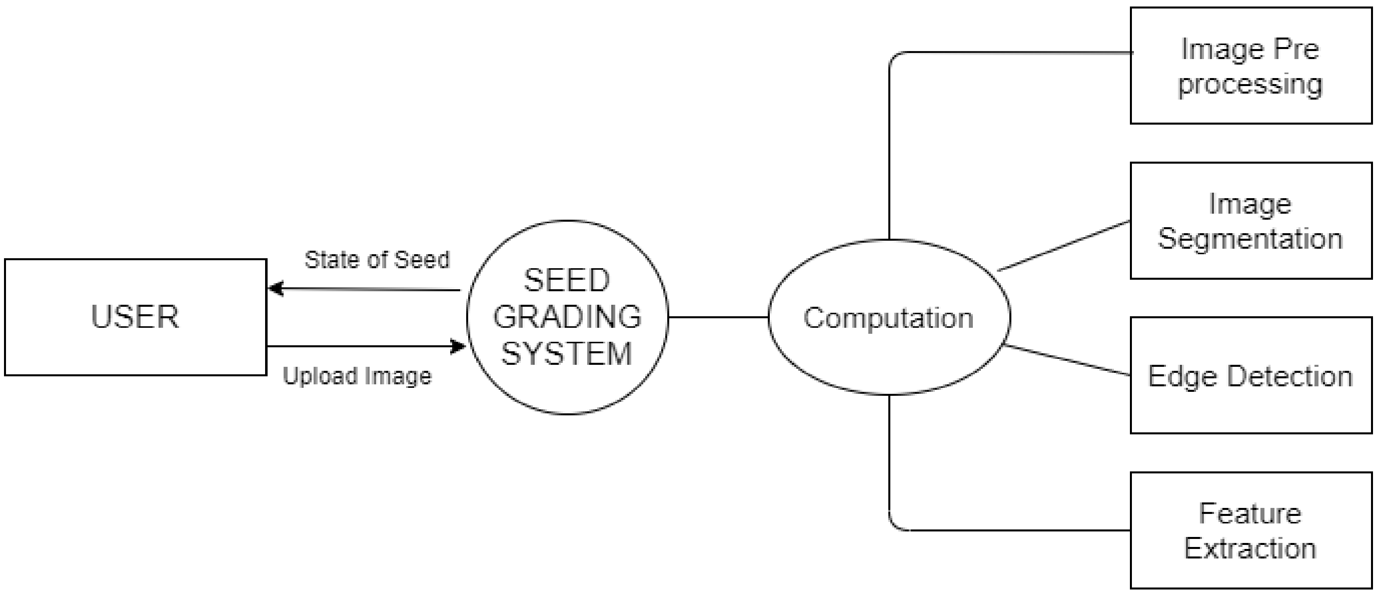


Fig 3.4 DFD - 1

3.4.3 DFD-2

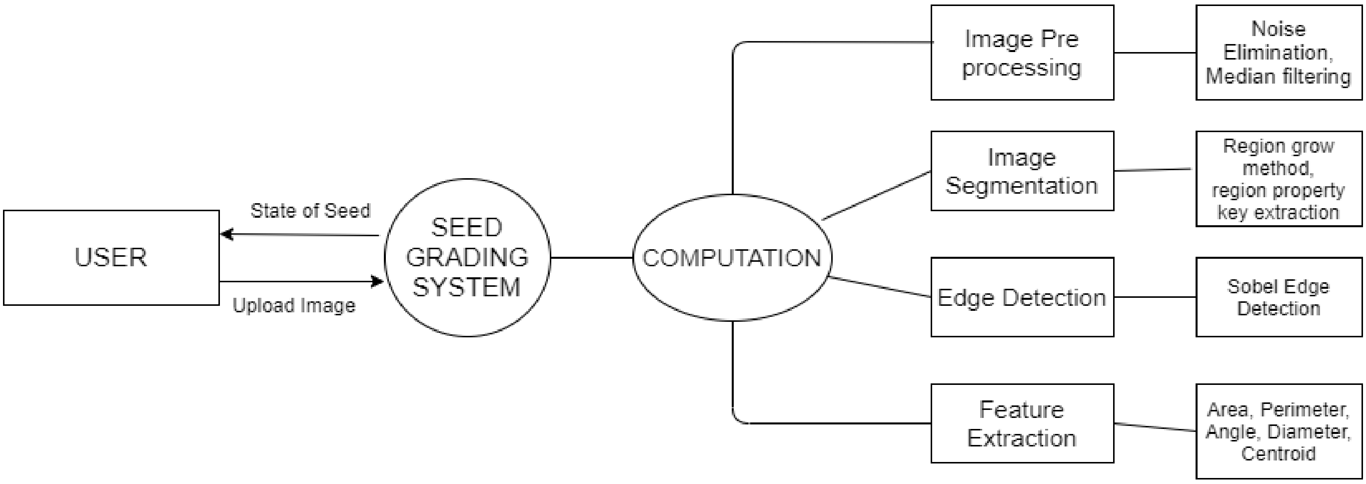


Fig 3.5 DFD - 2

3.5 UML Diagram

3.5.1 Use Case Diagram

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behavior, and not the exact method of making it happen. Use cases once specified can be denoted both textual and visual representation . A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

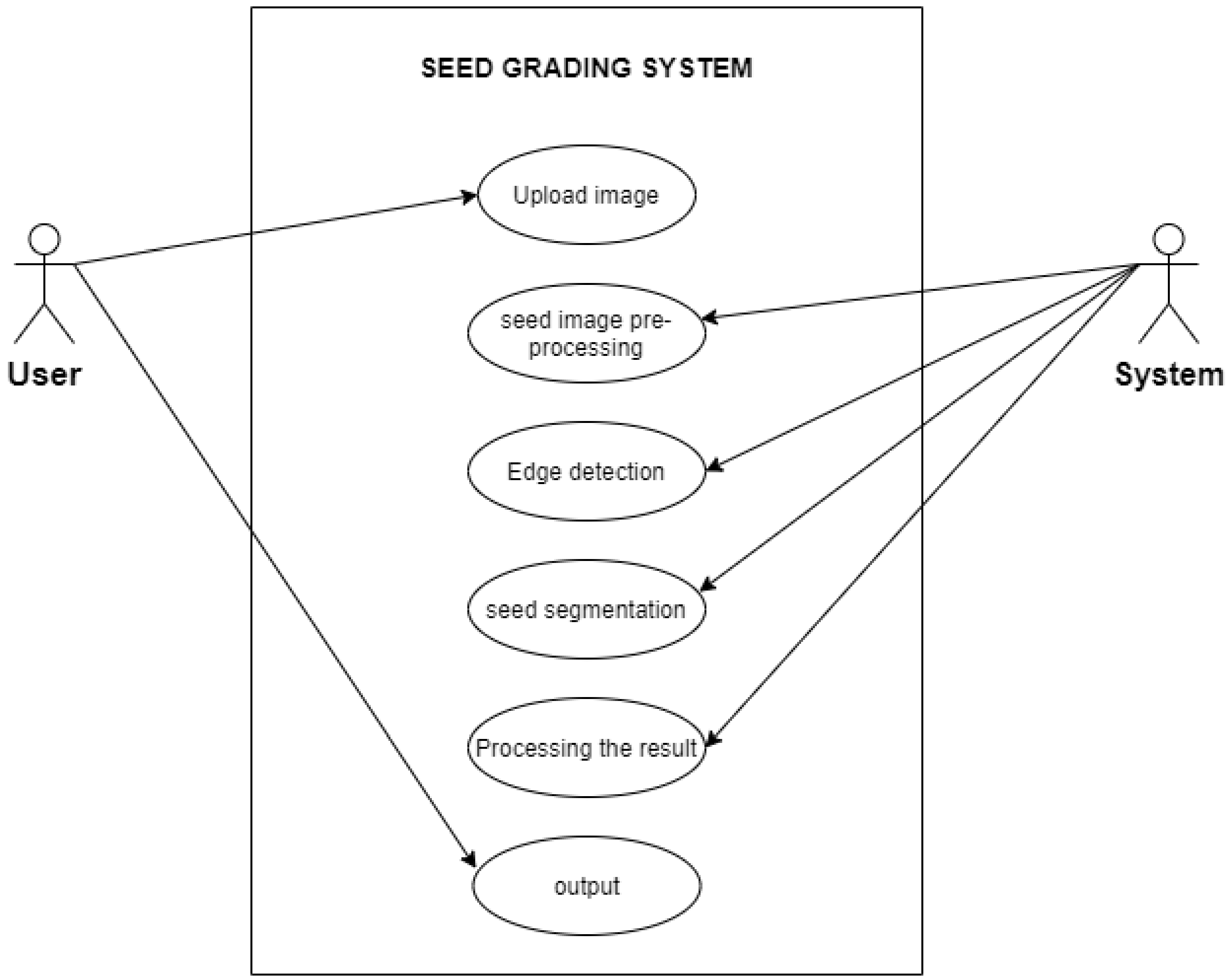


Fig 3.6 Use Case Diagram

3.5.2 Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

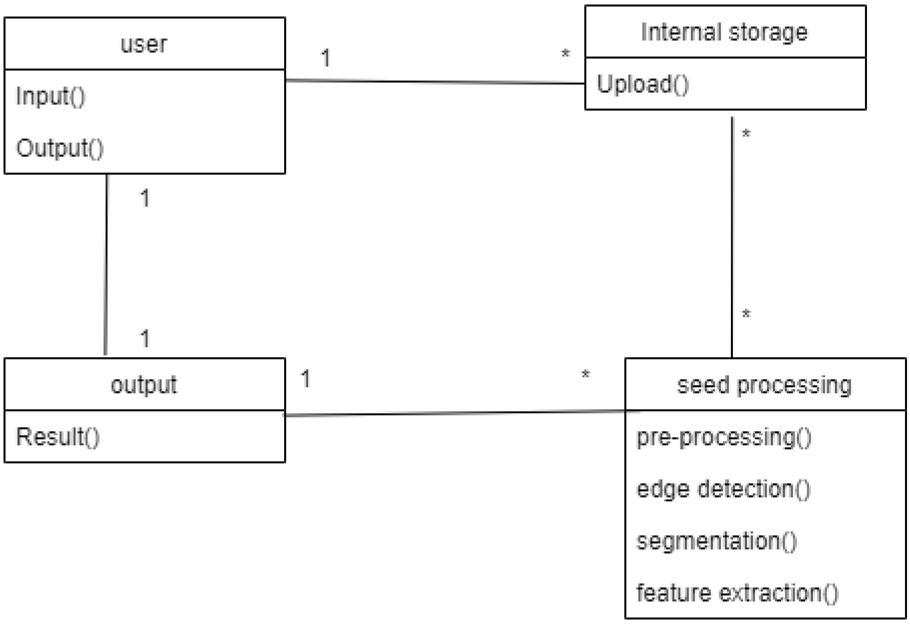


Fig 3.7 Class Diagram

3.5.3 Sequence Diagram

UML Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focused and they show the order of the interaction visually by using the vertical axis of the diagram to represent time, what messages are sent and when.

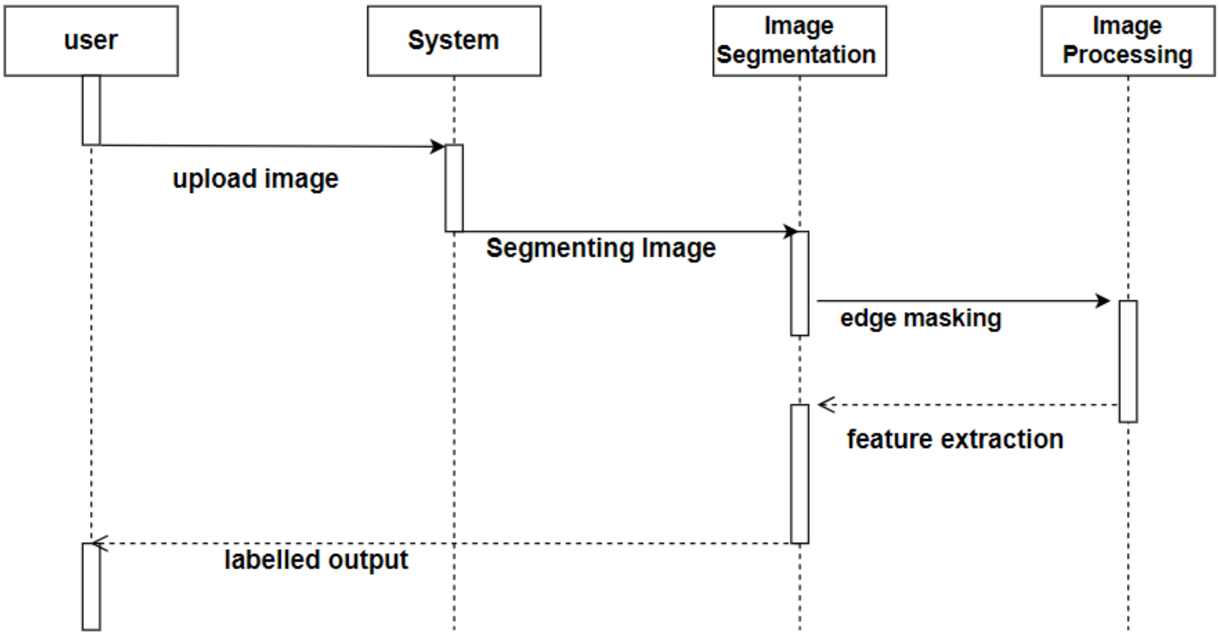


Fig 3.8 Sequence Diagram

3.5.4 Collaboration Diagram

A Collaboration is similar to a sequence diagram but the message in number format. A collaboration diagram resembles a flowchart that portrays the roles, functionality and behaviour of the individual objects as well as the overall objectives of the system in real time.

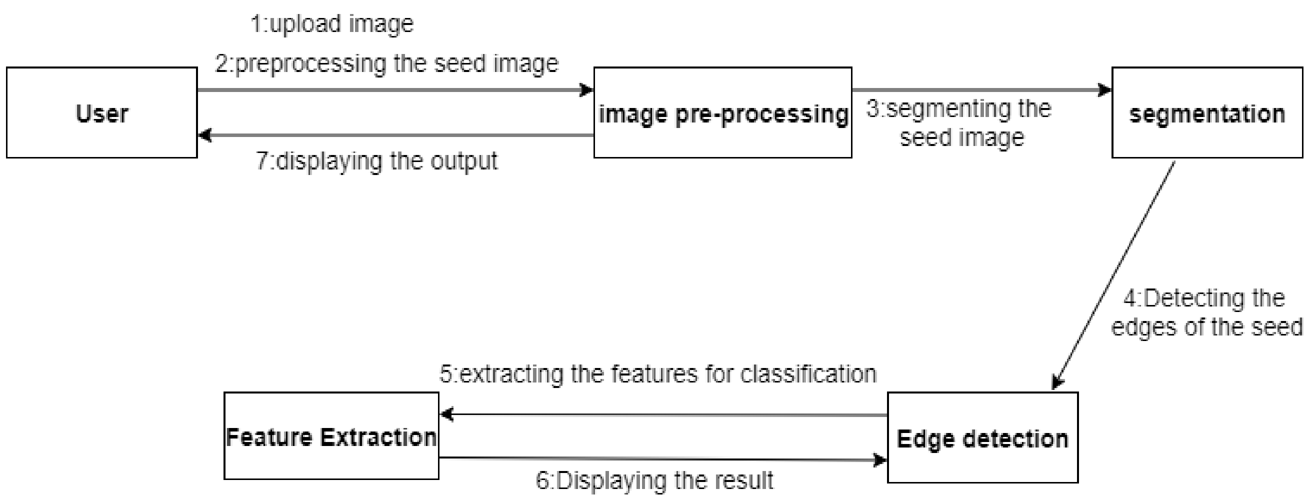


Fig 3.9 Collaboration Diagram

3.5.5 State Chart Diagram

The state chart diagram contains the states in the rectangle boxes and starts are indicated by the dot and finish is indicated by the dot encircled. The purpose of a state chart diagram is to understand the algorithm in the performing method.

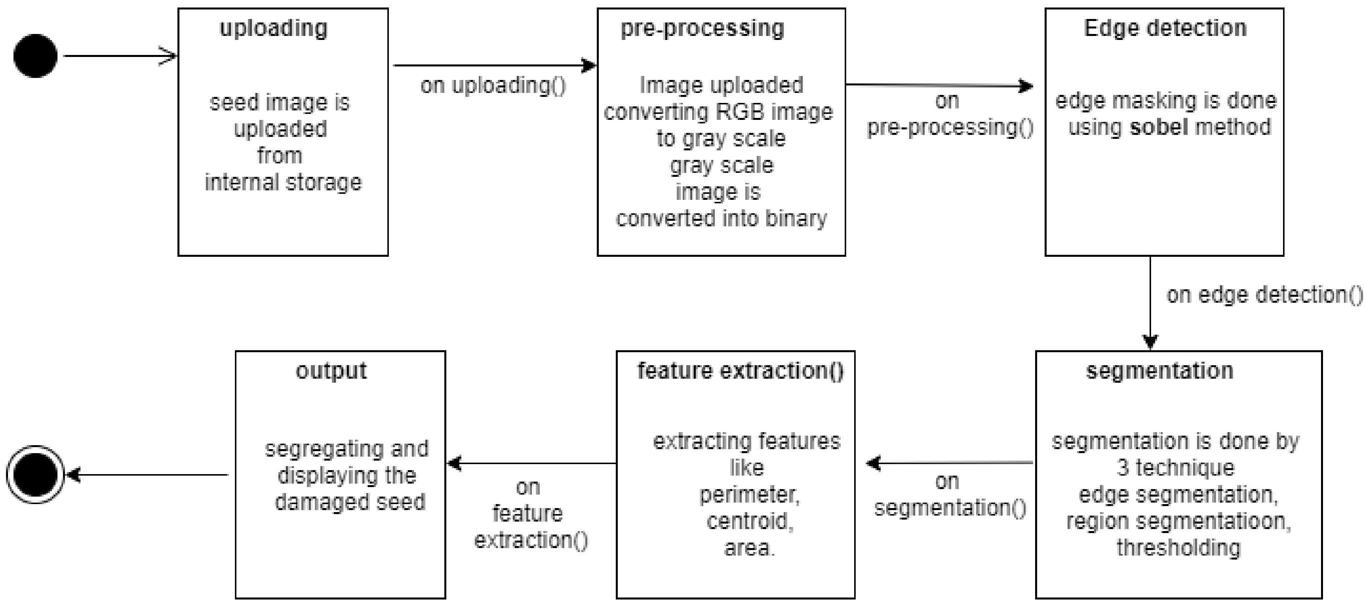


Fig 3.10 State Chart Diagram

3.5.6 Activity Diagram

Activity diagrams are graphical representations of work flow of stepwise activities and actions with support for choice, iteration and concurrency. An activity diagram shows the overall flow of the control.

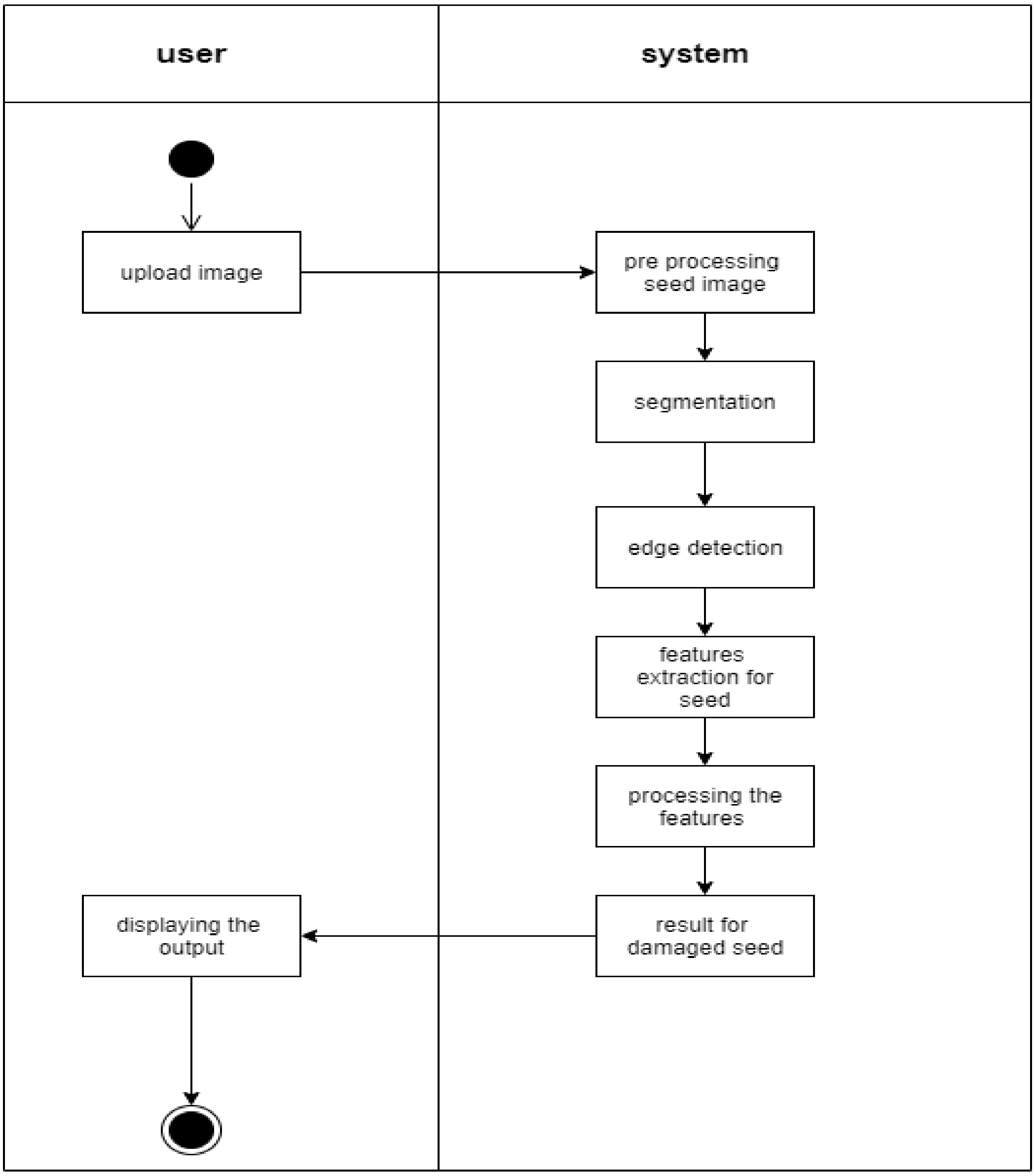


Fig 3.11 Activity Diagram

3.5.7 Component Diagram

The component diagram depicts how components are wired together to form larger components or software systems. They are used to illustrate the structure of an arbitrarily complex system.

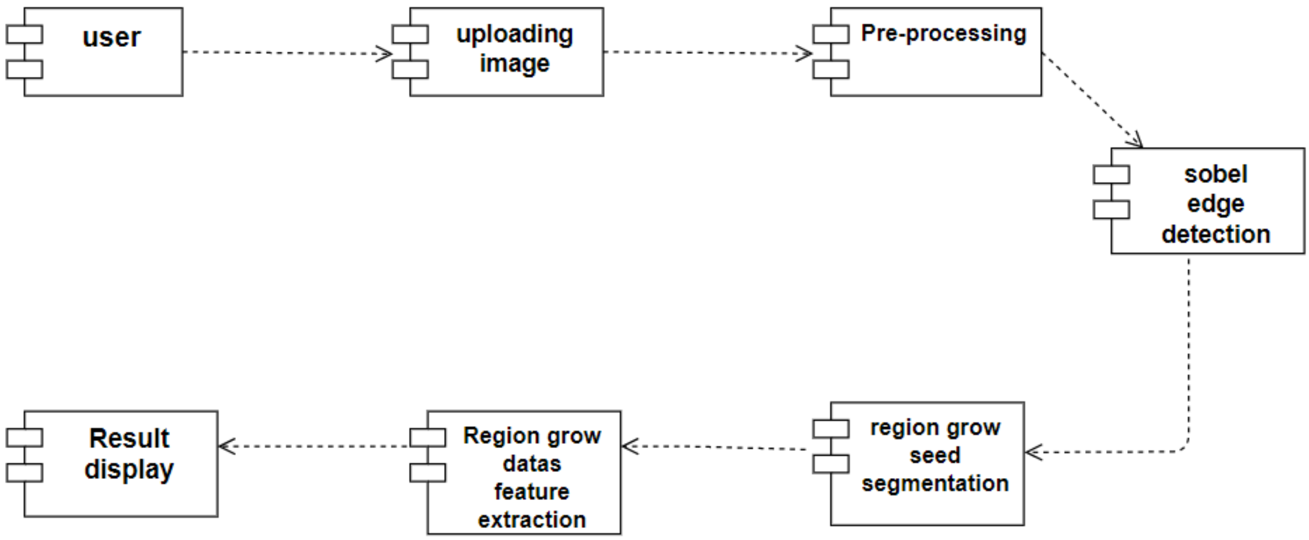


Fig 3.12 Component Diagram

3.5.8 Deployment Diagram:

A deployment diagram models the physical deployment of artifacts on nodes. It shows what hardware components exist, what software components run on each nodes and how the different pieces are connected.

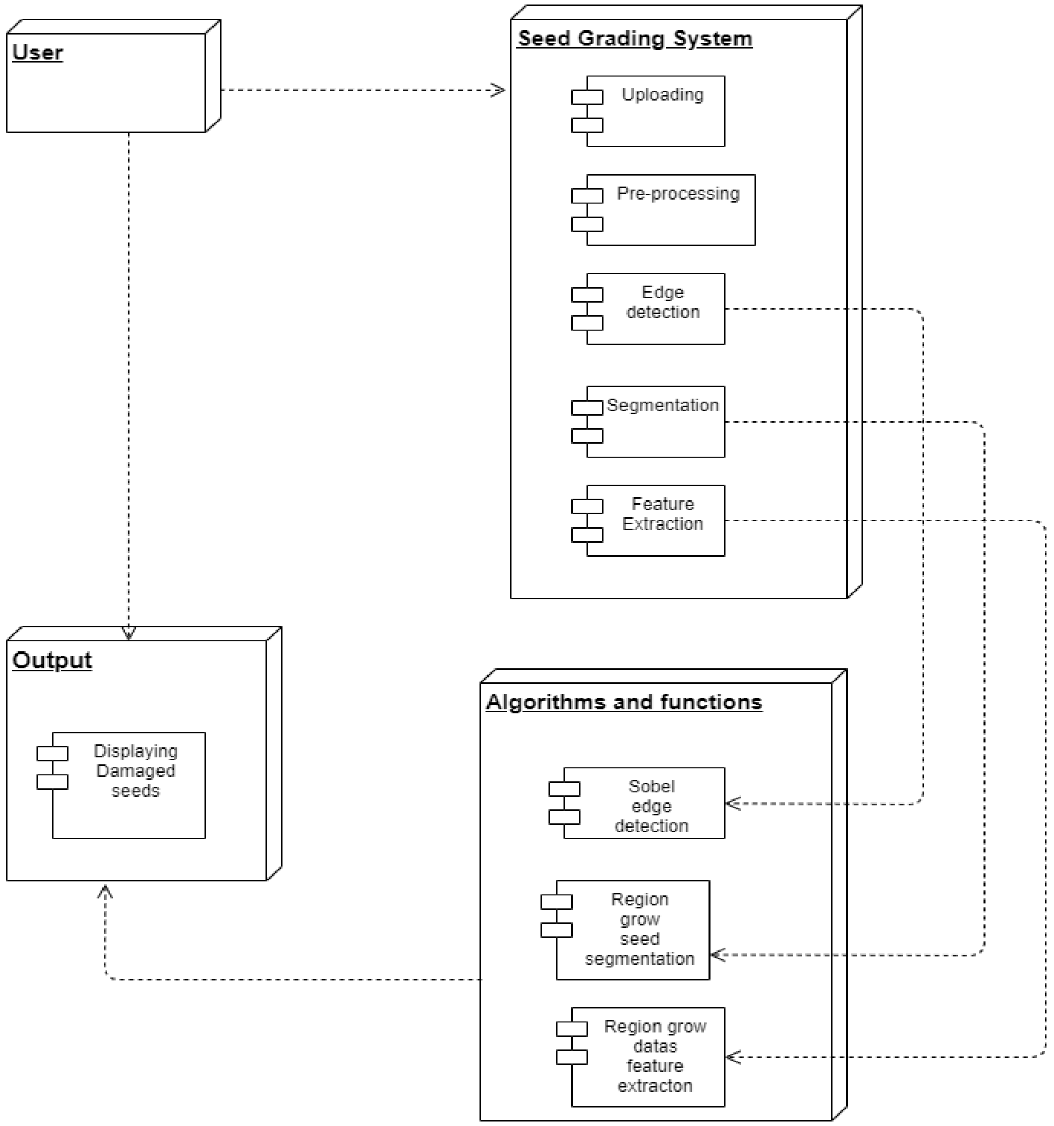


Fig 3.13 Deployment diagram

3.5.9 Package Diagram

Package diagram, a kind of structural diagram, shows the arrangement and organization of model elements in a middle to large scale project. Package diagrams can show both structure and dependencies between subsystems or modules, showing different views of a system, for example, as multi-layered application - multi-layered application model.

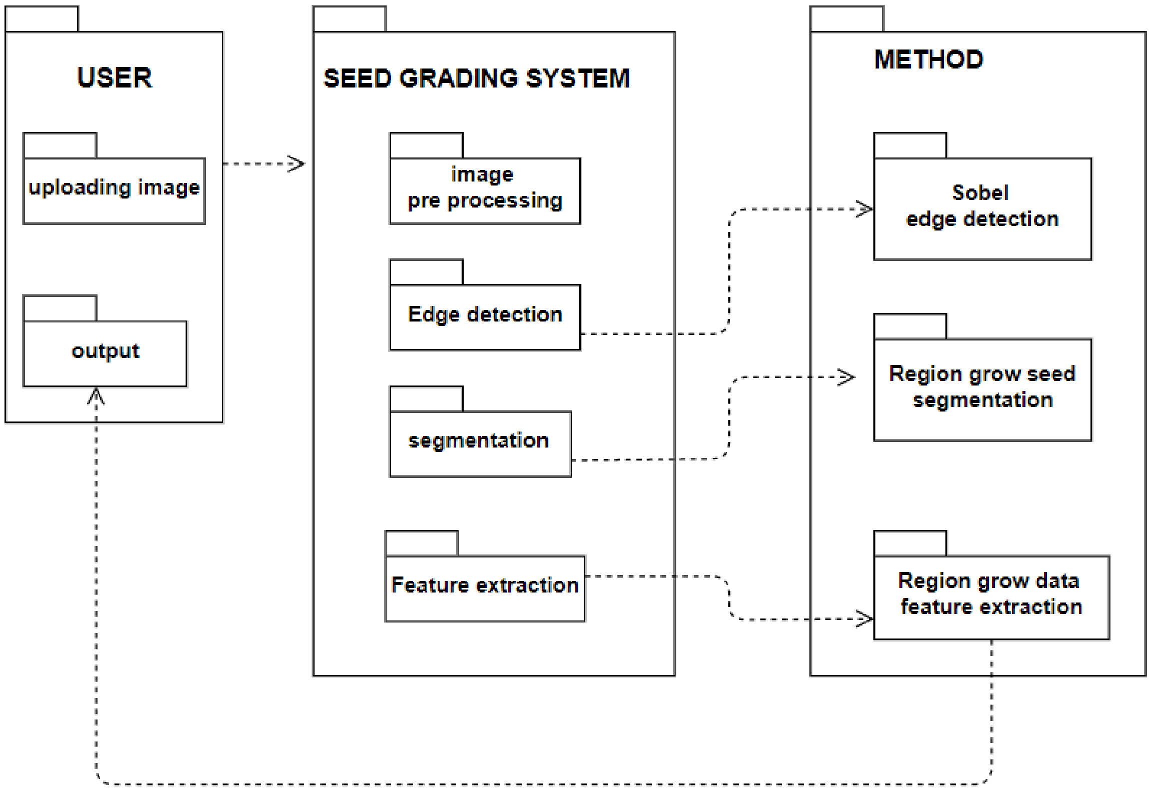


Fig 3.14 Package diagram

3.6 Summary

The nine(9), (Use case diagram, Class diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram, Component diagram, Deployment diagram, Package Diagram) Unified Modeling Diagrams for the system is drafted for a better understanding of the system's objects, Functions and classes.

CHAPTER 4

SYSTEM REQUIREMENTS

- The user should be able to upload the image of the seeds.
- The image to be uploaded should be comprehensible and unambiguous.
- The seed image should be free of any shadows.
- After uploading the user should be able to get the status of the seed.

4.1 Functional Requirements

4.1.1 Software Requirements

- Techniques to be used for implementation
  - Sobel edge detection
  - Noise filtering
  - Region growing algorithm
- Software Requirement:
  - MATLAB R2018a
  - Image Processing Toolbox
  - Computer Vision Toolbox

4.1.2 Hardware Requirements:

- Operating System - Windows 10(version 1709 or higher), Windows 7 Service pack 1, Windows Server 2019, Windows Server 2016.

- Processors - Minimum: Any Intel or AMD x86-64 processor.
  Recommended: Any Intel or AMD x86-64 processor with four logical cores and AVX2 instruction set support

- Disk - Minimum: 3 GB of HDD space for MATLAB only, 5-8 GB for a typical installation.
  Recommended: An SSD is recommended.

- Memory (RAM) - Minimum: 4 GB.
  Recommended: 8 GB

4.2 Nonfunctional Requirements

- Seed image should be returned after processing with respected labels.
- It should be possible to upload and store seed image directly and get their grades immediately.
- Larger size files manipulation should also be possible

4.2.1 Performance Requirements

For more improved performance we would need the following requirements:

- Noiseless datasets for achieving expected results
- Seed grade if processed from a clearly uploaded image would render better results than directly shooted from the system.

4.3 Summary

All the mentioned requirements are satisfied by installing all the necessary software packages for implementing seed segmentation and processing, therefore this helps us to meet the functional and non-functional requirements.

CHAPTER 5
IMPLEMENTATION

Image processing has numerous applications in many technical fields, but still its application in grading of seeds is not widely heard of. In this design, image processing technique is utilized for the grading of seeds, which helps the system in evaluating the number of damaged seeds.

5.1 Why Image Processing techniques?

The quality of seeds has a major effect on the yield, so the proper inspection of seed quality is very much important nowadays. The state of seed is one of the factors whose inspection is more difficult and more complicated than that of other factors. Presently, the grading of seed mainly depends on manual labour or by using machinery . These two methods can give more exact results but they have many limitations. Manual methods are time consuming and can also result in parallax error. Similarly the use of seed grading machines is very expensive and at the same time not very accurate. The nondestructive identification of seed grading on a large scale cannot be achieved accurately by using these two methods. So a non destructive, inexpensive inspection using machine vision based on digital image processing technology is much faster and efficient.

5.2 System Architecture

The basic workflow of the system

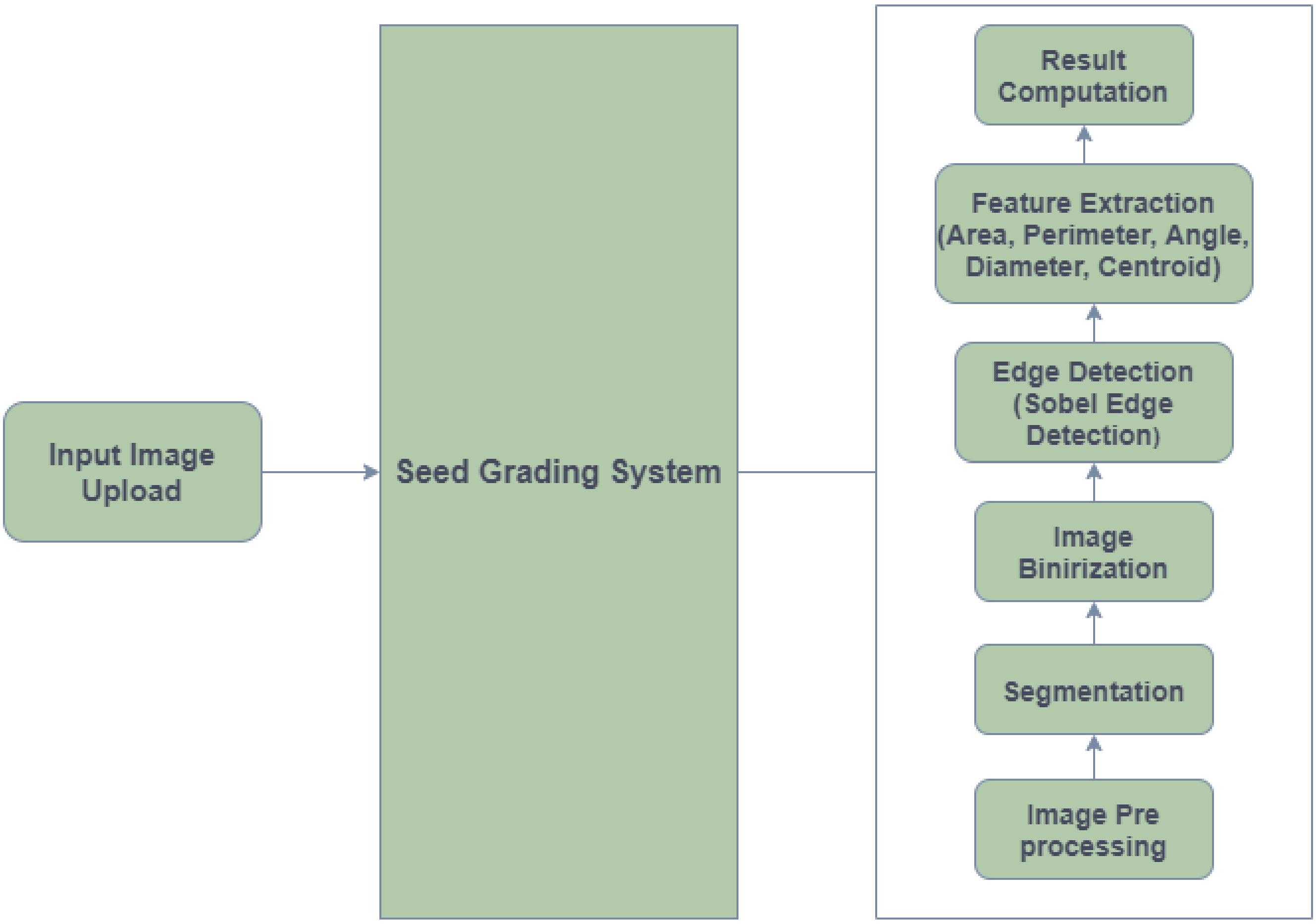


Fig 5.1 System Architecture

5.3 System Description

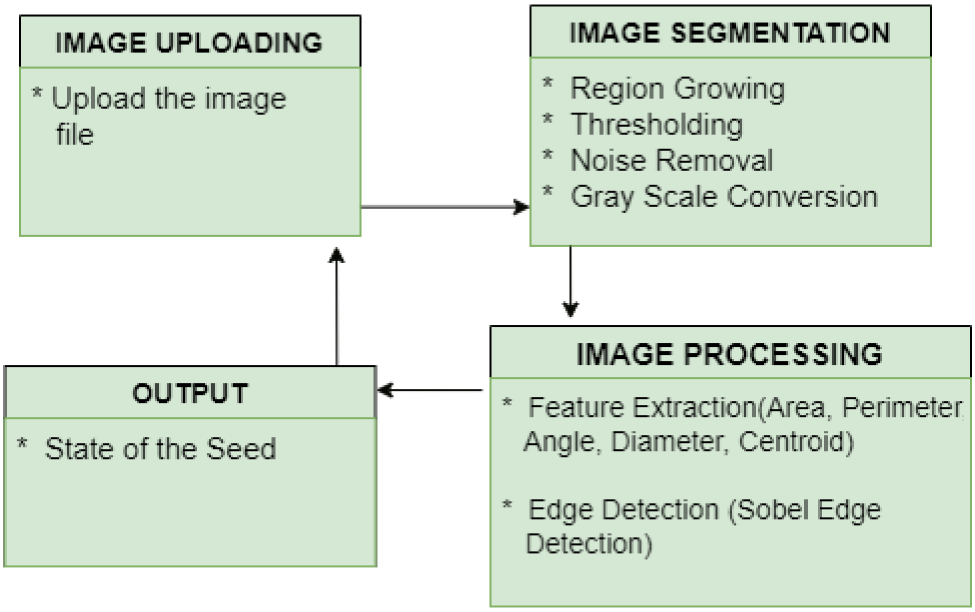


Fig 5.2 Modules of the System

5.3.1 Seed Image Uploading

The user will have to simply capture the seeds and upload into a respective file which is taken as input by the system for further processing to produce the desired outcome .

5.3.2 Image Analysis:

It is the process of differentiating the seed from the background and extracting quantitative information, which is used further for the decision making process.

5.3.3.1 Image Acquisition and Smoothening:

Image Acquisition is the first step in image processing. It is acquired getting the desired image for processing. Smoothing is done using Median Filters. Median filter is used for pre-processing, because it preserves the edges of the image during noise removal. Median filtering is extensively used in digital imaging since it conserves the ends of the image during noise exclusion. Salt and pepper noise are which with, median filters are predominantly effectual. Using median filter the noise in the input gray color image is removed.

5.3.3.2 Segmentation:

The subsequent step is to segment an image, which is one of the imperative stages in image analysis. The field of mathematical morphology contributes to a wide range of operators to image processing. We are only handling binary images. For a binary image, black pixels ("0") are normally taken to represent background regions, while white pixels ("1") denote foreground. The segmented image undergoes a series of morphological operations to detect the exact shape of the object. Segmentation is attained by three techniques such as Edge segmentation, Region segmentation and Thresholding

5.3.3.3 Thresholding:

It is the method of image segmentation. From a grey scale image threshold can create a binary image.This technique is based on absorption of light in their surfaces to characterize the regions of the image. Threshold is to separate the regions in an image with respect to the objects, which is to be analyzed. The separation of region is based on the variation of intensity between the object pixels and the background pixels.

5.3.3.4 Edge Detection:

Edge detection is based on recognition of edges by diverse edge operators. Discontinuities in color, Grey level, texture, etc. are detected by edge operators.

Sobel Edge Detection

The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image.

5.3.3.5 Region Growing Algorithm

The edge masked image is now used as a input for Region Growing Algorithmic approach to segmentation examines neighboring pixels of initial seed points and determines whether the pixel neighbors should be added to the region or not. The process is iterated on, in the same manner as general data clustering algorithms. Forming all pixels of similar size into a single region simply by checking the continuity and discontinuity of pixels.

5.3.3.6 Feature Extraction

The isolated region from the region growing algorithm is now used for, Extraction of quantitative information from segmented images is dealt with Feature Extraction. Object recognition and classifications is done based on various algorithms of morphological features. Some of the morphological features for classification purposes contain redundant, noisy and irrelevant information. The features which extracted from images of seed is Perimeter, Area, Minor-axis Length and Major-axis Length, diameter and centroid.

Algorithm

- Find all connected components in S(x,y) and erode them to 1 pixel.
- Form image fq (x,y)=1 if f (x,y) satisfies the predicate.
- Form image g(x,y)=1 for all pixels in fq (x,y) that are 8- connected with any seed point in S(x,y).
- Label each connected component in g(x,y) with a different label.

The predicate used for region growing is to compare the absolute difference between a seed point and a pixel to a threshold. If the difference is below it we accept the pixel as null.

5.4 Summary

These are modules available in the application which has its purpose of getting input, manipulation of the input and displaying the required output.

CHAPTER 6

PERFORMANCE EVALUATION

6.1 Eventual Progress and Accuracy:

Stage 1: The application was started with grading of a single seed in Machine Learning using python, but since we were using our own data set to train the system this gave an accuracy of only of 64%

Stage 2: To achieve more accuracy we moved our working platform from Anaconda to MATLAB using C.This gave us an accuracy of 98% for grading a single seed.

Stage 3: Then we eventually increased our requirement to group of seeds, mean while changing the complexity of our code,and finally achieved our end product with an accuracy of 92%

6.2 Summary:

On applying Region Grow with image segmentation and edge masking and with the use of Region Grow Property Key extraction we have modeled a system to upload a seed image to get the grade for that.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

In this project the Region Grow Algorithm profitably applied for grading seeds. Damaged seeds are detected among the good ones. And the labeled output image is displayed. The method used for seed identification is more optimized than the previous methods. Thus, this project can be used as an aid for farmers to produce profitable yields.

7.2 Future Enhancements

The current model accepts input images that are uploaded by the user. In future a camera module that captures, stores and accepts input images on spot could be implemented. Also the current model displays the output only using the morphological features of the seed, In future an option for color and texture based classification could be implemented.

REFERENCES

[1] Gurpreet Kaur & Bhupinder Verma, Measurement standards based grading of rice kernels by separating touching kernels for embedded imaging applications, International Journal of Electronics, Communication & Instrumentation Engineering Research and Development (IJECIERD) ISSN 2249-684X, Vol. 3, Issue 1, pp 127-134, Mar 2013.

[2] Salome Hema Chitra,S.Suguna and S.Naganandini Sujatha, "A Survey on Image Analysis Techniques in Agricultural Product", Indian Journal of Science and Technology, Vol 9(12), DOI:10.17485/ijst/2016/v9i12/77171, March 2016

[3] G.Ajay, M.Suneel,Quality evalution of rice grains using morphological methods, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, pp 35-37, January 2013.

[4] "Digital image processing using MATLAB " Third edition" by Rafael C. Gonzalez

[5] R. M. Haralick, K. Shanmugam, and I. Din stein, "Texture features for image classification," IEEE Trans. on System. Man, and cybern, vol 6, pp. 610-621, 1973.

[6] Ms.Sushma,Mr.Krishan Kumar,Mr.Rajender Kumar quality testing and grading of food grains using digital image processing IJEEEVol No.8 Issue No. 02 July-December 2016

[7] Salome Hema Chitra,S.Suguna and S.Naganandini Sujatha, "A Survey on Image Analysis Techniques in Agricultural Product", Indian Journal of Science and Technology, Vol 9(12), DOI:10.17485/ijst/2016/v9i12/77171, March 2016

[8] Chris Solomon TB. Fundamentals of Digital Image Processing.

[9] Sukhvir Kaur DS. Geometric Feature Extraction of Selected Rice Grains using Image Processing Techniques International Journal of Computer Applications. 2015;124(8).

[10] H. Rautio and O. Silvn, "Average Grain Size Determination using Mathematical Morphology and Texture Analysis".

[11] Monica Jhuria, Ashwani Kumar, Rushikesh Borse, "Image Processing For Smart Farming: Detection Of Disease And Fruit Grading", Proceedings of

the 2013 IEEE Second International Conference on Image Information Processing.

Appendix I : Sample Coding

```
clc;
clear all;
close all;
[filename,pathname]=uigetfile({'*.bmp;*.tif;*.tiff;
*.jpg;*.jpeg;*.gif','IMAGE
Files(*.bmp,*.tif,*.tiff,*.jpg,*.jpeg,*.gif)'},'Chose
Image File');
img=imread(cat(2,pathname,filename));
img=imresize(img,[512,512]);
inp_data=img;
figure;
imshow(inp_data);
title('Input image');
[xx yy xy]=size(img);
if xy==3
   im1=rgb2gray(img);
else
im1=img;
end
im1=medfilt2(im1,[3 3]); %Median filtering the
image to remove noise%
BW = edge(im1,'sobel'); %finding edges
[imx,imy]=size(BW);
msk=[0 0 0 0 0;
```

```
    0 1 1 1 0;
    0 1 1 1 0;
    0 1 1 1 0;
    0 0 0 0 0;];
B=conv2(double(BW),double(msk));
L = bwlabel(B,8);
mx=max(max(L));
[r,c] = find(L==17);
rc = [r c];
[sx sy]=size(rc);
n1=zeros(imx,imy);
for i=1:sx
   x1=rc(i,1);
   y1=rc(i,2);
   n1(x1,y1)=255;
end
figure,imshow(im1);
figure,imshow(B);
title('Edge Masking');
seg = subfunct(inp_data,'whole',400,0.02,'vector');
figure;
imshow(seg);
rgb = img;
% figure
% imshow(rgb)
d = imdistline;
delete(d);
gray_image = rgb2gray(rgb);
%imshow(gray_image);
```

```
[centers, radii] = imfindcircles(rgb,[20
200],'ObjectPolarity','dark', ...
   'Sensitivity',0.9,'Method','twostage');
 [val, pos]=min(centers);
 if pos(1)==pos(2)
    positon=pos(2);
 else
    [val2, pos2]=min(pos);
    positon=pos((pos2));
 end
imshow(rgb);
hold on
text_str =  'Damaged';
for i=1:length(centers)
   if i~=positon
h = viscircles(centers(i,:),radii(i,:),'Color','b');
   end
end
hold off
figure;
position = centers(i,:)-10;
RGB_text = insertText (rgb, position, text_str,
'FontSize', 18, 'BoxColor', 'red', 'TextColor',
'black');
imshow(RGB_text);

function indicator = checkstop(old,new,dt)
layer = size(new,3);
```

```
for i = 1:layer
    old_{i} = old(:,:,i);
    new_{i} = new(:,:,i);
end
if layer
    ind = find(abs(new)<=.5);
    M = length(ind);
    Q = sum(abs(new(ind)-old(ind)))./M;
    if Q<=dt*.18^2
        indicator = 1;
    else
        indicator = 0;
    end
else
    ind1 = find(abs(old_{1})<1);
    ind2 = find(abs(old_{2})<1);
    M1 = length(ind1);
    M2 = length(ind2);
    Q1 =
sum(abs(new_{1}(ind1)-old_{1}(ind1)))./M1;
    Q2 =
sum(abs(new_{2}(ind2)-old_{2}(ind2)))./M2;
    if Q1<=dt*.18^2 && Q2<=dt*.18^2
        indicator = 1;
    else
        indicator = 0;
    end
end
return
```

```
function H=Heaviside(z)
Epsilon=10^(-5);
H=zeros(size(z,1),size(z,2));
idx1=find(z>Epsilon);
idx2=find(z<Epsilon & z>-Epsilon);
H(idx1)=1;
for i=1:length(idx2)
H(idx2(i))=1/2*(1+z(idx2(i))/Epsilon+1/pi*sin(pi
*z(idx2(i))/Epsilon));
end;

function KG = kappa(I)
I = double(I);
[m,n] = size(I);
P = padarray(I,[1,1],1,'pre');
P = padarray(P,[1,1],1,'post');
fy = P(3:end,2:n+1)-P(1:m,2:n+1);
fx = P(2:m+1,3:end)-P(2:m+1,1:n);
fyy = P(3:end,2:n+1)+P(1:m,2:n+1)-2*I;
fxx = P(2:m+1,3:end)+P(2:m+1,1:n)-2*I;
fxy =
0.25.*(P(3:end,3:end)-P(1:m,3:end)+P(3:end,1:n)
-P(1:m,1:n));
G = (fx.^2+fy.^2).^(0.5);
K =
(fxx.*fy.^2-2*fxy.*fx.*fy+fyy.*fx.^2)./((fx.^2+fy.
^2+eps).^(1.5));
KG = K.*G;
```

```
KG(1,:) = eps;
KG(end,:) = eps;
KG(:,1) = eps;
KG(:,end) = eps;
KG = KG./max(max(abs(KG)));


function m = maskcircle2(I,~)
if size(I,3)~=3
   temp = double(I(:,:,1));
else
   temp = double(rgb2gray(I));
end
h = [0 1 0; 1 -4 1; 0 1 0];
T = conv2(temp,h);
T(1,:) = 0;
T(end,:) = 0;
T(:,1) = 0;
T(:,end) = 0;
thre = max(max(abs(T)))*.5;
idx = find(abs(T) > thre);
[cx,cy] = ind2sub(size(T),idx);
cx = round(mean(cx));
cy = round(mean(cy));
[x,y] =
meshgrid(1:min(size(temp,1),size(temp,2)));
m = zeros(size(temp));
[p,q] = size(temp);
       r = 9;
```

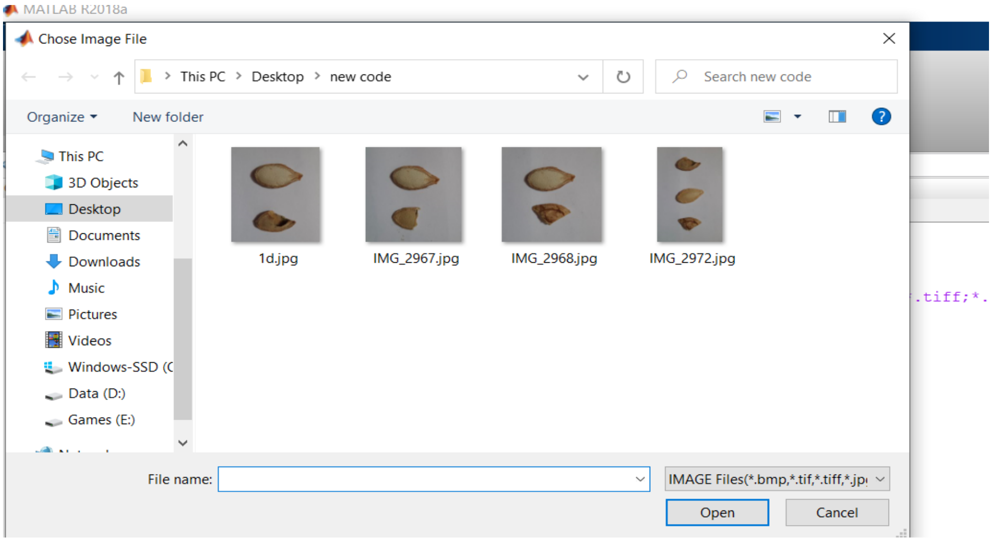
```
        m =
zeros(round(ceil(max(p,q)/2/(r+1))*3*(r+1)));
        siz = size(m,1);
        sx = round(siz/2);
        i = 1:round(siz/2/(r+1));
        j = 1:round(0.9*siz/2/(r+1));
        j = j-round(median(j));
        m(sx+2*j*(r+1),(2*i-1)*(r+1)) = 1;
        se = strel('disk',r);
        m = imdilate(m,se);
        m =
m(round(siz/2-p/2-6):round(siz/2-p/2-6)+p-1,roun
d(siz/2-q/2-6):round(siz/2-q/2-6)+q-1);
        tem(:,:,1) = m;
        M =
padarray(m,[floor(2/3*r),floor(2/3*r)],0,'post');
        tem(:,:,2) =
M(floor(2/3*r)+1:end,floor(2/3*r)+1:end);
        m = tem;
return

function showphi(I, phi, i)
for j = 1:size(phi,3)
    phi_{j} = phi(:,:,j);
end
imshow(I,'initialmagnification','fit','displayrange',
[0 255]);
  hold on;
```

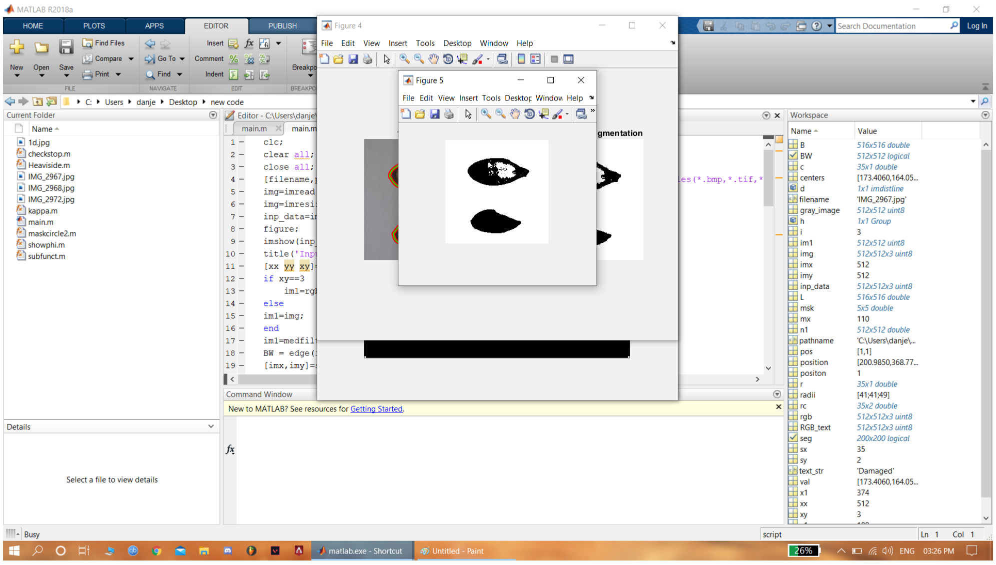
```
  if size(phi,3) == 1
      contour(phi_{1}, [0 0], 'r','LineWidth',4);
      contour(phi_{1}, [0 0], 'g','LineWidth',1.3);
  else
      contour(phi_{1}, [0 0], 'r','LineWidth',4);
      contour(phi_{1}, [0 0], 'x','LineWidth',1.3);
      contour(phi_{2}, [0 0], 'g','LineWidth',4);
      contour(phi_{2}, [0 0], 'x','LineWidth',1.3);
  end
  hold off;
  title([num2str(i) ' Iterations']);
  drawnow;

function seg = subfunct(I,mask,num_iter,mu,~)
   s = 200./min(size(I,1),size(I,2));
   if s<1
       I = imresize(I,s);
   end
             mask = maskcircle2(I,'whole');


        s = 200./min(size(I,1),size(I,2)); % resize
scale
        I = imresize(I,s);
        mask = imresize(mask,s);
        layer = size(I,3);
        if layer == 1
            display('only one image component for
vector image')
        end
```

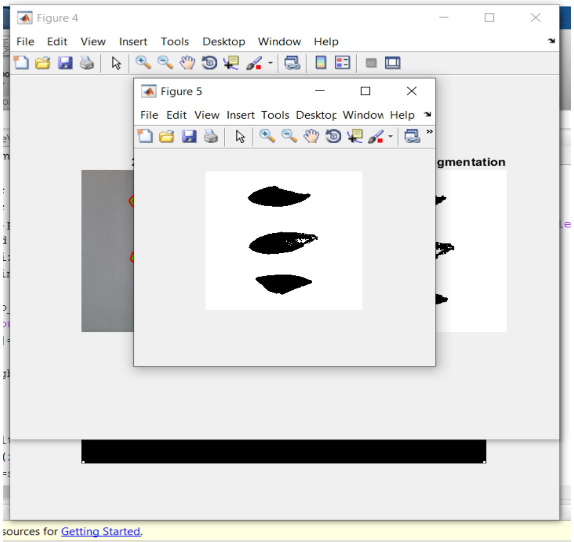
```
        P = double(I);
        mask = mask(:,:,1);
        phi0 =
bwdist(mask)-bwdist(1-mask)+im2double(mask)-
.5;
        force = eps;
          figure();
          subplot(1,2,1); title('Segmentation');
          for n=1:num_iter
              inidx = find(phi0>=0);
              outidx = find(phi0<0);
              force_image = 0;
              for i=1:layer
                  L = im2double(P(:,:,i));
                  c1 =
sum(sum(L.*Heaviside(phi0)))/(length(inidx)+ep
s);
                  c2 =
sum(sum(L.*(1-Heaviside(phi0))))/(length(outidx
)+eps);

force_image=-(L-c1).^2+(L-c2).^2+force_image;
              end
              force =
mu*kappa(phi0)./max(max(abs(kappa(phi0))))+1/
layer.*force_image;
              force = force./(max(max(abs(force))));
              dt=0.5;
              old = phi0;
```

```
            phi0 = phi0+dt.*force;
            new = phi0;
            indicator = checkstop(old,new,dt);
            if(mod(n,20) == 0)
              showphi(I,phi0,n);
            end;
            if indicator
               showphi(I,phi0,n);
               seg = phi0<=0;
               subplot(1,2,2); imshow(seg);
title('saliency region Segmentation');
               return;
            end
        end;
        showphi(I,phi0,n);
        seg = phi0<=0;
        subplot(1,2,2); imshow(seg); title('Object
Region Segmentation');
end
```

Output Snapshots:







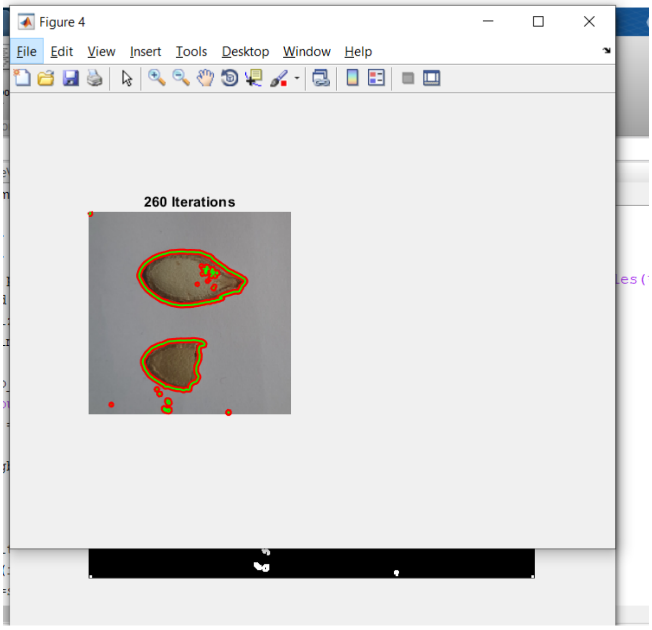


Figure 4
File Edit View Insert Tools Desktop Window Help
260 Iterations

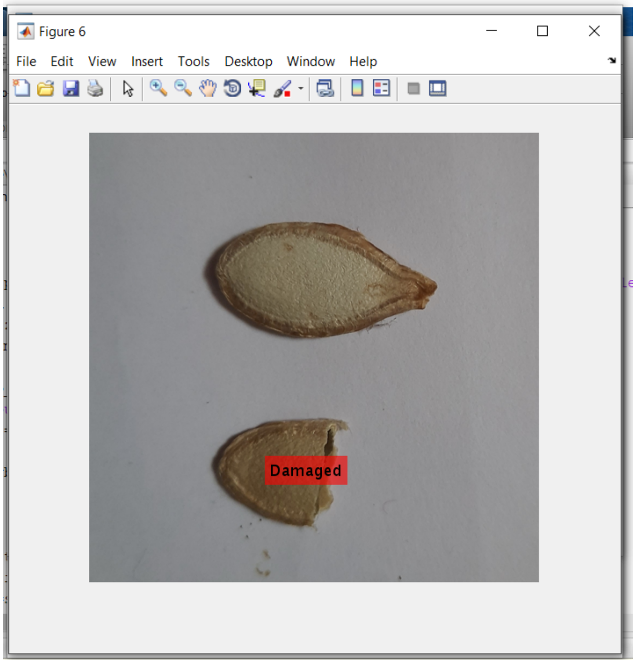


Figure 6
File Edit View Insert Tools Desktop Window Help
Damaged

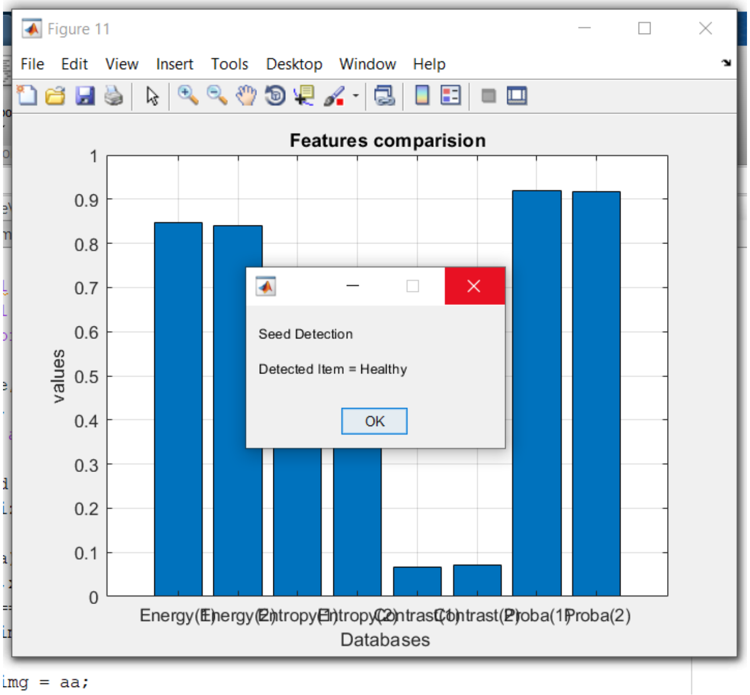


Figure 11
File Edit View Insert Tools Desktop Window Help
Features comparision
values
Databases
Energy(1) Energy(2) Entropy(1) Entropy(2) Contrast(1) Contrast(2) Proba(1) Proba(2)
Seed Detection
Detected Item = Healthy
OK
img = aa;

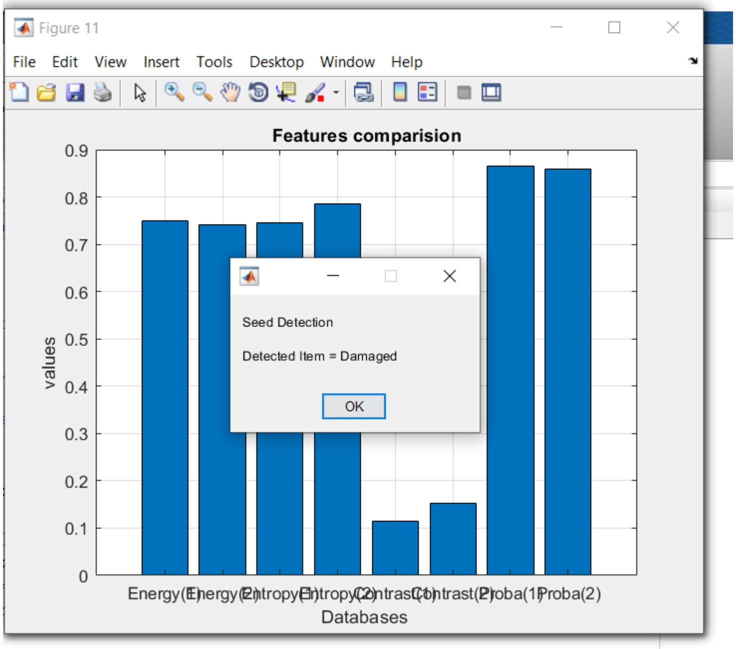


Figure 11
File Edit View Insert Tools Desktop Window Help
Features comparision
values
Databases
Energy(1) Energy(2) Entropy(1) Entropy(2) Contrast(1) Contrast(2) Proba(1) Proba(2)
Seed Detection
Detected Item = Damaged
OK