DEVELOPMENT OF AN AUTOMATED GRADING SYSTEM OF WHITE PEA BEANS USING IMAGE PROCESSING TECHNIQUES CONVERGENCE WITH ANN

BY

MESFIN FEKADU ABEZA

A Thesis Submitted as a Partial Fulfillment to the Requirements for the Award of the Degree of Master of Science in Software Engineering

To

DEPARTMENT OF SOFTWARE ENGINEERING
COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

August 2019
Approval Page

This is to certify that the thesis prepared by Mesfin Fekadu Abeza, titled: Development of an automated Grading system of white pea beans using image processing techniques convergence with ANN and submitted in partial fulfillment of the requirements for the Degree of Master of Science in Software Engineering complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Board:

| Advisor          | Signature, Date |
|------------------|-----------------|
| Co-advisor       | Signature, Date |
| External Examiner| Signature, Date |
| Internal Examiner| Signature, Date |
| Chairperson      | Signature, Date |
| DGC Chairperson  | Signature, Date |
| College Dean/Associate Dean for GP | Signature, Date |
Declaration

I hereby declare that this thesis entitled “Development of an automated grading system of white pea bean using image processing techniques convergence with ANN” was prepared by me, with the guidance of my advisor. The work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted, in whole or in part, for any other degree or professional qualification.

Author: ________________________________  Signature, Date: ________________________________

Witnessed by: ________________________________

Name of student Advisor: ________________________________  Signature, Date: ________________________________

Name of student Co-advisor: ________________________________  Signature, Date: ________________________________
ABSTRACT

White pea bean is a very important crop where its circulation in the market has to conform to the rules of quality inspection. Currently, white pea bean sample quality inspection is performed manually by human experts through visual evaluation and the constituents classified into foreign matter, rotten and diseased, healthy, broken, discolored, shriveled and pest damaged kernels. However, visual evaluation requires significant amount of time, trained and experienced people. Besides, it is affected by bias and inconsistencies associated with human nature. Such approach will not be satisfactory for large scale inspection and grading unless fully automated.

The goal of this research work is to develop a system capable of assessing the quality of White pea bean sample constituents using digital image processing techniques and artificial neural network classifier based on the standard for white pea bean set by the Ethiopian Standards Agency. A total of 24 features (14 color, 8 shape and 2 size) have been identified to model white pea bean sample constituents. For classification of White pea bean samples, a feedforward artificial neural network classifier with backpropagation learning algorithm, 24 input and 7 output nodes, corresponding to the number of features and classes respectively has been designed. The network is trained and its performance is compared against other classifiers both empirically and based on supporting facts from the literature. For the purpose of training the classifier, a total of 602 kernels and foreign matters have been collected from Ethiopian Grain Trade Enterprise. The training data is randomly apportioned into training (70%) and testing (30%). The classifier achieved an overall classification accuracy of 96.8%. The success rates for detecting foreign, rotten and diseased, healthy, broken, discolored, shriveled and pest damaged kernels are 94.9%, 96.5%, 96.3%, 97%, 97.9%, 97%, and 97.6%, respectively.

Keywords: Artificial neural network, White pea beans quality assessment, Reconstructed image, Image segmentation, Digital image processing
ACKNOWLEDGEMENTS

First and foremost, I would like to thank my God, for giving me the financial and spiritual strength to accomplish this research work in particular and my MSc in general.

Secondly, I would like to extend my gratitude to my advisor, Dr. Sudhir Kumar Mohapatra, my co-advisor Mr. Befekadu Belete for their guidance starting from the time of title selection up until the time of completion of this research work.

Thirdly, I would like to extend my special gratitude to the Ethiopian Grain Trading Enterprise for their help in obtaining white pea bean samples and for sample classification expertise they provided for this research.

I am also very thankful to my instructors and all staff members of the department of Software engineering for their contribution in one way or another for the success of my study.

Finally, I am very grateful to my father and mother and all the rest of my families, friends, and peers who, in one or the other way brought me up to a success in my academic endeavor.
Table of Contents

Approval Page .................................................................................................................. ii
Declaration ....................................................................................................................... iii
ABSTRACT....................................................................................................................... iv
ACKNOWLEDGEMENTS ................................................................................................. v
Table of Contents ........................................................................................................... vi
List of Figures .................................................................................................................. ix
List of Tables ................................................................................................................... ix
List of Abbreviations ..................................................................................................... xi

CHAPTER ONE ............................................................................................................ 1

1. INTRODUCTION ................................................................................................... 1

1.1. Background ......................................................................................................... 1

1.2. Motivation ......................................................................................................... 2

1.3. Statement of the Problem .................................................................................. 3

1.4. Objective of the study ....................................................................................... 4

1.4.1. General Objective ......................................................................................... 4

1.4.2. Specific Objectives ....................................................................................... 4

1.5. Scope and Limitations ...................................................................................... 4

1.6. Significance of the Research .......................................................................... 5

1.7. Application area ............................................................................................... 5

1.8. Cost Benefit Analysis ....................................................................................... 6

1.9. Organization of the document .......................................................................... 7

CHAPTER TWO ........................................................................................................... 8

2. Literature review .................................................................................................. 8

2.1. Machine vision system .................................................................................... 8

2.2. Digital image processing ............................................................................... 8

2.2.1. Image representation and display ............................................................... 9

2.2.2. Image preprocessing .................................................................................... 11

2.2.3. Image segmentations .................................................................................. 11

2.2.4. Feature Extraction ...................................................................................... 15

2.2.5. Classification .............................................................................................. 16

2.3. Related Work .................................................................................................. 23

2.4. Summary ......................................................................................................... 27
CHAPTER THREE

3. Methodology

3.1. Introduction

3.2. Sampling Techniques

3.3. Sample Collection

3.4. Feature Extraction

3.5. Model selection

3.6. Tools

3.7. Developing a Prototype

3.8. Evaluation Technique

CHAPTER FOUR

4. DESIGN FOR WHITE PEA BEAN QUALITY ASSESSMENT

4.1. Introduction

4.2. The Proposed System Architecture

4.3. Preprocessing

4.4. Segmentation

4.4.1. Segmentation Using Thresholding

4.5. Merging Segmented Images

4.6. Feature Extraction

4.6.1. Color Feature Extraction

4.6.2. Size Feature Extraction

4.6.3. Shape Feature Extraction

4.7. Classification

4.8. Grading of white pea beans

4.9. Summary

CHAPTER FIVE

5. EXPERIMENT

5.1. Introduction

5.2. Data Set

5.3. Implementation

5.4. Test Results

5.4.1. Naïve Bayesian Classifier Test Results

5.4.2. ANN Classifier Test Results
5.5. Discussion ........................................................................................................................................54
5.6. Summary........................................................................................................................................57

CHAPTER SIX .........................................................................................................................................58

6. CONCLUSION AND FUTURE WORK .................................................................................................58

6.1. Conclusion.......................................................................................................................................58
6.2. Contribution to Knowledge ...........................................................................................................59
6.3. Future Work....................................................................................................................................59

References............................................................................................................................................61

APPENDIX A – RGB COLOR SPLITTER AND FALSE REGION REMOVAL .............................65
APPENDIX B – SIZE FEATURES AND GETTING REGION OF INTEREST ..................................67
List of Tables

| Table | Description                                                                 | Page |
|-------|-----------------------------------------------------------------------------|------|
| 1.1   | Development Costs                                                           | 6    |
| 1.2   | Costs for running the system                                                | 6    |
| 1.3   | Cost of manual system                                                       | 6    |
| 2.1   | Summary of related works                                                    | 26   |
| 4.1   | Color feature values for sample of Healthy and RD Kernels                   | 42   |
| 4.2   | Screenshot showing the values of 2 size Features for 12 white pea bean Kernels | 44   |
| 4.3   | Screenshot Showing the Values of 8 Shape Features for 9 Healthy white pea bean Kernels | 45   |
| 5.1   | Data set description                                                        | 50   |
| 5.2   | Test Confusion Matrix of Naïve Bayesian classifier                          | 51   |
| 5.3   | Confusion Matrix Showing Overall Classification Accuracy                     | 53   |
| 5.4   | Performance of model in different classifiers                               | 53   |
| Figure | Description                                                                 | Page |
|--------|-----------------------------------------------------------------------------|------|
| 2.1    | Sampling and Quantization Process                                           | 10   |
| 2.2    | Representation of an MXN Numerical Array                                   | 11   |
| 2.3    | The Sobel convolution masks                                                | 14   |
| 2.4    | Roberts cross convolution mask                                              | 14   |
| 2.5    | Architecture of a Backpropagation Neural Network                           | 21   |
| 3.1    | Design and Methodology                                                      | 29   |
| 4.1    | The Proposed System Architecture                                           | 34   |
| 4.2    | Proposed flowchart of Preprocessing and image segmentation                 | 36   |
| 4.3    | (a) False Regions Shown as Tiny White Spots in the Background (b) white pea bean Kernels after False Region Removal | 37   |
| 4.4    | (A) Original Image (B) Result of Thresholding                              | 38   |
| 4.5    | White pea bean Image and its RGB Components (A) Original RGB Image (B) Binary Image of the Red Component (C) Binary Image of the Green Component (D) Binary Image of the Blue Component | 39   |
| 4.6    | Reconstructed Image of the Red, the Green and the Blue Binary Images        | 39   |
| 4.7    | Color Features Extraction Algorithm                                        | 43   |
| 4.8    | Size Features Extraction Algorithm                                         | 43   |
| 4.9    | Shape Features Extraction Algorithm                                        | 45   |
| 4.10   | Design of ANN Used for the Classification of white pea bean Sample          | 47   |
| 4.11   | Proposed Flowchart of classification and grading of grain                   | 47   |
| 5.1    | Cross_entropy Error Showing the Performance of the Trained ANN             | 52   |
| 5.2    | Spots of Grain                                                              | 55   |
| 5.3    | Comparison of Discriminative Power of Shape and Color features             | 56   |
### List of Abbreviations

| Abbreviation | Description                          |
|--------------|--------------------------------------|
| ANN          | Artificial neural networks           |
| B-P          | Backpropagation                      |
| DIA          | Digital Image Analysis               |
| ECX          | Ethiopian Commodity Exchange         |
| EGTE         | Ethiopian Grain Trade Enterprise     |
| ESA          | Ethiopian Standards Agency           |
| FD           | Fourier descriptor                   |
| F-F          | Feedforward                          |
| HSV          | Hue-saturation-value                 |
| MNN          | Multilayer neural network            |
| MSE          | Mean square error                    |
| PD           | Pest Damage                          |
| RD           | Rotten and Diseased                  |
| RGB          | Red-green-blue                       |
CHAPTER ONE

1. INTRODUCTION

1.1. Background

One of the most important manifestations of economic globalization is the expansion of international trade. While some developing countries have performed well in world markets, many have struggled to become fully integrated in the world trading system. The progressive liberalization of world trade through, for example, successive rounds of General Agreement on Tariffs and Trade (GATT) negotiations and the establishment of the World Trade Organization (WTO), however, has created opportunities for developing countries to access developed country markets more easily. Agricultural and food exports are of particular importance for many developing countries. For example, over 1980-97, agricultural and food products typically accounted for over 25% of total merchandise exports from sub-Saharan Africa. Further, agriculture is of great economic importance, both macro economically and in terms of the livelihoods of the rural population. Agriculture accounts for 61% of employment and 14% of GDP in developing countries, and 85% of employment and 36% of GDP in least developed countries (World Bank, 1999b) [1].

Haricot bean is one of the most important grain legumes grown in the low lands of Ethiopia, particularly in the Rift Valley. In these areas, white pea beans are grown for export purposes as well as for domestic consumption. Haricot bean is also a principal food crop particularly in the southern and eastern parts of Ethiopia [2].

Haricot beans has different varieties such as white, red and mixed colors [2]. The ECX (Ethiopian Commodity Exchange) mostly exported white haricot beans because of need of the market and most profitable agricultural product from other varieties of haricot beans.

White pea bean has become an important export item in the country’s pulse exports. In 2005/06 for instance, Ethiopia exported about 62,262 tons of haricot beans /mainly white pea beans/ valued at about 22 million USD or about 193.7 million ETB, with a unit value of export of 353 USD/mt. The value of export was destined mainly to various countries
such as: Sudan, Yemen, South Africa, UAE, USA, UK, Italy, Germany, Belgium and the Netherlands [2].

Accordingly, white pea beans produced has to go through the intensive care of Ethiopian Commodity Exchange (ECX) to certify that the supplied white pea bean has met the minimum requirement of national standard for domestic and international markets. ECX offers an integrated warehouse system from the receipt of white pea bean on the basis of industry accepted grades and standards for each traded white pea bean by type to the ultimate delivery. Coming white pea bean is placed in warehouses worked by ECX. There are few experts participating in grading of major agricultural products in ECX at Addis Ababa branch. This number of experts are insufficient as compared with the big task.

The increased awareness and sophistication of consumers have created the expectation for improved quality in consumer food products. This in turn has increased the need for enhanced quality monitoring. Quality itself is defined as the sum of all those attributes which can lead to the production of products acceptable to the consumer when they are combined. Quality has been the subject of a large number of studies. The basis of quality assessment is often subjective with attributes such as appearance, smell, texture, and flavour, frequently examined by human inspectors. Consequently, Francis found that human perception could be easily fooled. Together with the high labour costs, inconsistency and variability associated with human inspection accentuates the need for objective measurements systems. Recently automatic inspection systems, mainly based on camera-computer technology have been investigated for the sensory analysis of agricultural and food products. This system known as computer vision has proven to be successful for objective measurement of various agricultural and food products [3].

1.2. Motivation

Technological advancement is gradually finding applications in the agricultural and food industries, in response to one of the greatest challenges i.e. meeting the need of the growing population. Efforts are being geared towards the replacement of human operator with automated systems, as human operations are inconsistent and less efficient. Automation means every action that is needed to control a process at optimum efficiency as controlled
by a system that operates using instructions that have been programmed into it or response to some activities. Automated systems in most cases are faster and more precise [4].

With the progress in computer image vision technology, the gradation technique based on computer vision has developed. The computer vision gradation technology is real-time, objective, nondestructive, and can detect multi-index simultaneously, such as size, defecion, color, shape and the maturity [5].

In Ethiopia, technologies of image analysis or computer vision have not been explored in a significant manner in the development of automation in agricultural and food industries [6]. Specifically, Ethiopian white pea bean quality inspection is based on traditional conducts of grading system. This manual grading evaluation by visual inspection is labor intensive, time consuming and suffers from the problem of inconsistency and inaccuracy in judgement by different perceptions of human. Thus, we should automate grading system of white pea bean agricultural product. Automated white pea bean gradation system plays a significant role to increase the value of produces. Commonly, the gradation indices are shape, size, color, maturity, defection, etc [3].

Automated grading and sorting of agricultural products are getting special interest because of increased demand in different quality food with relative affordable prices by the different group of customers belongs to different living standards [7]. Therefore, the operation of imaging technology in the area will have a great importance to enable profitable activities by increasing efficiency and promoting the market.

1.3. Statement of the Problem

Quality assurance is one of the most important goals of any industry. The ability to manufacture high-quality products consistently is the basis for success in the highly competitive food industry. It encourages loyalty in customers and results in an expanding market share. The quality assurance methods used in the food industry has traditionally involved human visual inspection. Such methods are tedious, laborious, time-consuming and inconsistent [8].
Even white pea bean is the major agricultural product of Ethiopia [2], grading of this product is done using traditional and manual procedures. The time taking checkup operation of grading activity is very expensive, prone to error, less efficient, tedious, biased information for quality control. The manual methods of inspection of the major entities used, including appearance, shape, texture, size and color of white pea bean, exposing the quality valuation to inconsistent results.

The cost acquired for the training of experts is also has significant effect. Due to the fact that we need to replace this manual and traditional way of operation by automating grading system of white pea bean and this will make the operation fast, precise and also cost effective. The automated computer vision of grading system allows to eliminate the errors and biases in the entire processes.

1.4. Objective of the study

1.4.1. General Objective

The main objective of this study is to develop an automated grading system of white pea bean, based on white pea bean morphological and color features using digital image processing techniques and artificial neural network.

1.4.2. Specific Objectives

To achieve the general objective working on the following specific objectives:

- Investigation and analysis of samples for various constituents of white pea bean to identify features
- Select an appropriate methodology or tools to analyze the images of white pea beans
- Adopt and design an algorithm that are used for feature extraction
- Develop the prototype
- Test and evaluate the performance of the prototype implementation

1.5. Scope and Limitations

The purpose of this thesis work is to automate grading system of white pea bean using the approaches of image processing techniques.
Generally, this research work is grounded on the physical property of white pea bean that are characterized as morphological features and color and it does not include moisture content analysis, mass determination and chemical content analysis.

1.6. Significance of the Research

Automated grading has a substantial growth in the field of agricultural and food, in the developed and developing nations. The manual grading of white pea beans replaced by machine vision with the advantages of:

- cost effective
- good quality production
- minimize the needs of experts
- encourage good competent of exporters
- consistent, eradicating personal examining
- higher speed and
- more accurate grading can be achieved.

1.7. Application area

Computer vision systems are being used increasingly in the food industry for quality assurance purposes. The system offers the potential to automate manual grading practices thus standardizing techniques and eliminating tedious human inspection tasks [3].

The ECX (Ethiopian Commodity Exchange) will applied the involvement of the automated system of white pea beans after taking sample of white pea beans in the warehouse system. It will have a number of importance by replacing the manual approach by an automated system of all agricultural and food products.
1.8. Cost Benefit Analysis

✓ Development Costs

| No | Cost Type        | Cost  |
|----|------------------|-------|
| 1  | Transportation   | 1000.00 |
| 2  | Internet Access  | 500.00  |
| 3  | Telephone Fee    | 1000.00 |
|    | Total            | 2,500.00 |

✓ Costs for running the system

| No | Cost type                   | Per year | For 5 years |
|----|-----------------------------|----------|-------------|
| 1  | Salary for 1 operator       | 90,000.00| 450,000.00  |
| 2  | Digital Camera              | 20,000.00| 20,000.00   |
| 3  | Maintenance                 | 10,000.00| 50,000.00   |
|    | Total                       | 120,000.00| 520,000.00  |

✓ Cost benefit analysis

| No | Cost type                   | Per year   | For 5 years |
|----|-----------------------------|------------|-------------|
| 1  | Salary for 5 operators      | 450,000.00 | 2,250,000.00|
|    | Total                       | 450,000.00 | 2,250,000.00|

Based on the above tables we can calculate some mathematical operations:

\[(\text{Total development cost} + \text{Total cost of running the system}) - \text{costs of manual system})\]

\[\left(2,500+520,000\right) - 2,250,000 = -1,727,500\]
So, this result shows us the emergency provider centers are exposed for extra cost. Since they are using manual system. But if they use automated system, can protect loosing of 1,727,500 birr per five years.

1.9. Organization of the document
The rest of this thesis is organized into five chapters. In Chapter Two, literatures reviewed. In Chapter Three, image processing works that are related to cereal grain in general. Chapter Four the design of the proposed solution discussed. In Chapter Five, the experiments used to evaluate the performance of the proposed solution. In Chapter Six conclusions, future work and the contributions of this research work has done.
CHAPTER TWO

2. Literature review

2.1. Machine vision system
Computer vision is meaningful descriptions of physical objects from images [2]. In Timmermans [6], computer vision system contains the capturing, processing and analyzing images to simplify the objective evaluation of visual quality features in agricultural and food products.

The manual inspection of sorting and grading of food and agricultural products are replaced by machine vision systems in recent years [10]. With a wide inspection of different food and agricultural products, including defect detection, grading, sorting, counting, could be conducted with such automated systems. Machine vision incorporates many advantages over the conventional methods of inspection. Ability of being well-matched with other on-line processing tasks, taking dimensional measurements more accurately and consistently than a manual inspection [9].

2.2. Digital image processing
There are different stages of image analysis. The first step towards designing an image analysis system is digital image acquisition using cameras. Sometimes we may receive noisy images that are degraded by some degrading mechanism. One common source of image degradation is the optical lens system in a digital camera that acquires the visual information. If the camera is not appropriately focused, then we get blurred images. In such cases, we need appropriate techniques of refining the images so that the resultant images are of better visual quality, free from aberrations and noises [11].

The next step is the segmentation of objects of interest within the image. Segmentation is the process that subdivides an image into a number of uniformly homogeneous regions. Each homogeneous region is a constituent part or object in the entire scene. In other words, segmentation of an image is defined by a set of regions that are connected and non-overlapping, so that each pixel in a segment of the image acquires a unique region label that indicates the region it belongs to. After perceiving each segment, the next task is to extract a set of meaningful features such as texture, color and shape. These are important measurable entities which give measures of various properties of image segments. Some of the texture properties are coarseness, smoothness, regularity, etc., while the common
shape descriptors are length, aspect ratio, area, perimeter, circularity, etc. Each segmented region in an image may be characterized by a set of these features.

Finally based on the set of these extracted features, each segmented object is classified to one of a set of meaningful classes [11].

2.2.1. Image representation and display
An image is a set of points in a plane, each with its own luminance or color. One can think of any image as consisting of tiny, equal areas, or picture elements, arranged in regular rows and columns. The position of any picture element, or pixel, is determined on a plane. They can be binary (having only two distinct luminance values), grey-value (monochrome) images or color images [11]. Therefore, a digital image can be considered as a discrete representation of data possessing both spatial (layout) and intensity (color) information. An image may be continuous with respect to its spatial, and amplitude domains. To convert it to digital form, we have to sample the function’s coordinates and amplitudes. Digitizing the coordinate values are called sampling. Digitizing the amplitude values is called quantization [12].

To understand sampling and quantization, let us assume that \( f(s, t) \) represents a continuous image function of two continuous variables, \( s \) and \( t \). We convert this function into a digital image. Suppose that we sample the continuous image into a 2-D array, \( f(x,y) \), containing \( M \) rows and \( N \) columns, where \( (x, y) \) are discrete coordinates. For notational clarity and convenience, we use integer values for these discrete coordinates: \( x = 0, 1, 2..., M - 1 \) and \( y = 0, 1, 2..., N - 1 \). Thus, for example, the value of the digital image at the origin is \( f(0, 0) \), and the next coordinate value along the first row is \( f(0, 1) \). It does not mean that these are the values of the physical coordinates when the image was sampled. In general, the value of the image at any coordinate \( (x, y) \) is denoted, as \( f(x, y) \), where \( x \) and \( y \) are integers. The section of the real plane spanned by the coordinates of an image are called the spatial domain, with \( x \) and \( y \) being referred to as spatial variables or spatial coordinates [4, 5]. In general, the transformation process of sampling and quantization is shown in Figure 2.1.
There are two important ways to represent $f(x, y)$. The first way is a plot of the function, with two axes determining spatial location and the third axis being the values of $f$ which are also known as intensities as a function of the two spatial variables $x$ and $y$. However, complex images generally are too detailed and difficult to interpret from such plots. This representation is useful when working with gray-scale sets whose elements are expressed as triplets of the form $(x, y, z)$, where $x$ and $y$ are spatial coordinates and $z$ is the value of $f$ at coordinates $(x, y)$ [4, 6]. The second representation is simply to display the numerical values of $f(x, y)$ as an array called matrix. During development of algorithms, this representation is useful. In equation form, we write the representation of an $M \times N$ numerical array as shown in Figure 2.2 [12]. Both sides of this equation are equivalent ways of expressing a digital image quantitatively. The right side is a matrix of real numbers. Each element of this matrix is called an image element, picture element or pixel [12]. During the representation of digital image, the position of the origin of the $xy$ plane and the directions of the positive $x$ and $y$ axes are important. Accordingly, the origin of a digital image is at the top left. Moreover, the positive $x$-axis extends downward and the positive $y$-axis extends to the right. This is a conventional representation based on the fact that many image displays, for instance TV monitors, sweep an image starting at the top left and moving to the right one row at a time. Hence, the first element of a matrix is by convention
at the top left of the array, so choosing the origin of \( f(x, y) \) at that point makes sense mathematically [12, 13, 14].

\[
\begin{pmatrix}
  f(0, 0) & f(0, 1) & \ldots & f(0, N - 1) \\
  f(1, 0) & f(1, 1) & \ldots & f(1, N - 1) \\
  \vdots & \vdots & \ddots & \vdots \\
  f(M - 1, 0) & f(M - 1, 1) & \ldots & f(M - 1, N - 1)
\end{pmatrix}
\]

Figure 2.2 Representation of an MXN Numerical Array [12]

2.2.2. Image preprocessing

Image preprocessing is the initial processing of the raw image. The images captured are transferred onto a computer and are converted to digital images. Digital images though displayed on the screen as pictures, they are digits which are readable by the computer and are converted to tiny dots or pixel representing the real objects. Image preprocessing is one of the low-level processes in image processing. During image acquisition, we might end up in noisy images. There are many sources of noise in images. During image acquisition, proper focus of the camera is essential. Thus, if the camera is not properly focused then we get blurred images. There are also other causes for the presence of noise in images. Conditions such as foggy environment and relative motion between the object and the camera are causes that can introduce noise into images. Thus if the camera is given a push during the image capturing interval while the object is static, the resulting image will invariably be blurred and noisy. It is easy to notice that noise has negative impact on digital image processing. Digital image processing has ways to solve the problem of noise [11]

2.2.3. Image segmentations

Image segmentation is the process of dividing an image into regions or objects. It is a mid-level step in image processing. It is the first step in the task of image analysis. Segmentation of an image results in the division or separation of the image into regions of similar attribute. Therefore, the basic idea of image segmentation is to group individual pixels
together into regions if they are similar. Similar can mean they are the same intensity (shade of gray), form a texture, line up in a row, and create a shape, etc. Hence, pixels in a region have similarity according to some homogeneity criteria such as color, intensity or texture, so as to locate and identify objects and boundaries in an image [14, 15, 16]. As presented in [18], image segmentation is described as follows. A complete segmentation of an image \( R \) involves identification of a finite set of regions \( (R_1, R_2, R_3, \ldots, R_N) \) such that:

i. \( R = R_1 \cup R_2 \cup \ldots \cup R_N \) — The union of all the sub regions gives the original region

ii. \( R_i \cap R_j = \emptyset \) for \( i \neq j \) — The sub-regions don’t have an intersection

Segmentation algorithms are based on one of the two basic properties of gray-level values. One is based discontinuity of gray-level values; the other is based on the similarity of gray-level values.

In the gray level values discontinuity, we partition an image based on abrupt changes in gray level. The principal areas of interest within this category are the detection of lines and edges in an image. Thus if we can extract the edges in an image and link them, then the region is described by the edge contour that contains it. From this point of view, the connected sets of pixels having more or less the same homogeneous intensity form the regions. Thus the pixels inside the regions describe the region and the process of segmentation involves partitioning the entire scene in a finite number of regions. The second approach is similarity in the gray levels. It is based on the similarity among the pixels within a region. While segmenting an image, various local properties of the pixels are utilized. There are different types of well-established segmentation techniques. Among these, here we will describe histogram-based thresholding and edge detection [11].

### 2.2.3.1. Histogram Based Thresholding

The thresholding operation involves identification of a set of optimal thresholds, based on which the image is partitioned into several meaningful regions [11]. To threshold a grey level image means to compare each pixel value against a fixed number called a threshold. This activity of separating the foreground from the background of images is the first step in image segmentation and is carried out by converting a color image into binary image that has 0 and 255 as the only possible pixel values. The background pixels value could be
0 and the foreground pixel values could be 255 or vice versa. During thresholding, each pixel is examined to determine whether it belongs to the background or the foreground. This is done by comparing each pixel value to a certain constant value known as the threshold. Hence, if pixel value is less than the constant, it is set to 0 or 255 otherwise [14, 16, 19]. Unfortunately, the above explanation works fine for grey scale images as they are one dimensional. However, RGB images are composed of three color bands [12]. These color bands are known as the red, green and blue respectively. It is obvious that converting RGB images into grayscale ones results in the loss of color information which is vital for the identification of the white pea bean kernels. Due to this fact, in addition to the selection of threshold value, thresholding of RGB color images requires the selection of one of these three color bands. Thresholding is a simple but powerful approach for segmenting images having light objects on dark background [20]. Thus, gray level thresholding is based on the analysis of the histograms of an image. The analysis of the histogram depends on the number of its peak values.

2.2.3.2. Edge Detection

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a predecessor step to feature extraction and object segmentation. This process detects outlines of an object and boundaries between objects and the background in the image. An edge-detection filter can also be used to improve the appearance of blurred image. Edge detection is more common for detecting discontinuities in gray level than detecting isolated points and thin lines, as isolated points and thin lines do not occur frequently in most practical images [21]. There are different methods of edge detection techniques including Sobel Operators, Roberts Cross Edge Detector and Canny Edge Detector Technique. The Sobel operator performs a 2-D spatial gradient measurement on an image. Typically, it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. The Sobel edge detector uses a pair of 3x3 convolution masks (figure 2.3), one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows).
The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. It thus highlights regions of high spatial frequency which often correspond to edges. In its most common usage, the input to the operator is a grayscale image, as is the output. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. Figure 2.4 shows Roberts cross convolution mask.

Canny technique is very important method to find edges of an image and the critical value for threshold, after isolating noises from the image, with this noise detection having no adverse effect on the features of the edges in the image [21].

In summary, thresholding is an important part of image segmentation to create binary images. Binary image analysis is useful for image feature extraction. For example, it simplifies the computation of geometrical features of an image. Hence, for this research work, we have used histogram-based image thresholding as it is simple and computationally inexpensive.
2.2.4. Feature Extraction

An image feature is a distinguishing primitive characteristic or attribute of an image. One of the key factors of image analysis is the extraction of sufficient information that leads to a compact description of an examined image. Owing to the immense size of the digital images, it can be very time-consuming if an image is to be analyzed in its original form. To make the process of image analysis simple and less time consuming, some quantitative information is extracted from the objects to be analyzed in the image. By extracting region of interests, the computational cost of object recognition is greatly reduced, thus improving the recognition efficiency [24]. Image features have a major importance in image classification. There are several types of image features that have been proposed for image classification. Morphology, color and texture are some of the basic image features [11, 18, 22].

Morphological features are the geometric property of an image like shape and size. They are physical dimensional measures that characterize the appearance of an object. For instance, area and perimeter are some of the most commonly measured size features and similarly circularity measures the shape of image compactness.

Morphological features are widely used in automated grading, sorting and detection of objects in industry. In certain applications such as classification of cereal grains, these features, alone, are not sufficient for a high-performance inspection process and thus need to be combined with other features. Color and textural features are extracted from the properties of pixels inside the object boundary [24, 25, 26, 28, 29, 30, 31, 32].

In addition to geometrical features, color is one of the most widely used features for image classification. In an image, each pixel records a numeric value that is often the brightness of the corresponding point in the image. Several such values can be combined to represent color information. The most typical range of brightness values is from 0 to 255 (8-bit range), but depending on the type of camera, scanner or other acquisition device a large range of 10 or more bits, up to perhaps 16 (0 to 65,535) may be encountered. However, in most cases these images are still stored with a set of discrete integer grey values because it
is easier to manipulate such arrays and convert them to displays [22]. In line with this, the statistical values of color features like mean, mode, standard deviation, etc. are used for image classification.

There are several features used to measure the morphological, color and shape features of objects under investigation. The most important ones are kernel width (minor axis length), kernel length (major axis length), area, perimeter, color, aspect ratio, ovality, solidity and convexity [33]. Kernel length is defined as the largest distance that exists between the farthest ends of the kernel. Some authors refer to kernel length as major axis length, whereas kernel width of the kernel is the longest line that can be drawn through the object perpendicular to the kernel length. Some works refer to kernel width as minor axis length [19,34,35]. The area of a kernel is defined as the number of pixels contained within its boundary. Kernel area is computed by counting the total number of pixels belonging to the object in the binary image. If we do pixel by pixel walk around the edge of the kernel, we are computing its perimeter. The perimeter of a kernel is the length of its boundary. This parameter is termed as circumference in some works [19, 34, 35]. One of the common shape feature descriptors, aspect ratio, is computed by dividing the major axis length to that of the minor axis.

The work in [37] refers to this parameter as elongation. Likewise, ovality, solidity and convexity are also computed as a ratio of two different measurements. Ovality is the ratio of the area of an object to the area of an ellipse having the same major and minor axis to that of the object. When we take the ratio of the kernel area to the area of the convex hull, we get solidity of the kernel. In this context, convex hull is the smallest convex polygon that can contain the kernel seed region. The other parameter that uses the concept of convex hull is convexity. It is the ratio of the perimeter of the kernel and the perimeter of the convex-hull polygon [19, 34, 35, 36]. Therefore, image features such as morphology, color and texture are used as inputs to a pattern classifier that discriminates objects, white pea bean in our case, into different categories.

2.2.5. Classification
Classification is the process of finding a model that describes and distinguishes data classes or concepts. Such models are called classifiers and their purpose is to predict categorical class labels. Classification is a two-step process, consisting of a learning step and a
classification step. In the learning step, a classification algorithm builds the classifier by analyzing or “learning from” a training set made up of database tuples and their associated class labels whereas in the classification step the model is used to predict class labels for given data [38].

The derived model may be represented in various forms, such as classification rules which are also known as IF THEN rules, decision trees, mathematical formulae, or neural networks. A decision tree is a flowchart-like tree structure. A decision tree has nodes, branches and leaves. Each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions. Decision trees can easily be converted to classification rules. On the other hand, a neural network, when used for classification, is typically a collection of neuronlike processing units with weighted connections between the units [38].

Classification methods are mainly based on two types. They are supervised learning and unsupervised learning [11]. In supervised classification, the classifier is trained with a large set of labeled training pattern samples. The term labeled pattern samples means that the set of patterns whose class memberships are known in advance.

In unsupervised case, the system partitions the entire data set based on some similarity criteria. This results in a set of clusters, where each cluster of patterns belongs to a specific class. In the following sections, we will describe a statistical classifier and neural network classifier.

2.2.5.1. Naïve Bayesian Classification
Bayesian classifier is a statistical classifier. It can predict class membership probabilities such as the probability that a given data item is classified to a particular class. Bayesian classifiers work on the basis of Bayes’ theorem. Bayes theorem is named after Thomas Bayes. There is a simple Bayesian classifier, the naïve Bayesian classifier. Naïve Bayesian classifier is comparable with neural networks and decision trees, in terms of performance. Bayesian classifiers display a high accuracy and speed when the training data is large. They are designed to assign each feature vector to the most probable class [17, 38].
Assume that there are \( N \) classes of patterns \( C_1, C_2, \ldots, C_N \), and an unknown pattern \( x \) in a \( d \)-dimensional feature space \( x = [x_1, x_2, \ldots, x_d] \). Hence the pattern is characterized by \( d \) number of features. The problem of pattern classification is to compute the probability of belongingness of the pattern \( x \) to each class \( C_i \), \( i = 1, 2, \ldots, N \). The pattern is classified to the class \( C_k \) if probability of its belongingness to \( C_k \) is a maximum.

While classifying a pattern based on Bayesian classification, we distinguish two kinds of probabilities. They are priori probability and posteriori probability [11]. The priori probability indicates the probability that the pattern should belong to a class, say \( C_k \), based on the prior belief or evidence or knowledge. This probability is chosen even before making any measurements, i.e., even before selection or extraction of a feature. Sometimes this probability may be modeled using Gaussian distribution, if the previous evidence suggests it. In cases where there exists no prior knowledge about the class membership of the pattern, usually a uniform distribution is used to model it. For example, in a four class problem, we may choose the priori probability as 0.25, assuming that the pattern is equally likely to belong to any of the four classes.

The posteriori probability \( P(C_i|x) \), on the other hand, indicates the final probability of belongingness of the pattern \( x \) to a class \( C_i \). The posteriori probability is computed based on the feature vector of the pattern, class conditional probability density functions \( P(x/C_i) \) for each class \( C_i \) and priori probability \( P(C_i) \) of each class \( C_i \).

Bayesian classification states that the posteriori probability of a pattern belonging to a pattern class \( C_k \) is given by formula

\[
P(C_k|x) = \frac{P(x/C_k)P(C_k)}{\sum_{i=1}^{N} (P(x/C_i)P(C_i))}
\]

The denominator \( \sum_{i=1}^{N} P(x/C_i)P(C_i) \) in the above expression is the scaling term which yields the normalized value of the posteriori probability that the pattern \( x \) belongs to class \( C_i \). Hence, \( x \) belongs to class \( C_p \) when \( P(C_p|x) = \max\{P(C_1|x), P(C_2|x), \ldots, P(C_N|x)\} \).
2.2.5.2. Artificial Neural Network

Artificial neural networks (ANN) are highly distributed interconnections of adaptive nonlinear processing elements. In other words, they are large set of interconnected neurons, which execute in parallel to perform the task of learning. Hence, ANN resembles human brain in two respects. The first property is that knowledge is acquired by the network through a learning process. The other is interneuron connection strengths known as weights are used to store the knowledge, i.e., the weights on the connections encode the knowledge of a network. The neurons are modeled after the biological neurons and hence they are termed as neural networks [11, 24].

One of the key characteristics of a neural network is its ability to learn. A neural network is a complex adaptive system. Consequently, it can change its internal structure based on the information flowing through it. This is achieved through the adjusting of weights. Weight is a number that controls the signal between two neurons and it is associated with each connection. In Figure 2.13, each line represents a connection between two neurons and indicates the pathway for the flow of information. If the network generates a “good” output there is no need to adjust the weights. These values constrain how input data are related to output data. Weight values associated with individual nodes are also known as biases. They are used to reduce the difference between actual and desired output. Weight values are determined by the iterative flow of training data through the network. This means, weight values are established during a training phase in which the network learns how to identify particular classes using their typical input data characteristics [32, 40, 41].

One of the key elements of a neural network is its ability to learn. A neural network is not just a complex system, but a complex adaptive system. Therefore, it can change its internal structure based on the information flowing through it. Typically, this is achieved through the adjusting of weights. A neuron has many continuous valued input signals which represent the activity at the input. In addition to the input, a neuron has an output which represents the response of the neuron to the input signals. The relation between the input and output signals is described by the neuron's activation function. Neural networks are characterized by a lack of explicit representation of knowledge. There are no symbols or
values that directly correspond to classes of interest. Rather, knowledge is implicitly represented in the patterns of interactions between network components [41, 43].

A multilayer neural network (MNN) for learning by backpropagation (B-P) algorithm is an effective system for learning discriminants for classes from a set of examples. Such a network is made up of sets of neurons arranged in several layers. An example of such a multilayer neural network (MNN) for learning by backpropagation (B-P) algorithm is an effective system for learning discriminants for classes from a set of examples. Such a network is made up of sets of neurons arranged in several layers. An example of such a neural network appears in Figure 2.5. The connections between the neurons of adjacent layers relay the output signals from one layer to the next. These layers are named as the input, hidden and output layers. There can be any number of input, hidden and output layers connected in the network. The number of neurons in the input layer equals the dimension of the input vector. This number is equal to the number of features in the input data. The number of neurons in the output layer is determined by the number of the classes under investigation. However, the number of hidden layers and the number of neurons in each hidden layer depend on specific applications. The input layer receives the information and distributes the information to the next processing layer. The hidden and output layers process the incoming signals by amplifying or attenuating or inhibiting the signals through weighting factors. Except for the input layer neurons, the network input to each neuron is the sum of the weighted outputs of the neurons in the previous layer [41, 42, 43].
2.2.5.2.1. The Feedforward B-P Algorithm
Feedforward (F-F) neural networks are the most popular and most widely used models in many practical applications. They are known by many different names, such as multilayer perceptron. The solution to optimizing weights of a MNN is known as B-P. During normal operation, that is when it acts as a classifier, there is no feedback between layers, except during training. This means, all connections proceed from input nodes toward output nodes. This algorithm involves two phases. In the first phase, the inputs are taken in and propagated forward through the network to compute the outputs. This means, the inputs multiplied by the weights are summed and fed forward through the network. The second phase consists of backward pass through the network. In this phase, the difference between the actual output and desired output is computed and compared and an error signal is generated and is passed to each unit in the network and the appropriate weight changes are made. Once the error is reached at a desired rate, the network is said to have a set of weights that produce the correct output for every input. This means, the network stores the class knowledge in its weights and is ready to classify new input data. It helps achieve desired outputs from provided inputs under supervised learning. In a feed forward network
information always moves in one direction and hence, it never goes backwards. A graphical depiction of a typical F-F neural network is given in above Figure 2.5 [40, 41].

Unlike more analytically based information processing methods, neural computation effectively explores the information contained within input data, without further assumptions. Statistical methods are based on assumptions about input data ensembles (i.e. a priori probabilities, probability density functions, etc.). Neural networks, on the other hand build relationships in the input data sets through the iterative presentation of the data and the intrinsic mapping characteristics of neural topologies, normally referred to as learning.

There are two basic phases in neural network operation. They are training or learning phase and testing - recall or retrieval phase. In the learning phase, data is repeatedly presented to the network, while weights are updated to obtain a desired response. In testing phase, the trained network with frozen weights is applied to data that it has never seen.

Instead of sequentially performing a program of instructions, neural networks explore many hypotheses simultaneously using massive parallelism. Neural networks have the potential for solving problems in which some inputs and corresponding output values are known, but the relationship between the inputs and outputs is not well understood or is difficult to translate into a mathematical function [11, 24]. These conditions are commonly found in tasks involving grading and classification of agricultural products [31].

2.2.5.2.2. Why ANN?
We are choosing ANN in which Scenarios that involve complicated or imprecise data are difficult to extract patterns through normal algorithms. This is where the need to have neural networks arises. Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques [40]. ANNs have the potential of solving problems in which some inputs and corresponding output values are known, but the relationship between the inputs and outputs is difficult to translate into a mathematical function [36].
When compared to other methods, ANNs can tolerate noise better and exhibit low classification error rates [32, 45]. Moreover, compared to statistical methods, ANNs using the B-P network could be easily modified to accommodate more features [32].

2.3. Related Work
Advances in hardware and software for digital image processing have motivated several studies on the development of systems, to evaluate the quality of diverse agricultural products. Different grain qualities and varieties of assessment have been reported in the literature that are working in digital image processing. They use grain texture, morphology and color features to achieve their goals. The majority of these studies are focused on the application of computer vision system to agricultural products quality inspection and grading. Computer vision based inspection and grading of apple, oranges, strawberries, nuts, tomato, mushrooms, wheat, corn and rice are examples. These research works use image analysis for automatic information acquisition on quality of grain samples. Generally, based on their objectives research works is aimed at assessing the quality of cereal grain sample and that aim at identification of cereal grain varieties. Under this related work we discuss works that are associated with cereal grain quality assessment in general and white pea beans sample quality assessment in specific and also explains how different our work is from the other related works.

The work in [10], 23 morphological features were used for the discriminant analysis of different cereal grains using machine vision. Classification accuracies of 98, 91, 97, 100 and 91% were recorded for CWRS (Canada Western Red Spring) wheat, CWAD (Canada Western Amber Durum) wheat, barley, oats and rye, respectively. The relationship between color and texture features of wheat samples to scab infection rate was studied using a neural network method.

The work in [29], China use an image analysis technique to identify rice seed varieties was developed recently. A neural network model was used for pattern classification. In recording the images, one rice seed image was taken at a time. In this work, color and morphological features were used as classification parameters. They used MATHLAB 6.5 programming language to extract color and morphological features of individual seeds.
From color features of the mean and variance of RGB components were calculated. Six varieties (ey795, syz3, xs11, xy5968, xy9308, z903) rice seeds, which are widely planted in Zhejiang Province of China, were considered for the research work. The experimentation result indicated that the classification accuracies are 90.00%, 88.00%, 95.00%, 82.00%, 74.00%, 80.00% for ey7954, syz3, xs11, xy5968, xy9308, z903 respectively.

The main aim of the study in [34] is to elaborate complete methodology for the identification of varieties, the level of contamination and other visual features of malting barley with the use of computer science technologies, such as neural image analysis. The work classifies malting barley sample into three classes namely, Beatrix, Sebastian, and Xanadu. To do this, the work models barley using 46 different features composed of geometrical such as area, circumference etc. and non-geometrical such as color features. The work applied neural network for the classification of extracted features. The authors claimed that the optimum model for variety recognition is provided by the color features used to model barley. As a consequence, the work concluded that color features can alone be used to classify malting barley. However, the work does not mention the classification accuracy achieved.

In [35], the development of a digital imaging system and ANN capable of measuring the geometric and shape related parameters for differentiating between rains fed wheat grain cultivars in order to distinguish them. This work used 6 color, 11 morphological and 4 shape features to model wheat. Like most other related works, this work used ANN to classify wheat into 6 cultivars, namely, Sardari, Sardari39, Zardak, Azar 2, ABR1, and Ohadi. The Authors claimed that 86.48% of classification accuracy was achieved.

In [36], the ability of Multi-Layer Perceptron and Neuro-Fuzzy neural networks to classify corn seed varieties based on mixed morphological and color features has been evaluated. Average classification accuracy of corn seed varieties were obtained 94% and 96% by MLP and Neuro-Fuzzy classifiers respectively. However, the work dealt with healthy kernels only and do not address quality factors that describe damaged kernels such as discolored, PD, shriveled, RD and broken.
The work in [46] emphases on the identification of corn kernel shape for the purpose of discriminating between whole and broken kernels of maize. This work, does not address quality factors of maize such as shriveled, discolored and PD. Moreover, the segmentation technique used is based on the green channel of the image. This was one of its shortcomings.

The work in [47] modeled damaged, shriveled and foreign matters found in corn sample. According to the work, damaged kernels are those that are broken or discolored. However, according to ESA, each one of these is separate quality factors. Moreover, the work does not cover maize sample quality factors, namely, discolored, PD and RD kernels of maize. In short, this work does not detect discolored, PD or broken maize kernels. Instead, it simply addresses all these quality factors as damaged.

The work in [48] is about corn kernel damage evaluation. The primary objective of this work was to develop a computer vision system to capture corn kernel images and to classify the images into categories of sound and damaged (germ-damaged and blue-eyed and mold-damaged). This work claimed that about 90% of all damaged corn kernels in the Midwestern U.S. corn market could be classified into either germ–damaged or blue–eye mold–damaged categories. However, the quality factors, namely, shriveled, broken, discolored and PD are not addressed by this work.

The work in [49] is the classification of Ethiopian coffee based on region of growth. This work is based on healthy coffee aimed at discriminating different varieties of Ethiopian coffee using image processing technology. In this work, morphological and color features were extracted from coffee bean images that were taken from six regions of Ethiopia, namely, Bale, Harar, Jima, Limu, Sidamo and Welega. The work tested the classification accuracy of each selected feature set, using Naïve Bayes and neural network classifiers. The experiment was conducted under three scenarios of the features data set such as morphology, color and both morphology and color features.
The work in [50] use a trial investigation of the use of computer vision in sorting fresh strawberries, based on size and shape, showed a result that the developed system was able to sort the 600 strawberries tested with an accuracy of 94-98% into three grades based on shape and five grades on size.

*table 2. 1 summary of related works*

| Authors                              | Title                                                                 | Main work                                                                 | Limitations                                      |
|--------------------------------------|-----------------------------------------------------------------------|--------------------------------------------------------------------------|--------------------------------------------------|
| B.Raba, K.Nowakowski, and P. Boniecki | Visual Quality Evaluation of Malting Barley with use of Neural Image Analysis | The work classifies malting barley sample into three classes namely, Beatrix, Sebastian, and Xanadu. | the work does not mention the classification accuracy achieved. |
| A. Pazoki, F. Farokhi, and Z. Pazoki  | Corn Seed Varieties Based on Mixed Morphological and Color Features Using Artificial Neural Network | classification accuracy of corn seed varieties were obtained 94% and 96% | do not address quality factors                   |
| Habtamu Minassie Aycheh              | Image Analysis for Ethiopian Coffee Classification”, Unpublished Master’s Thesis, | classification of Ethiopian coffee based on region of growth | work is based on healthy coffee beans and also used ImageJ software |
| A. Pazoki and Z. Pazoki              | Classification System for Rain Fed Wheat Grain Cultivars Using Artificial Neural Network | This work used 6 color, 11 morphological and 4 shape features to model wheat…86.48% achieved |                                                  |
| A. Nasirahmadi and N. Behroozi-Khazaei | Identification of bean varieties according to color features using ANN | ANN were applied to identify varieties, based on color features…finally achieved an accuracy of 96% | Only used color features excluding morphological features |
| K. Liao, M. Paulsen, J. Reid, B. Ni, and E. Bonifacio, | Corn Kernel Shape Identification by Machine Vision Using Neural Network Classifier | discriminating between whole and broken kernels of maize. | does not address quality factors and segmentation is based on green channel |

### 2.4. Summary
The experimental results of this research work claimed that morphological features have more discriminative power to classify white pea bean based on growing regions than color features. This fact was shown to be true by using both Naïve Bayes classification and neural network classification. The work claims that the classification accuracy of white pea bean increases when the morphological and color features were used together. This research work, however, has the following shortcomings. The first shortcoming is that color features are extracted including the color of the background of the image. Naturally, however, background color is not part of the color feature of white pea bean beans. Therefore, the color features extracted do not truly represent white pea bean beans. The third shortcoming of this research work is the lack of any proposed algorithm or model to extract (morphological and color) features from white pea beans. All the claimed extractions are done from within GUI of ImageJ software. Because the work is based on healthy coffee beans, it does not recognize broken, shriveled, discolored or PD white pea beans.

It can be concluded from the above researches that morphological structures and color are the most viable features used in computer vision systems for inspection and grading of agricultural products.
In addition to this, neural network is widely in use due to its high performance in the classification accuracy of agricultural products than other classification techniques like statistical classifiers. So, as a classification technique, Artificial Neural Network is the most appropriate technique for classification, inspection and grading of agricultural products. Thus, according to my finding there is no any relevant work that are done for white pea beans. Our research can be considered as a new work specifically for white pea beans and we address what most of the above studies do not address, namely broken, shriveled, discolored, PD and RD quality factors of agricultural products as they are taken as a damaged seed.
CHAPTER THREE

3. Methodology

![Diagram of Methodology Process]

3.1. Introduction

In order to accomplish the objectives of the research and have sufficient knowledge of the study, literatures on contemporary development of image analysis related to cereals quality detection have been reviewed. The review included various kinds of materials including books, previous research works, Internet and articles. From these insights review images analysis techniques and tools that were employed on agricultural products quality identification and disease detection and that were relevant to this work has selected.

3.2. Sampling Techniques

Sampling is one of the main procedures in white pea bean classification and quality assessment. In the current practice of the manual system, the sample drawer draws a ‘representative’ sample of 3kg per 10 tons of a truck, which is an average carrying capacity of a truck, on its arrival. From this 3kg, 125g is used for the analysis and the remaining was used for other references.

In this regard, we have taken 60 images in which the images contain 602 white pea bean kernels. From these samples, 70% were used for training and 30% were used for testing.
and validation purposes. The samples of white pea beans were obtained from the warehouse of Ethiopian Commodity Exchange.

3.3. Sample Collection
We need white pea bean sample to carry out this research work. The sample should contain enough number of representative kernels from each class type. These grains were obtained from coffee and tea quality control & liquoring center at Addis Ababa, a wing of the ECX accomplishing the mentioned grading tasks. Then, images of these grains are taken. A digital camera model cannon Model SD630 with specification of 12.1 mega pixels, used to capture white pea bean kernel images.

The data acquisition system in this research paid due concern with this regard to generate clear, unbiased and simplified digital white pea bean sample database for further analysis and processing. Blue background, with perpendicular and fixed orientation of imaging with the beans suitably spaced for the sake of ease of segmentation activities comprises the major adjustments of the data retrieval phase. The images captured likewise using a digital camera were then transferred into a computer, displayed on a screen and stored on the hard disk in JPEG format as digital color images.

The level and quality of illumination affect digitizing activities using computer vision systems as with the human eye [9]. The performance of the illumination system greatly influences the quality of an image and plays an important role in the overall efficiency and accuracy of the system, underlining the need for manipulation of the illumination system specifications like type, angle and the use of constant light [51]. The aim is to provide the digitizing system with uniform lightning or balanced illumination. Adjustments of the imaging environment with the provision of a suitably uniform light and prohibition of the interference of external lighting sources assisted attainment of a uniform and balanced illumination for capturing the sample white pea bean images.

3.4. Feature Extraction
Automated computer systems for classification, sorting and grading of agricultural products demand the extraction of relevant features that characterize the items under study. This research involved the extraction of morphological and color features from digitized
images of sampled white pea beans to generate a useful input database for quality value classification. Color features of the sample white pea beans were extracted from segmented white pea bean images. Morphological features were extracted from the binary images of the gray scale images of the original white pea bean color images.

3.5. Model selection
Developing the quality value classification demands suitable and applicable selection of models to run, compute and analyze the empirical dataset generated through image processing and analysis approaches. Artificial Neural Network, Naïve Baye’s, and C4.5 classification model were employed to carry out the intended tasks of developing the quality value classifier. The Naïve Bayes classifier is also found an important classification approach that requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification [52]. Neural networks are data driven self-adaptive methods in that they can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model. They are also recognized as universal functional approximates, in that the neural networks can provide a projection of any function with an arbitrary accuracy [53]. In addition, C4.5 classifiers was found to be important for the classification problem. It creates a decision tree based on the attribute values of the available training data in order to classify a new item, by identifying the attribute that discriminates the various instances most clearly. Possibility of higher information gain is raised as a consequence from the feature that tells most about the data instances.

3.6. Tools
MATLAB version R2016b was used to implement the prototype of the system. MATLAB is software developed by MathWorks. Visio 2013 was used for designing the system architecture, algorithms and neural network classifier. Weka 3.6.4 was used to implement the Naïve Bayes and C4.5 classification model. J48 is weka implementation of C4.5. Weka is a machine learning software, written in Java, and developed at the University of Waikato in New Zealand. It is open source software.
3.7. Developing a Prototype
Image of white pea bean first captured using a high-resolution camera, then a pre-processing activity have done on each image of white pea bean by extracting the colour and major feature information of white pea bean and then a prototype have developed for grading of white pea beans is done on the basis of agricultural procedure of Ethiopia Commodity Exchange standards. According to the standard, white pea beans are divided into five classes.

3.8. Evaluation Technique
Each model was evaluated by running a test dataset on the classifier built using the training dataset. The model performance of the classifiers was returned as an output that contains performance matrices and percentage accuracy measures for each class, further summarized into a confusion matrix. Confusion matrix is a kind of a contingency table, used to drive true positives, true negatives, false positives and false negatives indicating the correct/incorrect allotment of samples into their respective classes.
CHAPTER FOUR

4. DESIGN FOR WHITE PEA BEAN QUALITY ASSESSMENT

4.1. Introduction

Having a good design, a great impact on the research work to be successful. Likewise, a successful white pea bean sample quality assessment system should first be designed in a way that can model white pea bean sample contents as accurately as possible. This chapter discusses the design of the system, the proposed solution, and the features to be extracted.

4.2. The Proposed System Architecture

The proposed system architecture for assessing white pea bean sample consists of five components: preprocessing, segmentation, feature extraction, classification and grading. The preprocessing component removes false regions based on size. In the segmentation component, individual white pea bean sample constituents are separated from the background and from each other using a hybrid of color structure tensor and thresholding. Representative features of color, shape, and size are computed in feature extraction component. Finally, the classification component uses ANN to classify constituents of white pea bean sample into one of the seven classes. The system architecture for the proposed system is illustrated in Figure 4.1.
figure 4. 1 The Proposed System Architecture
4.3. Preprocessing
Preprocessing component performs the job of preprocessing the input image. The component does the preliminary task of making the input image ready for the segmentation component. Due to lack of smoothness of the background of the images taken, in a segmented image there could be some groups of pixels having the foreground grey level value while being enclosed in the background. Naturally, these regions belong to the background but they appear as foreground objects. In this work, we call these as false regions. False regions are removed in this component.
In order to do so, we have to set their grey level value back to that of the background. We can determine if a region is false by computing its size. To determine whether a region is false or not, we analyzed foreground objects and found out that false regions are those group of pixels whose size are less than 500 pixels. The algorithm for preprocessing relies on the saturation and hue images. False region is identified by computing the area of each region and setting the values of the pixels in those regions to the background grey value. The preprocessing flowchart is shown in figure 4.2.

4.4. Segmentation
The segmentation component of our proposed architecture is responsible for carrying out the work of separating white pea bean sample constituents from each other and from the background of the image. This component contains thresholding and the thresholding sub-component extracts information from each of the three binary images to form an intermediate image called reconstructed image. The final job of the segmentation component is carried out by the merger sub-component. The merger sub-component combines outputs from the structure tensor segmentation and the thresholding sub-components and makes a new binary image known as merged image. A merged image contains all the information required to extract the features which are used for classification. The proposed segmentation algorithm is thresholding algorithms. It is based on the results thresholding segmentation techniques. The proposed flowchart segmentation is shown in figure 4.2.
figure 4.2. Proposed flowchart of Preprocessing and image segmentation
4.4.1. Segmentation Using Thresholding

As described in Chapter Two, thresholding is one of the available segmentation techniques. It works based on grey-level images. The result of thresholding is a black and white binary image. However, since white pea bean grains are whitish in color, parts of the white pea bean images will resemble the background when converted into binary image. This is in contrast with a successful segmentation of images which results in full separation of background of the image from the foreground without information loss. This information loss negatively impacts the white pea bean grain recognition process because the color, shape and size information of each grain would be lost to the background. The main effect of this phenomenon would be manifested during shape and size recognition. Since some of white pea bean grains information are lost to the background, the healthy grains appear to be broken and assume different shape, area and size values than the ir original nature. Moreover, segmentation using the technique of thresholding is sensitive to shading and specularity. Hence feature modeling and extraction will not be accurate and effective.
Therefore, in order to remove such problems, we should take care of at image acquisition phase of white pea beans also good lightning effect could change the effect of shadowing.

![Figure 4.4](image)

**Figure 4.4** (A) Original Image (B) Result of Thresholding

### 4.5. Merging Segmented Images

An RGB image consists of different information in its three constituent grey scale component images. Some of the information that makes up the RGB image could be found in one of the component images and it may be missing in the rest. For instance, based on empirical investigation, we have learned that special region information for discoloration is found in the blue and green component binary images and it is missing in the red component binary image. Therefore, it is necessary to bring the information that is found in RGB component images together and form a reconstructed image. However, this reconstructed image does not have complete boundary information of the white pea bean kernels. Therefore, it is necessary to include kernel boundary information into the reconstructed image. But, this information is always available in the color structure tensor segmented image. The reconstructed image of the images in Figure 4.6(b), Figure 4.6(c) and Figure 4.6(d) is shown in Figure 4.6.
figure 4. 5. White pea bean Image and its RGB Components (A) Original RGB Image (B) Binary Image of the Red Component (C) Binary Image of the Green Component (D) Binary Image of the Blue Component

figure 4. 6 Reconstructed Image of the Red, the Green and the Blue Binary Images
4.6. Feature Extraction
Feature extraction component is responsible for extracting the descriptive features of the white pea bean sample constituents. This component contains the color features extraction, the shape features extraction and the size features extraction sub-components. As stated in Chapter One, one of the objectives of this research work is to identify seven kinds of white pea bean sample constituents based on their color images by using digital image processing (DIP). This task depends on analysis of quantitative data extracted from images. However, processing of all quantitative data of white pea bean sample is computationally inefficient. Thus, representative features of white pea bean sample constituents are selected and extracted. The features extracted from white pea bean images can be grouped into three categories, namely, color, size, and shape. To describe white pea bean sample constituents, we identified 14 color features, 2 size features, and 8 shape features.

4.6.1. Color Feature Extraction
Color is one of the visual attributes of white pea bean kernels. It helps distinguish among certain classes of white pea bean sample. For instance, the categories of RD and discolored white pea bean kernels can be satisfactorily distinguished by using their color features. Moreover, color has the ability to isolate foreign matters from among white pea bean kernels. Fourteen color features have been selected to represent the color features of white pea bean sample constituents. The first set of color features, extracted are based on the RGB color model. These features are the mean values of the red, green, and blue components of each image as computed from its three color channel functions using Equation (16) (17) (18). In this equation, the pixel value functions \( r(x, y) \), \( g(x, y) \), and \( b(x, y) \) are the respective channels of the RGB color model [19].

\[
R = \frac{1}{N} \sum_{k=1}^{N} r(x, y) \tag{16}
\]

\[
G = \frac{1}{N} \sum_{k=1}^{N} g(x, y) \tag{17}
\]

\[
B = \frac{1}{N} \sum_{k=1}^{N} b(x, y) \tag{18}
\]

where \( x, y, k \), and \( N \) are positive integers.
In addition to these, we identified three additional color features, namely, *spot-red, spot-green* and *spot-blue* based on the RGB model that correspond to the damage areas within a white pea bean kernel. These features are calculated using Equation (16) (17) (18) based on the area of the damaged region within a white pea bean kernel. These features are discussed in Chapter Five. The screenshot of the numerical values of the 14 color features corresponding to 12 healthy white pea bean kernels is shown in Table 4.3. In this table, the columns titled SpotR, SpotG and SpotB represent the spot-red, spot-green and spot-blue features respectively.

The second set of color features measured in this work is based on the HSV color model. As described in Chapter Two, the HSV color model is the other most commonly used color model. In this model, color is described by three components: hue, saturation and value (intensity). The colors of the RGB space are usually not easy for humans to interpret. However, the hue, saturation and value space, HSV color space is, by contrast, intuitive. Hue is an attribute associated with the dominant pure color such as pure blue, pure red, etc. Saturation is the amount of white light that is mixed with a hue while intensity (value) is defined as a measure of the brightness of light. In the HSV color space, the intensity attribute is decoupled from the color information. Moreover, the hue (H) and saturation (S) attributes are closely related to the way human beings perceive color. The H, S, and V attributes can be derived from the RGB model components as discussed in Chapter Two. In this research work, the color features are extracted using the mean values of each component of the HSI model and are calculated for each foreground region by using Equation (16), (17) and (18). In this equation, the functions h(x, y), s(x,y), and v(x, y) are the respective channels of the HSV color model [19].
In addition to these, we identified three additional color features, namely, spot-hue, spot-saturation and spot-value, based on the RGB model that correspond to the damage areas within a white pea bean kernel. These features are calculated using Equations (16), (17), and (18) based on the area of the damaged region within a white pea bean kernel. These six spot features of white pea bean kernels are very helpful in the process of distinguishing healthy and damaged kernel. The values of these features for 9 healthy and 9 RD kernels are given in Table 4.1.

**Table 4.1 color feature values for sample of Healthy and RD Kernels**

|         | Healthy          | Rotten and Diseased |
|---------|------------------|---------------------|
| Spot-Hue| 1.28             | 1.31                |
| Spot-Saturation | 0.58          | 0.59                |
| Spot-Value | 0.45               | 0.48                |
| Spot-Red  | 0.68             | 0.69                |
| Spot-Green| 0.03              | 0.00                |
| Spot-Blue | 0.57              | 0.51                |
| Spot-Hue  | 1.15             | 1.18                |
| Spot-Saturation | 0.37          | 0.38                |
| Spot-Value | 0.31              | 0.36                |
| Spot-Red  | 0.61             | 0.62                |
| Spot-Green| 0.38              | 0.37                |
| Spot-Blue | 0.64              | 0.75                |
| Spot-Hue  | 1.39             | 1.20                |
| Spot-Saturation | 0.49          | 0.50                |
| Spot-Value | 0.40               | 0.41                |
| Spot-Red  | 0.64             | 0.62                |
| Spot-Green| 0.21              | 0.19                |
| Spot-Blue | 0.60              | 0.50                |
| Spot-Hue  | 1.24             | 1.20                |
| Spot-Saturation | 0.53          | 0.51                |
| Spot-Value | 0.44              | 0.40                |
| Spot-Red  | 0.62             | 0.49                |
| Spot-Green| 0.12              | 0.34                |
| Spot-Blue | 0.29              | 0.35                |
| Spot-Hue  | 1.23             | 1.30                |
| Spot-Saturation | 0.58          | 0.55                |
| Spot-Value | 0.45               | 0.43                |
| Spot-Red  | 0.62             | 0.65                |
| Spot-Green| 0.02              | 0.09                |
| Spot-Blue | 0.30              | 0.50                |
| Spot-Hue  | 1.22             | 1.30                |
| Spot-Saturation | 0.59          | 0.55                |
| Spot-Value | 0.46               | 0.44                |
| Spot-Red  | 0.62             | 0.64                |
| Spot-Green| 0.01              | 0.02                |
| Spot-Blue | 0.22              | 0.39                |
| Spot-Hue  | 1.14             | 1.16                |
| Spot-Saturation | 0.39          | 0.35                |
| Spot-Value | 0.36              | 0.29                |
| Spot-Red  | 0.58             | 0.61                |
| Spot-Green| 0.41              | 0.43                |
| Spot-Blue | 0.48              | 0.75                |
The spot-hue, spot-saturation and spot-value, features are calculated using Equations (19), (20), and (21) respectively. These features are discussed in Chapter Five. The screenshot of the numerical values of all the color features corresponding to 12 healthy white pea bean kernels is shown in Table 4.3. In this table, the columns titled SpotH, SpotS and SpotV represent spot-hue, spot-saturation and spot-value features respectively. The proposed algorithm for color features extraction is shown in figure 4.8.

4.6.2. Size Feature Extraction

This research work uses two size features, namely, area and perimeter features to determine the size of white pea bean sample constituents. As discussed in Chapter Two, area is the total number of pixels corresponding to a single kernel. Likewise, perimeter is the total number of pixels around a kernel region. In addition to standing by itself as one feature, area serves to derive all the color features as shown in Equations (16) and. All the fourteen color feature values are determined by dividing the sum of their particular color values by the area. The proposed algorithm for the size features extraction is shown in Figure 4.9.
Sample data of area and perimeter features extracted from 9 discolored white pea bean kernels are shown in Table 4.2.

\textit{Table 4.2 Screenshot showing the values of 2 size Features for 12 white pea bean Kernels}

|    | Area     | Perimeter |
|----|----------|-----------|
| 1  | 0.54127  | 94.3973   |
| 2  | 0.47304  | 163.0217  |
| 3  | 0.48349  | 92.6878   |
| 4  | 0.40779  | 82.2472   |
| 5  | 0.4271   | 82.1643   |
| 6  | 0.28441  | 99.6808   |
| 7  | 0.47051  | 93.3484   |
| 8  | 0.32512  | 129.3741  |
| 9  | 0.48754  | 90.7134   |

\textbf{4.6.3. Shape Feature Extraction}

Shape descriptors are numbers that are computed from a two dimensional shape. The shape descriptors can thus be considered as an approximate description of the shape. Shapes can be examined for their similarity by way of using their respective shape descriptors. Hence, shape similarity somehow corresponds to similarity of the shape descriptors. The shape of white pea bean kernel is believed to be good for the discrimination between foreign matters and white pea bean grains. Moreover, it is also a helpful tool in the separation process of broken and whole white pea bean kernels. The \textit{eight} shape descriptors identified in this research work are \textit{count of convex hull sides}, \textit{aspect ratio}, \textit{ovality}, \textit{triangularity}, \textit{convexity}, \textit{solidity}, \textit{major axis to area ratio} and \textit{Fourier descriptor}. Count of convex hull polygon sides is the number of sides of convex hull polygon. Triangularity is defined as the area of a triangle with a base equal to the minor axis of the kernel and height equal to the major axis. Major axis to area ratio is the ratio of the major axis to the area of white pea bean kernel. The rest of the shape descriptors are explained in Chapter Two. Sample data of these 8 shape features is shown in Table 4.3. The eighth shape descriptor is the Fourier Descriptor (FD). We used FDs to describe the contour of an object. First, we compute a set of FDs for a healthy white pea bean kernel. Then, we compute the FDs of an unknown
object and compare it to the known white pea bean kernel by ignoring the first component of the descriptors. The known object, whose FDs are the most similar to the unknown object’s FDs, is the object the unknown object is classified to.

*Table 4.3 Screenshot Showing the Values of 8 Shape Features for 9 Healthy White Pea Bean Kernels*

| ConvexHull | AxisRatio | Ovality | Triangularity | Convexity | Solidity | MajorAxis/area | Fourier |
|------------|-----------|---------|--------------|-----------|----------|----------------|---------|
| 1          | 0.8340    | 375.2500| 0.6415       | 0.0116    | 0.9786   | 0.0122         | 0.0906  |
| 2          | 0.7474    | 370.3710| 0.9946       | 0.6401    | 0.0169   | 0.9834         | 0.0130  |
| 3          | 0.7108    | 364.9950| 0.9776       | 0.5112    | 0.0148   | 0.9654         | 0.0138  |
| 4          | 0.6748    | 401.6500| 0.9554       | 0.5594    | 0.0138   | 0.9649         | 0.0136  |
| 5          | 0.6592    | 341.9450| 0.9816       | 0.6486    | 0.0138   | 0.9664         | 0.0154  |
| 6          | 0.6666    | 401.6060| 0.9547       | 0.5490    | 0.0138   | 0.9664         | 0.0109  |
| 7          | 0.7311    | 348.0320| 0.9882       | 0.6442    | 0.0172   | 0.9631         | 0.0142  |
| 8          | 0.8430    | 376.0280| 0.9904       | 0.6428    | 0.0213   | 0.9792         | 0.0120  |
| 9          | 0.7987    | 377.2020| 0.9838       | 0.6471    | 0.0145   | 0.9657         | 0.0124  |

Fourier descriptors are invariant to scaling, translation and rotation. These properties make Fourier descriptors suitable to compare object shapes having a range of different sizes and orientation. Sample data showing Fourier descriptors of 9 healthy white pea bean kernels is shown in Table 4.6. Besides, sample of a healthy white pea bean kernel, its original contour and resampled contour of 128 points is shown in Figure 4.9. The proposed algorithm for shape features extraction is shown in Figure 4.10.
4.7. Classification

Classification component contains ANN classifier and the class count sub-components. The ANN classifies white pea bean sample into seven classes. The class count sub-component is responsible for counting the number of sample constituents belonging to each class. Although there are other methods like mathematical functions, rule-based algorithm or statistical methods available for classification, we chose ANNs over others. There are several reasons for choosing neural networks over other methods for the purpose of this research work. The classification of grain kernels cannot be easy using unique mathematical functions. This is due to the variation in morphology, color and texture of the grain kernels under consideration. ANNs have the potential of solving problems in which some inputs and corresponding output values are known, but the relationship between the inputs and outputs is difficult to translate into a mathematical function. When compared to other methods, ANNs can tolerate noise better and exhibit low classification error rates. Moreover, compared to statistical methods, ANNs using the BP network could be easily modified to accommodate more features. To add empirical experience to the above claims, we trained naïve Bayesian classifier and ANN classifier on the same training data set. We compared their performance based on classification accuracy and we found out that ANN performs better than the naïve Bayesian classifier. The neural network architecture in this work is a three-layered F-F network with sigmoid hidden and softmax output neurons. Such network can classify vectors arbitrarily well, given enough neurons in its hidden layer. The input layer contains 24 neurons corresponding to each 24 inputs and the output layer consists of 7 neurons corresponding to each 7 output classes. Softmax is a neural transfer function. Transfer functions calculate a layer's output from its net input. The network is designed to have only one hidden layer consisting of 45 nodes. The hidden layer of the neural network is composed of 45 neurons. This number of neurons in the hidden layer is selected empirically based on the performance it exhibited over smaller and larger number of neurons. Moreover, the decision to use only 1 hidden layer is made based on facts found in the literature. There is no reason to use any more than one hidden layer. The network is designed to use B-P algorithm training. To measure the performance of the network during
training phase, we preferred to use cross-entropy error function over mean square error (MSE). Compared to MSE, cross entropy function is proven to accelerate the backpropagation algorithm and to provide good overall network performance. The architectural design of this ANN is depicted in Figure 4.11. The proposed classification flowchart is shown in Figure 4.12.

![Design of ANN Used for the Classification of white pea bean Sample](image)

*figure 4. 10 Design of ANN Used for the Classification of white pea bean Sample*

![Proposed Flowchart of classification and grading of grain](image)

*figure 4. 11 Proposed Flowchart of classification and grading of grain*
4.8. Grading of white pea beans
The grading of white pea beans is done on the basis of agricultural procedure of Ethiopia Commodity Exchange standards. According to the standard white pea beans are divided into five classes (Grade 1 to Grade 5).

4.9. Summary
We proposed a system architecture that consists of four components, namely, preprocessing, segmentation, feature extraction and classification. The preprocessing component does the job of removing noise and false regions. The output of the preprocessing component fed into the segmentation component. The segmentation component contains our novel segmentation technique that combines color structure and thresholding segmentation techniques. The third component, feature extraction, performs feature extraction on the output of the segmentation component. This component extracts a total of 24 (14 color, 8 shape and 2 size) features that are identified for the purpose of modeling the different characteristics of white pea bean sample constituents. The fourth component, classification, classifies white pea bean data based on the features extracted by the feature extraction component. This component consists of a neural network classifier consisting of 24 input nodes and 7 output nodes corresponding to the number of inputs features and the number of output classes.
5. EXPERIMENT

5.1. Introduction
In this chapter, we report a set of experimental results carried out to test the effectiveness of our proposed system. Accordingly, the type of classifier, the data set used and the results achieved in the classification process have discussed. Alongside these, the discriminative power of color, size, and shape are tested and compared.

5.2. Data Set
A total of 602 of White pea bean kernels and foreign matter are prepared to train, validate and test the proposed model. These 602 White pea bean sample constituents are separated into their corresponding 7 classes based on their characteristics. Therefore, we finally have 7 outputs each corresponding to each of the 7 classes. The data were partitioned randomly into training, validation and test sets. Image acquisition is done using cannon Model SD630 with specification of 12.1 mega pixels. The images taken are all 24-bit color JPEG format. The number of White pea bean kernels per image is different for the different classes as shown in Table 5.1. During image acquisition, the camera is mounted on a stand which provides easy vertical movement. The distance between the camera and the sample was fixed at 14 cm to maintain the same vertical distance on each image taken. During background color selection, we compared a red, blue black, and light blue colors. We observed that the light blue color makes a good contrast with the foreground objects and achieved better segmentation result. Consequently, for each image, a blue background is used. The samples of White pea bean are placed directly under the camera for image acquisition. For neural network training, 70% of the data is used. The rest of the data is used for validation and testing each consisting of 15% of the input data. The training set is presented to the network during training. The training set is used to fine tune the weights of the network. Whereas, the validation set are used to measure network’s generalization ability, and to halt training when generalization stops improving. The testing data have no effect on training and so provide an independent measure of network performance during and after training. Similarly, for naïve Bayesian classification, 70% of the data is used for training and the rest 30% is used for testing. As this is supervised effort, the training data needs to be labeled. The labels are presented to the neural network as binary code. Since
there are 7 classes into which the White pea bean sample constituents are to be classified, the corresponding number of bits in the binary code is also set to seven. These classes and the number of images used for each in the training process are shown in Table 5.1.

**Table 5.1 Data set description**

| Target Class Description | Binary Code (Class labels) | Number of Kernels |
|--------------------------|---------------------------|-------------------|
| Foreign matter           | 0000001                   | 59                |
| RD                       | 0000010                   | 89                |
| Healthy                  | 0000100                   | 157               |
| Broken                   | 0001000                   | 35                |
| Discolored               | 0010000                   | 142               |
| Shriveled                | 0100000                   | 34                |
| PD                       | 1000000                   | 86                |
| **Total**                |                           | **602**           |

5.3. Implementation
MATLAB version R2016b tool is used to develop the prototype of the system. Moreover, the specification of the computer on which the system is implemented is Intel Core i7 laptop computer with 8GB RAM and 2.3 GHz processor.

5.4. Test Results
Tests are conducted on naïve Bayesian, C 4.5 and ANN classifiers to determine the best performing classifier based on the criterion of classification accuracy.

5.4.1. Naïve Bayesian Classifier Test Results
The performance of the naïve Bayesian classifier was tested with 181 (30% of the training data) data items. The test confusion matrix of the trained naïve Bayesian classifier is depicted in Table 5.2. The diagonal elements show instances that were correctly classified. For this classifier, the classification accuracy of RD, shriveled, broken, discolored, healthy, foreign, and PD are 54.1%, 100%, 93.5%, 100%, 51.5%, 100%, and 58.6% respectively.
The overall classification accuracy obtained is 76.79%. This is calculated by summing the number of correctly classified kernels in each class and dividing the result by the total number of test data (181). The classification accuracy for the classes RD, healthy, and PD is below 70%. This has affected the overall performance of the naïve Bayesian classifier to significantly underperform, compared to the neural network classifier.

### 5.4.2. ANN Classifier Test Results

After the data was partitioned as explained in Section 5.2, the neural network is trained. The whole process, i.e., training, validation and testing took only 2 seconds. During training, cross-entropy was used as the error function. The neural network training process is halted at the 62th iteration (epoch) at which the validation error started to rise and the training error was dropping. This training process is shown Figure 5.1.

| Output Class | RD | Shriveled | Broken | Discolored | Healthy | Foreign | PD |
|--------------|----|-----------|--------|------------|---------|---------|----|
| RD           | 13 | 0         | 2      | 0          | 3       | 0       | 10 |
| Shrived      | 4  | 34        | 0      | 0          | 5       | 0       | 0  |
| Broken       | 0  | 0         | 29     | 0          | 0       | 0       | 0  |
| Discolored   | 2  | 0         | 0      | 16         | 3       | 0       | 0  |
| Healthy      | 2  | 0         | 0      | 0          | 17      | 0       | 3  |
| Foreign      | 0  | 0         | 0      | 0          | 9       | 0       | 0  |
| PD           | 3  | 0         | 0      | 0          | 4       | 0       | 17 |
| **Classification Accuracy** | **54.16%** | **100%** | **93.5%** | **100%** | **51.5%** | **100%** | **58.6%** |

*Table 5.2 Test Confusion Matrix of Naïve Bayesian classifier*
Accordingly, classification accuracies of 96.7%, 97.8%, and 96.7% have been achieved for training, validation and testing respectively. Moreover, an overall classification accuracy of 96.8% is achieved. This accuracy is calculated by dividing the total number of correctly classified kernels by 602 (by the total number of kernels in the sample). The confusion matrix showing the overall classification results (including training, validation and testing) is shown in Table 5.3.
Since the naïve Bayesian classifier resulted in 76.79% of classification accuracy and the ANN achieved 96.8% for the same, we conclude that ANN outperforms naïve Bayesian classifier.

### Table 5.4 Performance of Model in Different Classifiers

| Model (% correctly classified) | Morphology Feature | Color Feature | Combined Feature Performance |
|-------------------------------|--------------------|---------------|-----------------------------|
| C 4.5                         | 73.91%             | 64.27%        | 82.09%                      |
| ANN                           | 82.8%              | 75.6%         | 96.8%                       |
| Naïve Bayes                   | 74.01%             | 66.79%        | 76.79%                      |

### 5.4.2.1 Scenario One

PD (Pest damaged), discolored, and RD (rotten and diseased) areas in kernels are modeled using the area occupied by the damage and the corresponding hue, saturation, value (intensity), red, green and blue color values of the areas. In this work, the hue, saturation, value, red, green and blue values that are associated with damaged areas within a kernel are termed as spot-hue, spot-saturation, spot-value, spot-red, spot-green and spot-blue. In this scenario the effectiveness of the features spot-hue, spot-saturation, spot-value, spot-
red, spot-green and spot-blue attributes are studied. Accordingly, we retrained the classifier without the inclusion of these attribute in the training data. As a result, we observed that the discriminative power of spot-hue, spot-saturation and spot-value, spot-red, spot-green and spot-blue attributes are so high that without these features, the classification accuracy of the ANN classifier drops significantly. The overall classification accuracy of the ANN classifier dropped to 82.8%.

5.4.2.2. Scenario Two
In this scenario, the discriminative power of the size feature is examined. We experimented to see the effect of area by training the neural network excluding area attribute from the feature data set and training the neural network. Originally, we incorporated area as a feature to model the size of White pea bean kernels with the intention to discriminate between shriveled and other kernels. Therefore, in this scenario, we expected shiveled kernels to be misclassified into other classes. However, the accuracy of the classifier was observed to reduce from 96.8% to 94.4%.

5.4.2.3. Scenario Three
In this scenario we examined the discriminative power of shape features. In light of this, we trained an ANN without the inclusion of these features in the training data set. We found out that the overall classification accuracy of the classifier was reduced from 96.8% to 93.4%.

5.4.2.4. Scenario Four
In this scenario, the discriminative power of color features is examined. This is done by training the ANN without the inclusion of color features in the training set. In doing so, we found out that the ANN classification accuracy dropped from 96.8% to 65.5%.

5.4.2.5. Comparison with Manual work
Finally, the system’s performance is compared against the manual counterpart based on the time taken and efficiency to do the same job by an expert from the EGTE. The expert has taken 4 to 8 minutes to identify and count foreign, rotten and diseased, healthy, broken, discolored, shveled and pest damaged kernels from a mixture. However, our proposed system completed the job within 45 seconds.

5.5. Discussion
In Scenario one, no reduction in classification accuracy is observed for the class foreign matters and the class broken. The significant drop in classification accuracy of the classes
discolored, RD and PD classes is due to the fact that all these kernels have spots on their surface as shown in Figure 5.2.

The spots on PD kernels indicate that the kernels are eaten up by the pest on that location. The spots on the discolored and RD kernels indicate that they have been discolored or rotten or damaged at those locations. PD, discolored and RD areas within a kernel are different from the rest of the kernel area due to their unique color characteristics. PD kernels have unique color at the spot location. Similarly, we have learned that discolored and RD kernels have unique color characteristics at their discoloration and rottenness spots. The proposed Flowchart shown in figure 4.2, distinguishes these three kernel types (classes) based on their color and area characteristics of their respective spots. The proposed algorithm is found to be effective for this purpose as shown by the high percentage of classification accuracy shown in the confusion matrix shown in Table 5.3.

In scenario two, all shriveled kernels (97.0%) are classified correctly without including the attribute area from the feature set is that even though it is excluded, it is still present in composite features such as average red, green, blue, hue, saturation, intensity values. This is because composite features are calculated by summing the respective color value of a kernel and dividing it by the kernel area. Hence, the effect of area will not be ruled out by its exclusion from the feature set. Thus, we can conclude that, color features have the
highest discriminative power and size feature has the second highest discriminative power. However, we found out that shape features have the least discriminating power for assessment of White pea bean sample. The comparison of discriminative power of size, color and shape features is presented in Figure 5.3 as column chart. In this chart, the discriminative power of size, color and shape features is compared by using the observed drop in the classification accuracy during scenario 2, scenario 3 and scenario 4.

![Comparison of Discriminative Power of Features](image)

**figure 5.3 Comparison of Discriminative Power of Shape and Color features**

One of the challenges of this work is the lack of proper laboratory settings for image acquisition. In addition to this, the quality of the camera, the image acquisition environment and other imaging factors may affect the result. Moreover, the number of grains in the collected White pea bean sample for some classes like PD is very small. This has its own effect in the achieved result. The other major issue is that some kernels exhibit the properties of more than one class which results in misclassification. The final issue of this work is the lack of data for the classification of filth. Hence, we excluded this class from the research.
5.6. Summary
We have shown that our proposed segmentation algorithm and the models we used to represent features of White pea bean sample fulfilled their intended purposes. This is shown by a high level of classification accuracy we achieved. Moreover, it has been shown that the discriminative power of color features is significantly greater than that of size and shape features.
CHAPTER SIX

6. CONCLUSION AND FUTURE WORK

6.1. Conclusion
White pea bean is used as a major food item around the world and especially in sub-Saharan Africa. Countries, including Ethiopia, produce White pea bean both for domestic and export consumptions. In industrialized countries, White pea bean is largely used as livestock feed and as a raw material for industrial products. Besides, White pea bean is used as input to factories that produce processed food products. White pea bean grains may be damaged during harvesting, storing, and transportation. Some of the damage types merely reduce the quality of the grain while others make it unsafe to eat. Because of this, governments impose a standard on White pea bean destined either to the inland or overseas market to assure its quality. These standard sets criteria by which White pea bean quality is evaluated. The standard is based on morphological and chemical characteristics of White pea bean.

Currently, there is no automated technique that can assess White pea bean quality. Rather, White pea bean quality is assessed manually. However, manual evaluation takes significant amount of time and requires trained and experienced people. This is especially evident during large scale inspection. Naturally, this manual process of quality assessment is prone to bias and inconsistencies. In order to eliminate most of the shortcomings of the manual work, it is important to employ automated quality assessment system. Automated White pea bean quality assessment has many important advantages over the manual technique. The major advantage of automated White pea bean quality assessment is its objective nature. This helps to describe visible attributes accurately, without bias and inconsistencies. Compared to the manual counterpart, automated systems take lesser time and effort. Therefore, in this research work, automatic White pea bean quality assessment system is developed to classify White pea bean sample consistently and objectively. For this, best segmentation algorithm is developed to identify the damaged areas of White pea bean kernels. A total of 24 features are identified to model the constituents of White pea bean sample. Moreover, system architecture is designed that works based on the proposed segmentation algorithm. For classification purpose, a feedforward artificial neural network
with 24 input nodes and 7 output nodes and backpropagation algorithm, corresponding to the number of input features and output classes respectively is designed. The network’s performance is compared against other existing classifiers both empirically and based on supporting facts from the literature.

Results show that the overall success rate for the classification of White pea bean sample is 96.8%. The success rates for detecting foreign, rotten and diseased, healthy, broken, discolored, shriveled and pest damaged kernels are 94.9%, 96.5%, 96.3%, 97%, 97.9%, 97%, and 97.6%, respectively. Moreover, these results show that, the proposed segmentation algorithm and system architecture are effective in assessing the quality of White pea bean sample constituents according to the standard set for White pea bean sample by ESA. Hence, it is feasible to assess the quality of White pea bean sample using digital image processing and ANN. Therefore, it is practically possible to void the negative aspects of the manual work.

6.2. Contribution to Knowledge
This research work has contributed the following to the area of digital image processing in relation to automatic White pea bean grain quality assessment. First, we proposed Best segmentation algorithm that is appropriate for the segmentation of White pea bean sample images in particular. The algorithm could potentially be extended for other grains as well. Second, we identified a total of 24 features that are used to successfully classify White pea bean samples. Third, we designed a neural network classifier that can successfully label White pea bean sample constituents. Finally, we proposed system architecture for the automatic assessment of the quality of White pea bean sample.

6.3. Future Work
Though this study has been able to assess the White pea bean sample quality successfully, few works still remain unsolved. The following are the possible future works.

- In this study, White pea bean sample images are captured using a single camera. Therefore, the camera captures the side of White pea bean sample constituents facing it only. For complete automated White pea bean sample inspection, the system should consider both sides of each kernel. For this, double sided image
acquisition could be done using two cameras and a transparent material to lay the sample constituents on.

- The other potential work for the future is solving the problem of touched kernels. When kernels touch each other, the segmentation and feature extraction stages consider the touched kernels as one.

- Shadows that could possibly be introduced at image capturing phase. Therefore, future studies can extend by proposing an existing algorithm or new algorithms to remove the effect of shadow.

- Due to lack of training data, this research work has not included filth. Therefore, future studies can extend this work to include filth as the eighth class to which White pea bean sample constituents could be classified to.
References

1. S. H. a. R. LOADER, "Barriers to Agricultural Exports from Developing Countries: The Role of Sanitary and Phytosanitary Requirements," 2000.
2. "www.ECX.com.et," [Online].
3. D.-W. S. Tadhg Brosnan, "Improving quality inspection of food products by computer vision - a review," *Journal of Food Engineering*, 6 May 2003.
4. RAJI A. O., A. O. ALAMUTU, "PROSPECTS OF COMPUTER VISION AUTOMATED SORTING SYSTEMS IN AGRICULTURAL PROCESS OPERATIONS IN NIGERIA," *the CIGR Journal of Scientific Research and Development*, vol. Vol. VII, February 2005.
5. Xu Liming *, Zhao Yanchao, "Automated strawberry grading system based on image processing," September 2009.
6. E. Asfaw, "Physical Quality and Grading Systems of Ethiopian Coffee in Demand-Supply Chain, Four Decades of Coffee Research and Development in Ethiopia," 2007.
7. B. Tudu, "An Automated Machine Vision Based System for Fruit Sorting and Grading," 2012.
8. S. Gunasekaran, "Computer vision technology for food quality assurance," vol. Vol. 7, August 1996.
9. Narendra, V.G. and Hareesh, K.S., 2010. Prospects of Computer Vision Automated Grading and Sorting Systems in Agricultural and Food Products for Quality Evaluation, International Journal of Computer Applications (0975 – 8887), Volume 1 – No. 4.
10. Majumdar, S., Jayas, D.S., Bulley, N.R., 1997. Classification of cereal grains using machine vision1: Morphological features. In: 1997 ASAE Annual International Meeting Technical Papers, Paper No. 973101.
11. Tinku Acharya and Ajoy K. Ray, *Image Processing Principles and Applications*, Jhon Wiley, 2005.
12. R. Gonzalez and R. Woods, *Digital Image Processing*, Third Edition, AddisonWesley, 2008.
13. C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing*, First Edition, Wiley-Blackwell, 2011.
14. T. Acharya and A. Ray, *Image Processing Principles and Applications*, John Wiley, 2005.
15. S. Saini and K. Arora, “A Study Analysis on the Different Image Segmentation Techniques”, *International Journal of Information and Computation Technology*, Vol. 4, No. 14, 2014, pp. 1445-1452.
16. W. Pratt, *Digital Image Processing*, Fourth Edition, John Wiley, 2007.
17. C. Solomon and T. Breckon, Fundamentals of Digital Image Processing, First Edition, Wiley-Blackwell, 2011.
18. William K. Pratt: Digital image processing, PIKS Scientific inside, John Wiley, 4th Edition, 2007
19. B. Armstrong, “Using Digital Image Analysis for Assessing the Quality of Wheat and Barley”, Unpublished Master’s Thesis, School of Science and Engineering, University of Ballarat, December 2004.
20. R. Dass, Priyanka, and S. Devi, “Image Segmentation Technique”, International Journal of Electronics and Communication Technology, Vol. 3, Issue 1, January - March 2012, pp. 66-70.
21. Salem, S.A., 2010. Image Segmentation by Using Edge Detection, International Journal on Computer Science and Engineering Vol. 02, No. 03, 804-807.
22. John C. Russ: The Image Processing Handbook, CRC Press, 3rd Edition, 1998
23. Shapiro, L. G. and G. C. Stockman, Computer Vision, Prentice Hall, 2001.
24. Coolen, A. C. C., R. Kühn and P. Sollich: Theory of Neural Information Processing Systems, Oxford University Press, 2005
25. Majumdar, S. and D.S. Jayas: Classification of cereal grains using machine vision. IV. Combined morphology, color and texture models. Trans. ASAE, 43(6):16891694, 2000.
26. Granitto, P. M., H. D. Navone, P. F. Verdes and H. A. Ceccatto: Automatic Identification of Weed Seeds by Color Image Processing, Instituto de Fisica Rosario(CONICET- Universidad Nacional de Rosario), Argentina, 2000
27. Paliwal, J., N. S. Shashidhar and D. S. Jayas: Grain Kernel Identification Using Kernel Signature, Transactions of the ASAE, VOL. 42(6): 1921-1924, 1999.
28. Zayas, I., Y. Pomeranz and F.S Lai: Discrimination of Wheat and Nonwheat Components in Grain Samples by Image Analysis. Cereal Chemistry, 6(3):233-237, 1989.
29. Liu, Zhao-yan, Cheng Fang, Ying Yi-bin and Rao Xiu-qin: Identification of Rice Varieties Using Neural Networks. Journal of Zhejiang University Science, 6B (11):1095-1100, 2005.
30. Visen, N.S., J. Paliwal, D.S. Jayas, and N.D.G. White: Specialist Neural Networks for Cereal Grain Classification, Biosystems Engineering 82 (2): 151–159, 2002.
31. Jayas, D. S., J. Paliwal and N. S. Visen: Multi-layer Neural Networks for Image Analysis of Agricultural Products, Journal of Agricultural Engineering Research. 77(2):119-128, 2000.
32. Ding, K. and S. Gunasekaran: Shape feature extraction and Classification of food Material Using Computer Vision, Food and Process Engineering Inst. of ASAE, 1994.
33. S. Arun Balaj and K. Baskaran, “Design and Development of Artificial Neural Networks (ANN) System Using Sigmoid Activation Function to Predict Annual Rice Production in Tamilnadu”, International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol. 3, No.1, 2013, pp. 13-31.

34. B. Raba, K. Nowakowski, and P. Boniecki, “Visual Quality Evaluation of Malting Barley with use of Neural Image Analysis”, Journal of Research and Applications in Agricultural Engineering Vol. 60, No. 1, 2015, pp. 80-83.

35. A. Pazoki and Z. Pazoki, “Classification System for Rain Fed Wheat Grain Cultivars Using Artificial Neural Network”, African Journal of Biotechnology Vol. 10, No. 41, 2011, pp. 8031-8038.

36. A. Pazoki, F. Farokhi, and Z. Pazoki, “Corn Seed Varieties Based on Mixed Morphological and Color Features Using Artificial Neural Network”, Research Journal of Applied Sciences, Engineering and Technology Vol. 6, Issue 19, 2013, pp. 3506-3513.

37. C. Lira and P. Pina, “Automated Grain Shape Measurement Applied to Beach Sands”, Journal of Coastal Research SI, Vol. 56, 2009, pp. 1527-1531.

38. J. Han, M. Kamber, and J. Pei, Data Mining Concepts and Techniques, Third Edition, Morgan Kaufmann Publishers, 2012

39. C. Solomon and T. Breckon, Fundamentals of Digital Image Processing, First Edition, Wiley-Blackwell, 2011.

40. D. Shiffman, The Nature of Code, Simulating Natural Systems with Processing, 2013.

41. P. Sharma, N. Malik, N. Akhtar, and H. Rohilla, “Feedforward Neural Network: a Review”, International Journal of Advanced Research in Engineering and Applied Sciences (IJAREAS), Vol. 2, No. 10, 2013, pp. 25-34. [27] B. Sudha, P. Gopikannan, A. Shenbagarajan and C. Balasubramanian, “Classification of Brain Tumor Grades using Neural Network”, in Proceedings of the World Congress on Engineering, Vol. 1, 2014.

42. Sonali and B. Maind, “Basics of Artificial Neural Network”, International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC), Vol. 2, Issue. 1, 2014, pp. 96-100.

43. B. Sudha, P. Gopikannan, A. Shenbagarajan and C. Balasubramanian, “Classification of Brain Tumor Grades using Neural Network”, in Proceedings of the World Congress on Engineering, Vol. 1, 2014

44. Jayas, D. S., J. Paliwal and N. S. Visen: Multi-layer Neural Networks for Image Analysis of Agricultural Products, Journal of Agricultural Engineering Research. 77(2):119-128, 2000.
45. H. Lu, R. Setiono and H. Liu, “Effective Data Mining Using Neural Networks”, IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, 1996, pp.
46. K. Liao, M. Paulsen, J. Reid, B. Ni, and E. Bonifacio, “Corn Kernel Shape Identification by Machine Vision Using Neural Network Classifier”, University of Illinois at Urbana-Champaign.
47. C. Bulaong, Q. Ramirez, C. Mangaang, V. Circa and T. Carbonel, “Digital Image Processing System For Corn Quality Analysis”, Technology Resource Development of the National Food Authority, December 2008.
48. W. Steenhoek, K. Misra, R. Hurburgh, and J. Bern, “Implementing a Computer Vision System for Corn Kernel Damage Evaluation”, American Society of Agricultural Engineers (ASAE), Vol. 17, No. 2, 2000, pp. 235-240.
49. Habtamu Minassie Aycheh, “Image Analysis for Ethiopian Coffee Classification”, Unpublished Master’s Thesis, School of Graduate Studies, Addis Ababa University, January 2008.
50. Nagata, M., Cao, Q., Bato, P.M., Shrestha, B.P., Kinoshita, O., 1997. Basic study on strawberry sorting system in Japan. In: 1997 ASAE Annual International Meeting Technical Papers, Page No. 973095, ASAE.
51. Meftah, S. A., Abdul Rashid, M.S., 2011. Recent Methods and Techniques of external Grading systems for agricultural crops quality Inspection-Review International Journal of food engineering, Volume 7, issue3.
52. Wikipedia-Naïve-Bayes-Classifier, http://en.wikipedia.org/wiki/Naive_Bayes_classifier, Visited on May 11-2010.
53. Zhang, G.P., 2000. Neural Networks for Classification: A Survey, IEEE Transactions on systems, MAN AND CYBERNETICS-PART C: Applications and reviews, VOL.30, NO. 4.
APPENDIX A – RGB COLOR SPLITTER AND FALSE REGION REMOVAL

clc;
close all;
clear;

RGB=imread('C:\Users\Mes\Desktop\white pea image\20190510_175213.jpg');
fontSize=16;
HSV=rgb2hsv(RGB);
[H,S,V]=rgb2hsv(RGB);

level=graythresh(RGB);

binaryHue=imbinarize(H,level);
binarySaturation=imbinarize(S,level);

gRedChannel = RGB(:, :, 1);
gGreenChannel = RGB(:, :, 2);
gBluChannel = RGB(:, :, 3);

bR=imbinarize(gRedChannel,level);
bG=imbinarize(gGreenChannel,level);
bB=imbinarize(gBluChannel,level);

binarySaturation= binarySaturation(:,:);
for ii=1:size(bS,1)
    for jj=1:size(bS,2)
        if (binaryHue (ii) == 0)
            bS(ii) = 0;
        end
        if (binarySaturation (ii) == 0)
            binarySaturation (ii) = 255;
        else
            binarySaturation (ii) = 0;
        end
    end
end
[lImg,num]=bwlabel(binarySaturation);
Measure=regionprops(lImg,'all');
%cc=[];
%removed=0;
numberOfBlobs=size(Measure,1);
for k=1:numberOfBlobs
    areas=Measure(k).Area;
    if (areas < 500)
        for j=1:k
            binarySaturation (j)=0;
        end
    end
end
[lbl , num2]=bwlabel(lImg);
Measure=regionprops(lbl,'all');
rows = size(Measure, 1);
columns = size(Measure, 2);
for ii = 1:rows
    for jj = 1:columns
        if (binarySaturation(ii) == 0)
            gRedChannel(ii) = 0; gGreenChannel(ii) = 0; gBlueChannel(jj) = 0;
        end
    end
end

level2 = 0.4;
gRR = imbinarize(gRedChannel, level2);
gGG = imbinarize(gGreenChannel, level2);
gBB = imbinarize(gBlueChannel, level2);
f = (gRR & gGG & gBB);
figure,
imshow(f, []); title('Red, Green... merged image');
smallestAcceptableArea = 100;
f = uint8(bwareaopen(f, smallestAcceptableArea));
figure,
imshow(f, []); title('RGB merged image after bwareaopen');

maskedImageR = f .* gR;
maskedImageG = f .* gG;
maskedImageB = f .* gB;

subplot(3, 3, 2);
RRR = imbinarize(maskedImageR);
imshow(RRR);
title('Masked Red Image', 'FontSize', fontSize);
figure, % subplot(3, 3, 3);
GGG = imbinarize(maskedImageG);
imshow(GGG);
title('Masked Green Image', 'FontSize', fontSize);
figure, % subplot(3, 3, 4);
BBB = imbinarize(maskedImageB);
imshow(BBB);
title('Masked Blue Image', 'FontSize', fontSize);

% Concatenate the masked color bands to form the rgb image.
% maskedRGBImage = cat(3, maskedImageR, maskedImageG, maskedImageB);
% maskedRGBImage = (maskedImageR & maskedImageG & maskedImageB);

% Show the masked off, original image.
figure, % subplot(3, 3, 5);
imshow(maskedRGBImage); title('merged image (R G B)');
figure(1)
set(gcf, 'position', get(0, 'screensize'));
subplot(131), imshow(RGB); title('original image');
% subplot(132), imshow(maskedImageR); title('R');
% subplot(133), imshow(maskedImageG); title('G');
% subplot(131), imshow(maskedImageB); title('B');
subplot(132), imshow(maskedRGBImage); title('false region removed image');
%%

binaryImage=bwareaopen(maskedRGBImage,500);
figure,
imshow(binaryImage,[]);
title('..emoved small objects');
%%
[lbl,numObject]=bwlabel(binaryImage);
measure=regionprops(lbl,'Eccentricity','Perimeter','Area','BoundingBox');
%%
figure,
imshow(binaryImage,[]),
title('visualize labeled binary')
%vislabels(lbl)
areas=[measure.Area];
p=[measure.Perimeter];

for Object=1:numObject
    if(areas(Object)<1)
        centroid=[measure(Object).Centroid];
        h=rectangle('position',d(Object).BoundingBox,'EdgeColor','r');
        %set(h,'EdgeColor',[.75 0 0]);
        hold on;
    end
end
if blobO>3
    title(['there are',num2str(numObject),'objects in the image']);
end
hold off;
