Addis Ababa University

College of Natural Sciences

Development of Automatic Maize Quality Assessment System Using Image Processing Techniques

Daniel Hailemichael Lemessa

A Thesis Submitted to the Department of Computer Science in Partial Fulfillment for the Degree of Master of Science in Computer Science

Addis Ababa, Ethiopia

November 2015

Addis Ababa University

College of Natural Sciences

Daniel Hailemichael Lemessa

Advisor: Yaregal Assabie (PhD)

This is to certify that the thesis prepared by *Daniel Hailemichael Lemessa*, titled: *Development of Automatic Maize Quality Assessment System Using Image Processing Techniques* and submitted in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Committee:

Name	Signature	Date
Advisor:		
Examiner:		
Examiner:		

ABSTRACT

Maize is a very important crop where its circulation in the market has to conform to the rules of quality inspection. Currently, maize sample quality inspection is performed manually by human experts through visual evaluation and the constituents will be 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 maize sample constituents using digital image processing techniques and artificial neural network classifier based on the standard for maize set by the Ethiopian Standards Agency. A novel segmentation technique is proposed to segment and lay the foundation for feature extraction. A total of 24 features (14 color, 8 shape and 2 size) have been identified to model maize sample constituents.

For classification of maize 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 534 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 97.8%. The success rates for detecting foreign, rotten and diseased, healthy, broken, discolored, shriveled and pest damaged kernels are 100%, 95.2%, 98.6%, 98.8%, 100%, 98.4%, and 94.8%, respectively.

Keywords: Artificial neural network, Maize quality assessment, Reconstructed image, Merged image, Color image segmentation, Digital image processing, Color structure tensor

Dedicated To

My mom (Tsehay W/Yohnnes), wife (Selamawit Hailu), daughter (Christian Daniel)

And

The great Meles Zenawi Asres

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my creator and savior, Jesus Christ, 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. Yaregal Assabie for his 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 maize samples and for sample classification expertise they provided for this research.

I also would like to extend my special appreciation to Minale Habetemichael for providing me with Matlab 2014a and Visio 2013.

Most of all, I would like to thank my wife and my daughter for the encouragement, patience and love they granted to me in the long process of this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANN	Artificial neural networks
B-P	Backpropagation
CMY	Cyan-magenta -yellow
CMYK	Cyan-magenta-yellow-black
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
KNN	K Nearest Neighbor
MNN	Multilayer neural network
MSE	Mean square error
PD	Pest Damage
RD	Rotten and Diseased
RGB	Red-green-blue
SVM	Support Vector Machines

CHAPTER ONE: INTRODUCTION

1.1 Background

Maize is one of the most extensively cultivated cereal crops on earth. More is produced, by weight, than any other grain, and almost every country on earth cultivates maize for a variety of uses [1]. Globally, maize is a staple crop, and many people rely on it as a primary source of nutrition. In addition to playing a major role in the human diet, it is also used as livestock food. Maize is processed to make an assortment of products ranging from high fructose maize syrup to bio fuels [1].

Maize was introduced into Africa in the 1500s and has since become one of the dominant food crops of the continent. Like many other regions, it is consumed as a vegetable although it is a grain crop. The grains are rich in vitamins A, C and E, carbohydrates, and essential minerals, and contain 9% protein. Moreover, they are also rich in dietary fiber and calories which are a good source of energy [2]. Worldwide consumption of maize is more than 116 million tons, with Africa consuming 30% and sub:Saharan Africa 21%. However, Lesotho has the largest consumption per capita with 174 kg per year. Eastern and Southern Africa uses 85% of its production as food, while Africa as a whole uses 95%, compared to other world regions that use most of its maize as animal food [2]. Ninety percent of white maize consumption is in Africa and Central America. It fetches high income in Southern Africa where it represents the main staple food. Yellow maize is preferred in most parts of South America and the Caribbean. It is also the preferred animal food in many regions as it gives a yellow color to poultry, egg yolks and animal fat [2]. Maize is processed and prepared in various forms depending on the country. Ground maize is prepared into porridge in Eastern and Southern Africa, while maize flour is prepared into porridge in West Africa. Ground maize is also fried or baked in many countries. In all parts of Africa, fresh maize is boiled or roasted on its cob and served as a snack. Popcorn is also a popular snack [2].

Maize is also an important crop in Ethiopia. It is grown in the mid highland areas of the country. There are huge tracts of land in all regions suitable for maize farming. Maize is

mainly produced in Southern Nations, Nationalities and People Regional State and Oromia where there are about 1.77 million hectares under cultivation [3].

Among cereals, maize accounts for the largest share in total production and the total number of farm holdings involved. In Ethiopia, in the year 2010/11, maize accounted for 28% of the total cereal production, compared to 20% for teff and 22% for sorghum, the second and third most cultivated crops. About eight million smallholders were involved in maize production in 2010/11, compared to 6.2 million for teff and 5.1 million for sorghum. Maize is the largest and most produced crop in Ethiopia. According to the data of the Central Statistical Agency (CSA), in 2007/08, maize production was 3.75 million tones, 25% higher than teff and 41% higher than sorghum. Maize yield is the highest among cereal crops. Between 2003/04 and 2007/08, maize production expanded by 103% [7].

Ethiopia is the second largest producer of maize in Eastern and Southern Africa, following South Africa. Between 2000 and 2010, it accounted for 12.3% of the total maize production in the region, compared to 36.3% for South Africa. Tanzania, the third largest producer, accounted for 11.7% of the total maize production. With improved infrastructure and expanded use of improved production technology, Ethiopia has the potential of exporting maize to the region [7].

Six major staples, maize, teff, wheat, sorghum, barley and enset (false banana), dominate the national food basket in Ethiopia. Maize is the single most important cereal, accounting for 17% of the per capita calorie intake, followed by sorghum (14%) and teff (11%) in the years 2003/4 and 2007/8 [7].

Maize trading and circulation in Ethiopia is subjected to a standard set by the Quality and Ethiopian Standards Agency (ESA). This standard sets criteria by which maize quality is evaluated. The standard is based on morphological and chemical characteristics of maize. When consignment of maize arrives at the authority, small amounts of maize will be taken from different sacks and get mixed. Based on this, the maize sample will be manually classified into foreign matter, broken, pest damaged (PD), rotten and diseased (RD), discolored and shriveled and healthy classes.

1.2 Motivation

ESA sets the standard to various agricultural and non agricultural commodities. It also tests if a commodity destined for inland or overseas market conforms to the quality standard it sets. However, currently the organization does not apply automated techniques to control the quality of commodities. Generally, 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 in Ethiopia.

Particularly, Ethiopian maize quality inspection is based on manual ways of classification and grading system. Therefore, the implementation of digital imaging technology in the sector of quality control of maize will have a major impact to facilitate commercial activities by increasing efficiency.

1.3 Statement of the Problem

Maize is a very important crop in the economy of Ethiopia as large quantities of the crop is circulated in the market for both *domestic* and *export* purposes. However, maize quality may be negatively affected during production, storage and transportation. Consequently, the maize may become unsafe for human consumption. In order to make sure maize quality is maintained, governments set quality standards to which trading and circulation of maize should conform to. These standards set maize quality based on parameters such as amount of broken, insect damaged, discolored, and shriveled grains, etc.

Currently, maize 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 in the process of exporting the crop. Moreover, this manual process is subjected to bias and inconsistencies associated with human nature.

Automated maize quality assessment has many important advantages over the manual technique. The major advantage of this technique is that it is an objective method for

measurement of morphological features such as shape, size, color and surface texture. It helps experts to describe visible attributes accurately. Hence, the use of automated image analysis to assess maize sample helps to eliminate the problems associated with manual evaluation. After the initial outlay for equipment and research unlike other systems, image analysis has very few additional costs. The speed of analysis is much higher than any of the conventional methods [5].

1.4 Objectives of the Study

General Objective

The main objective of this research is the design and implementation of a model that enables us to assess maize sample quality using digital image processing (DIP) techniques based on the visual quality standard specified for maize by ESA.

Specific Objectives

The specific objectives of this research work are to:

- Conduct literature review on previous studies made on maize and other cereals.
- Collection of samples representing the various constituents of maize sample.
- Identify features the can be used to model maize sample constituents.
- Design appropriate segmentation algorithm.
- Develop algorithms that are used for extracting features.
- Design a neural network classifier.
- Implement the algorithms developed for segmentation, feature extraction and classification.
- Evaluate the performance of the implemented prototype.

1.5 Scope and Limitations

This work is limited to the determination of maize sample quality based on size, color and shape characteristics of maize sample and does not include moisture content analysis, mass determination and chemical content analysis.

1.6 Methodology

Literature Review

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 will be reviewed. The review will include various kinds of materials including books, previous research works, Internet and articles. From these insight reviews image analysis techniques and tools that were employed on agricultural products quality identification and disease detection and that were pertinent to this work will be selected.

Sample Collection

We need maize sample to carry out this research work. The sample should contain enough number of representative kernels from each class type. These grains will be collected from the Ethiopian Grain Trading Enterprise (EGTE) after proper sorting into their respective categories. Then, images of these grains will be taken. A digital camera model cannon Model SD630, will be used to capture maize kernel images.

Tools

MATLAB version R2014a will be used to implement the prototype of the system. MATLAB is software developed by MathWorks. Visio 2013 will be used for designing the system architecture, algorithms and neural network classifier.

Prototype Development

To test its effectiveness, the developed system will be evaluated. This will show us its strengths and weaknesses

1.7 Application of Results

This research work will have several applications. These are:

- The objective assessment of maize quality so as to avoid the subjectivity, bias, inconsistency and tiredness associated with human nature.
- The rapid evaluation of maize yield content so as to make decision on optimum cleaning strategies.
- The facilitation of quality based transactions. For instance, the facilitation of the buying and selling activities carried out by the Ethiopian Commodity Exchange (ECX), or EGTE.
- The minimization of corruption (bribery) that might occur during manual grading of maize as merchants can bribe grading experts.
- The facilitation of maize grade determination during export of the crop to other countries. This is in line with the ambition of Ethiopia to export maize and other grains in larger quantities.

1.8 Organization of the Document

The rest of this thesis is organized into five chapters. In Chapter Two, literature will be reviewed. In Chapter Three, image processing works that are related to cereal grain in general and maize in particular will be reviewed. Chapter Four will discuss the design of the proposed solution. In Chapter Five, the experiments used to evaluate the performance of the proposed solution will be discussed. In Chapter Six conclusions, future work and the contributions of this research work will be presented.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

The beginning of image processing techniques dates back to the late 1960s and early 1970s and was used in medical imaging, remote earth resources observation, and astronomy. Since then, digital image processing has been growing vigorously. Other application areas are now using digital image processing techniques. Now, the applications of digital image processing can be found in areas of product inspection, robotic vision, and scientific research [6].

In this chapter, maize production and consumption facts followed by concepts related to digital image representation, processing and classification will be discussed.

2.2 Maize Sample Constituents

The following are the descriptions of maize sample constituents.

- **Foreign matters** are all organic and inorganic materials such as plant parts, sand, soil, glass, filth other than maize [4]. Foreign matters are shown in Figure 2.1.

Figure 2.1: Foreign Matters

- **Broken kernels** are broken pieces of maize kernels [4]. Broken kernels are shown in Figure 2.2.

Figure 2.2: Broken Kernels

- **Pest damaged** are kernels with obviously weevil-bored holes or which have evidence of boring or tunneling, indicating the presence of insects, insect webbing or insect refuse, or degermed grains, chewed in one or more than one part of the kernel which exhibit evident traces of an attack by vermin [4]. Samples of PD grains are shown in Figure 2.3.

Figure 2.3: PD Grains

- **Rotten and diseased** are grains made unsafe for human consumption due to decay, moulding, or bacterial decomposition, or other causes that may be noticed without having to cut the grains to examine them [4]. Samples of RD grains are shown in Figure 2.4.

Figure 2.4: Rotten and Diseased Maize Grains

- **Discolored** grains are kernels materially discolored by excessive heat, including that caused by excessive respiration (heat damaged) and dried damaged kernels. Kernels may appear darkened, wrinkled blistered, puffed or swollen, often with discolored, damaged germs. The coat may be peeling or may have peeled off completely, giving kernels a checked appearance [4]. Samples of discolored grains are shown in Figure 2.5.

Figure 2.5: Discolored Grains

- **Immature/Shriveled grains** are maize grains which are underdeveloped, thin and papery in appearance [4]. Samples of shriveled grains are shown in Figure 2.6.

Figure 2.6: Shriveled/Immature Grains

- **Filth** is impurity of animal origin [4]. Filth is shown in Figure 2.7.

Figure 2.7: Filth (dung) Shown in the Circle

- **Healthy** maize kernels are those kernels having the normal size and have no damage. Examples of healthy maize kernels are shown in Figure 2.8.

Figure 2.8: Healthy Maize Grains

2.3 Digital Image Representation

There are numerous ways to acquire images, but our objective in all is the same. The objective of acquiring images is to generate digital images from sensed data. The output of most sensors is a continuous voltage waveform. The amplitude and spatial behavior of these waveforms are related to the physical phenomenon being sensed. Therefore, in order to create a digital image, we need to convert the continuous sensed data into digital form [6, 10].

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 is called sampling. Digitizing the amplitude values is called quantization [6].

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 is called the spatial domain, with x and y being referred to as spatial variables or spatial coordinates [6, 11].

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) [6, 10].

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 x N numerical array as shown in

Figure 2.9 [6]. 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 [6].

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 [6, 10, 11].

$$f(x, y) = \begin{pmatrix} f(0, 0) & f(0, 1) & \dots & f(0, N-1) \\ f(1, 0) & f(1, 1) & \dots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1, 0) & f(M-1, 1) & \dots & f(M-1, N-1) \end{pmatrix}$$

Figure 2.9: Representation of an MXN Numerical Array

2.4 Image Color Fundamentals

There are a number of distinct image types and they are expressed as 2D or 3D arrays. A 2D array that assigns one numerical value from the set {0, 1} to each pixel in the image is known as *binary image*. Likewise, a 2D array that assigns one numerical value to each pixel from the set {0, 1, 2 ... 255} is called *grayscale image*. Both binary and grayscale images consist of single color channel. On the other hand, a 3-D array that assigns three numerical values to each pixel, each value corresponding to the red, green and blue (RGB) image components respectively is called RGB or true-color image. In the case of binary and grayscale images, each pixel location only contains a single numerical value representing the signal level at that point in the image. Since these are only numbers, there need to be a way to change them to the actual colors. The conversion from this set of numbers to an actual (displayed) image is achieved through a *color map*. In order to give a visual representation of the data, a color map assigns a specific shade of color to each numerical level in the image. In the case of binary and grayscale images, the color map assigns all shades of grey from black (zero) to white (maximum) according to the signal level [6, 11].

However, in the case of true color images, the full spectrum of colors can be represented as a triplet vector, typically the (R, G, B) components at each pixel location. Here, the color is represented as a linear combination of the basic colors or values. True color images may be considered as consisting of three 2-D planes. Other representations of color are also possible and used quite widely, such as the (H, S, V) (hue, saturation and value (or intensity)). In this representation, the intensity V of the color is decoupled from the chromatic information, which is contained within the H and S components [11].

The representation of colors in an image is achieved using a combination of one or more color channels that are combined to form the color used in the image. The representation we use to store the colors, specifying the number and nature of the color channels, is generally known as the color space [11].

Color space or color model refers to a coordinate system where each color stands for a point. The often used color models consist of the RGB (red-green-blue) model, CMY (cyan- magenta-yellow) model, CMYK (cyan- magenta-yellow- black) model and HSI (hue-saturation- intensity) model [13].

RGB model consists of three components. These three components are combined together to produce composite colorful images. Each image pixel is formed by a number of bits. The number of these bits is called pixel depth. A full color image is normally 24 bits, and therefore the total number of the colors in a 24-bit RGB image is 16,777,216. RGB (or true color) images are 3-D arrays that we may consider conceptually as three distinct 2-D planes, one corresponding to each of the three red (R), green (G) and blue (B) color channels. RGB is the most common color space used for digital image representation as it conveniently corresponds to the three primary colors which are mixed for display on a monitor or similar device [6, 11].

It is possible to separate and view the red, green and blue components of a true-color image. The colors that are typically present in a real image are nearly always a blend of color components from all three channels. A common misconception is that, for example, items that are perceived as blue will only appear in the blue channel and so forth. Items perceived as blue, for example, will certainly appear brightest in the blue channel. This means items perceived as blue will contain more blue light than the other colors. In addition to the blue part, however, they also have milder components of red and green. If we consider all the colors that can be represented within the RGB representation, then we appreciate that the RGB color space is essentially a 3-D color space (cube) with axes R, G and B. Each axis has the same range 0-1. This is scaled to 0–255 for the common 1 byte per color channel, 24-bit image representation. The color black occupies the origin of the cube, at position (0, 0, 0), corresponding to the absence of all three colors. White color occupies the opposite corner at position (1, 1, 1), indicating the maximum amount of all three colors. All other colors in the spectrum lie within this cube. The RGB color space is based upon the portion of the electromagnetic spectrum visible to humans. RGB color space is represented in Figure 2.10 [12]. Colors

along the main diagonal have gray values from back at the origin to white at point (1, 1, 1) [7, 11, 12].

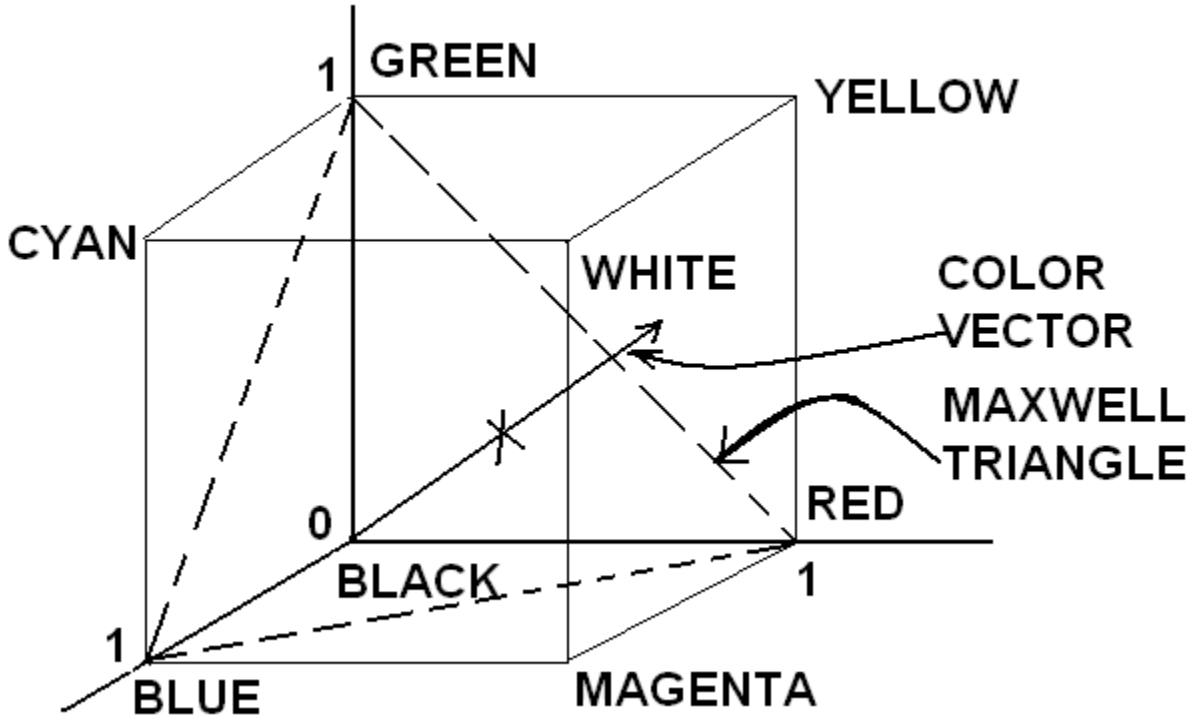


Figure 2.10: RGB Color Space as a 3-D Cube

In digital image processing we use a simplified RGB color model that is optimized and standardized towards graphical displays. However, the primary problem with RGB is that it is perceptually nonlinear. By this we mean that moving in a given direction in the RGB color cube does not necessarily produce a color that is perceptually consistent with the change in each of the channels. For example, starting at white and subtracting the blue component produces yellow; similarly, starting at red and adding the blue component produces pink. For this reason, RGB space is inherently difficult for humans to work with and reason about. This is because it is not related to the natural way we perceive colors. As an alternative we may use perceptual color representations such as HSV [11, 13].

Since RGB model is not natural to the way human beings perceive color, it is important to have an alternative way of representing true color images in a manner that is more natural to the human perception and understanding. HSV (Hue-saturation-value) is one of such color spaces. It is popular in image analysis applications.

Unlike RGB model, changes within HSV color space follow a perceptually acceptable color gradient. Moreover, from an image analysis perspective, it allows the separation of color from lighting to a greater degree. We can use the advantage HSV color space offers by converting RGB image into HSV representation. H stands for hue which is the dominant wavelength of the color. S stands for saturation and is the 'purity' of color (in the sense of the amount of white light mixed with it). V stands for value is the brightness of the color which is also known as luminance. The formulae for the conversion of RGB to HSV space are shown in Equations (1), (2), (3) and (4) [11].

$$H = \begin{cases} \theta \\ 360 - \theta \end{cases} \tag{1}$$

$$\theta = COS^{-1}\left(\frac{0.5[(R-G)+(R-B)]}{\left[[(R-G)^2+(R-B)(G-B)]^{\frac{1}{2}}\right]} \right) \tag{2}$$

$$S = 1 - \frac{3[\min(\mathrm{R}, \mathrm{G}.\mathrm{B})]}{\mathrm{R}+\mathrm{G}+\mathrm{B}} \tag{3}$$

$$I = \frac{1}{3(\mathrm{R}+\mathrm{G}+\mathrm{B})} \tag{4}$$

2.5 Digital Image Processing

The field of digital image processing refers to processing digital images by means of a digital computer. It involves processing or altering an existing image in a desired manner. Image processing can be seen as composed of three high level stages or

processes. These stages are termed as low, mid, and high level processes. Low-level processes involve primitive operations such as image preprocessing to reduce noise, contrast enhancement, and image sharpening. A low-level process is characterized by the fact that both its inputs and outputs are images. Mid-level processing on images involves tasks such as segmentation. This is the task of describing those objects to reduce them to a form suitable for computer processing, and classification of individual objects. Unlike low-level processing, a mid-level process is characterized by the fact that its inputs generally are images, but its outputs are attributes extracted from those images. This attributes could be, edges, contours, and the identity of individual objects. Finally, higher-level processing involves "making sense" of a group of recognized objects [6]. These steps of image processing are discussed in the following sections.

2.5.1 Image Preprocessing

Image preprocessing is one of the low 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 [10].

2.5.2 Image Segmentation

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 [9, 10, 12].

Several of different segmentation techniques are available, but there is not a single method which can be considered good for different images. All techniques are not equally good for a particular type of image. Hence, algorithm development for one class of images may not always be applied to other class of images. As a result, there are many challenging issues like development of a unified approach to image segmentation which can be applied to all types of images. Often, the selection of an appropriate technique for a specific type of image is a difficult problem. Thus, in spite of several decades of research, there is no universally accepted method for image segmentation. Therefore, this problem remains a challenging issue in the field of image processing and computer vision [13].

The choice of one segmentation technique over another and the level of segmentation are decided by the particular type of image and characteristics of the problem being considered. Basically, there are two categories of segmentation techniques, namely, edge-based and region based segmentation. They are based on detection of discontinuities and detection of similarities respectively. The segmentation technique based on detection of discontinuities is the process of partitioning an image based on abrupt changes in intensity. Examples of such algorithms include all edge detection algorithms. Edge detection techniques are generally used for finding discontinuities in gray level images. Edges are local changes in the image intensity and they typically occur on the boundary between two regions. Spatial masks can be used to detect all types of discontinuities in an image. These spatial masks are commonly referred to as edge detection operators. These spatial masks are grouped into two classes. The first group is called the first order derivative group and includes the Prewitt, Sobel, Canny and Test Operators. Canny is the most promising one, but takes more time as compared to Sobel operator. The second group is known as the second order derivative group. This group includes Laplacian and Zero-crossings operators. Edge-based segmentation is based on

the fact that the position of an edge is given by an extreme of the first-order derivative or a zero crossing in the second-order derivative. On the contrary, region based methods are based on continuities. These techniques divide the entire image into sub regions depending on some similarity rules (criteria). Examples of algorithms that work based on detection of similarity include thresholding and region growing. Whether the segmentation technique used is discontinuity or region based, the end result of any segmentation process is a binary image [9, 13, 14].

Since color and morphological features of maize need to be extracted on a single kernel basis, each of the maize kernels in the image should be recognized and isolated from the rest and from the background of the image. Therefore, the purpose of segmentation is to isolate individual maize grains from each other and from the background so that their size, color and shape features can be extracted on a single kernel basis. To achieve this, the boundary of each maize kernel should be identified using a proper technique such as color structure tensor segmentation algorithm.

a) Color Structure Tensor Based Segmentation

Tensors are simply mathematical objects that can be used to describe physical properties, just like scalars and vectors. In fact tensors are merely a generalization of scalars and vectors. A scalar is a zero rank tensor, and a vector is a first rank tensor [15].

The structure tensor is a matrix derived from the gradient of a function. It summarizes the predominant directions of the gradient in a specified neighborhood of a point, and the degree to which those directions are coherent. The structure tensor is often used in image processing and computer vision where it is used as a matrix representation of partial derivative information of an image. These partial derivatives of images represent the gradient or "edge" information of the image. It also has a more powerful description of local patterns as opposed to the directional derivative through its coherence measure [15].

When representing gradient information derived from an image, there are several options to choose from. One such choice is the directional derivative. Directional

derivatives provide a vector representation of the gradient information. Its magnitude reflects the maximum change in pixel values while the phase is directed along the orientation corresponding to the maximum change. These two components are calculated as per Equations (5) [15] and (6) [15] respectively.

$$|\vec{v}| = \sqrt{\left(I_x\right)^2 + \left(I_y\right)^2} \quad (5)$$

$$\boldsymbol{\theta v} = \mathbf{tan^{-1}}\left(\frac{I_y}{I_x}\right) \quad (6)$$

where I_x and I_y are the partial derivatives of I.

Although the majority of images are in color format nowadays, the computer vision community still uses luminance based segmentation and feature extraction techniques. Obviously, extensions of these feature detection techniques to the color domain are highly desired. However, differential-based segmentation of color images is hindered by the multi-channel nature of color images. The reason behind this fact is that the derivatives in different channels can point in *opposite* directions; hence cancellation might occur by simple addition [15].

The solution to this problem is attained by the *structure tensor* for which opposing vectors reinforce each other. The color structure tensor algorithm combines feature detectors with photometric invariance to construct invariant features. Photometric invariance is a desired property for color image descriptors. It ensures that the description has certain robustness with respect to scene incidental variations such as changes in viewpoint, object orientation, and illuminant color. Experiments show that color features perform better than luminance based features and that the additional photometric information is useful to discriminate between different physical causes of features during the segmentation process. As mentioned above, simply adding the differential structure of different channels may result in a cancellation even when evident structure exists in the image. A well known mathematical method for which

vectors in opposite directions reinforce each other is provided by tensor mathematics. Tensors describe the local orientation rather than the direction. To be more precise, the tensor of a vector and the tensor of the same vector rotated over 180 degrees are equal [15, 16, 17].

Figure 2.11: (a) Isotropic Case (b) Uniform Case

Although the directional derivative is relatively computational inexpensive, it does possess a weakness. In scenarios where there is no preferred direction of gradient, the directional derivative formula results in a zero magnitude. This scenario can be exemplified by an isotropic structure. An example of such an isotropic structure is a black circle on a white background as shown in Figure 2.11 (a) [15]. There is clearly gradient information but, since there is no preferred phase, the directional derivative cancels itself out and results in zero magnitude. The same is true for the case where the original input is a uniformly colored region as shown in Figure 2.11 (b) [15]. Again, the directional derivative magnitude is zero as there is no gradient information with which to calculate. Therefore, we need a representation capable of discerning between these two examples and properly reflect the presence of gradient information in the first case, but none in the second [15].

A different method of representing gradient information is by using the structure tensor. It also makes use of the $\mathbf{I_x}$ and $\mathbf{I_y}$ values, however, in a matrix form. The term tensor refers to a representation of an array of data. The rank (or order) of a tensor is defined by the number of directions (and hence the dimensionality of the array) required to describe it. This implies that a vector is a tensor of rank one. A two-dimensional matrix

is a tensor of rank two and so and so forth. The 2D structure tensor matrix is formed as per Equation (7) [15].

$$\boldsymbol{S} = \begin{bmatrix} (I_x)^2 & I_x I_y \\ I_x Iy & (I_y)^2 \end{bmatrix} \tag{7}$$

where I_x and I_y are the partial derivatives of I.

Eigen decomposition is then applied to the structure tensor matrix 'S' to form the eigenvalues and eigenvectors (λ1, λ2) and (e1, e2) respectively. Eigen decomposition is the method to decompose a square matrix into its eigenvalues and eigenvectors. Eigenvalues and eigenvectors are numbers and vectors respectively. Both are gradient features that give us a more precise description of the local gradient characteristics, which analyzes the structure of this matrix. For example, e1, is a unit vector directed normal to the gradient edge, while the e2 vector is tangent. The eigenvalues measure the underlying certainty of the gradient structure along their associated eigenvector directions. The coherence is obtained as a function of the eigenvalues. This value is capable of distinguishing between the isotropic and uniform cases where the directional derivative is unable to do. The coherence is calculated as per Equation (8) [15].

$$\left\{ \frac{\boldsymbol{\lambda 1 - \lambda 2}}{\boldsymbol{\lambda 1 + \lambda 2}} \;\middle|\; \begin{matrix} \textbf{if } (\boldsymbol{\lambda 1 + \lambda 2}) > 0 \\ \textbf{0 otherwise} \end{matrix} \right\} \tag{8}$$

where λ1, λ2 are the eigenvalues.

b) Photometric Invariant Edge Detection

By using a physical reflection model it is possible to differentiate between different physical causes of edges. It is, for example, possible to differentiate between scene incidental edges such as shadow, shading and specular edges and high informative edges such as material edges. An example of an image and its corresponding

photometric invariant images, coined quasi-invariants are shown in Figure 2.12 [15]. Figure 2.12(a) and Figure 2.12(b) show the original image and its corresponding gradient images respectively. Figure 2.12(c) shows the results of the shadow-shading quasi-invariant which does not respond to shading, shadow edges, such as the shading edges on the green-block, and the shadow edge on the ground next to the red ball. It does however respond to specularities, as can be seen on the red ball and the yellow torus. Figure 2.12(d) shows an edge-image showing the results of the shadow-shading and specular quasi-invariant. In this image also the specular edges are removed, such as the highlight on the red ball. Only the more informative material edges are detected [15].

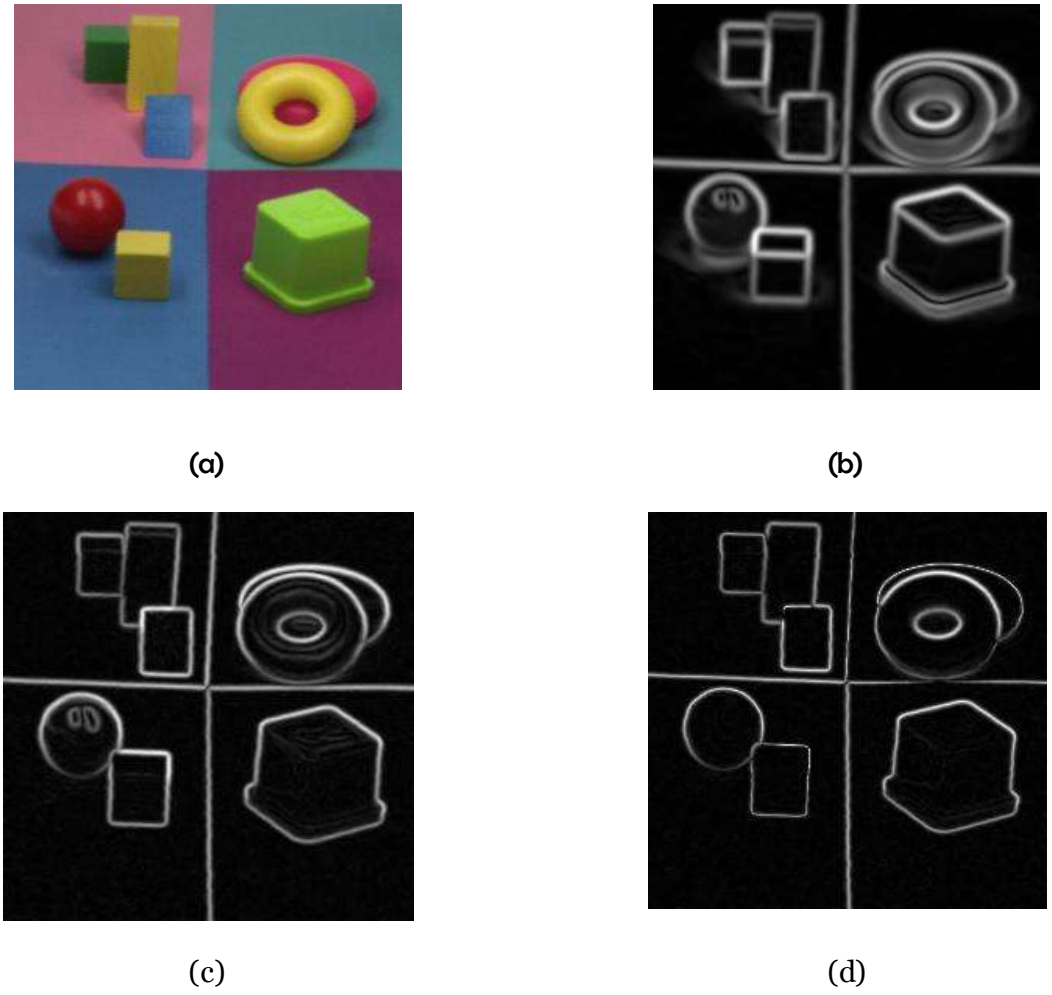


Figure 2.12: (a) Original Image (b) Gradient Image (c) Shadow-Shading Invariant Image (d) Shadow-Shading and Specularity Invariant Image

c) Thresholding

Any image consists of background and foreground objects. Depending on their objectives, image analysis works may or may not need the background information of the images. For instance, works related to content based image retrieval require the background information of images, as it is part and parcel of the image's identity. However, works such as those related to facial and finger print recognition need to separate the background and foreground parts of images. Likewise, since maize image backgrounds are not part of maize sample image, we need to isolate and remove it. In fact, image background introduces unwanted color values into the extracted color features. Hence, background objects should be distinguished and separated from the foreground maize objects. This process of separating foreground objects from the background is achieved through a technique called thresholding [10, 12].

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 [10, 12, 18].

Unfortunately, the above explanation works fine for grey scale images as they are one dimensional. However, RGB images are composed of three color bands [6]. 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 maize 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 [13].

Selection of a good threshold value results in a segmented image in which each foreground pixel is recognized as belonging to the foreground and each background pixel is recognized as belonging to the background. Two types of thresholding methods are in existence, global and local thresholding. When the threshold value is constant, the approach is called global thresholding otherwise it is called local thresholding. Global thresholding methods can fail when the background illumination is not uniform. In local thresholding, multiple thresholds are used to compensate for uneven illumination. Threshold selection of this value can be done manually by trial and error or automatically. Manual segmentation is good for fine tuning and understanding the operation. However, it is time consuming and impractical for many applications. We need techniques that select threshold values automatically [9, 11, 19].

Therefore, many image processing tasks require full automation, and there is often a need for some criterion for selecting a threshold automatically. This is because when the level of illumination changes during image acquisition, images tend to have different intensities [13, 19].

2.5.3 Feature Extraction

After segmentation is done, feature extraction is the next major activity in digital image processing. It is the task of describing objects to reduce them to a form suitable for computer processing. This form is called an image feature. Image feature is a distinguishing primitive characteristic or attribute of an image. This section describes several types of image features that have been proposed for feature extraction [6, 12].

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 [21].

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 [18, 28, 31].

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 woks [18, 28, 31]. .

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 [29] 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 [18, 23, 28, 31].

Even though ovality, solidity and convexity describe shape in general, we need a robust shape feature descriptor that is invariant to translation, rotation and scaling. One of such shape descriptors is the Fourier descriptor (FD). Fourier descriptor is used to describe the boundary of a shape in two dimensional space using Fourier methods. Shape descriptors are numbers that are computed from a two-dimensional shape. A shape can be reconstructed from its shape descriptors. The shape descriptors can thus be considered as an approximate description of the shape. Shapes cab be examined for their similarity by way of using their respective shape descriptors. Hence, shape similarity somehow corresponds to similarity of the shape descriptors. Every connected object has a closed contour that can be represented as a sequence of the pixel coordinates. In order to get FDs of a shape, the first step is to take Npoints from the digital boundary of the shape putting it on the XY plane. At this point, we can choose to

take all the pixels occupied by the boundary of the shape, or we can take samples from them. We can do this by traveling the boundary of the shape in the anti-clockwise direction at constant speed for a time period of *N* seconds, taking a coordinate every second. Now, we have a complete set of coordinates describing the boundary. We can call each coordinate (x_t, y_t) where $0 < t \leq N-1$. Now, each coordinate can be expressed as a complex number as given by Equation (9) [35] for $t=0, 1, 2...N-1$. After doing this, we have *N* complex numbers instead of 2 times *N* real numbers. For $k = 0, 1, 2...N-1$, the discrete Fourier Transform of $z(t)$ is given by Equation (10) [35]. The complex coefficients c_k are called the FDs of the boundary. Applying inverse Fourier Transform to c_k restores $z(t)$ as represented by Equation (11) for $k=0, 1, 2...N-1$ [35, 36].

$$z(t) = x(t) + j.y(t) \tag{9}$$

$$ck = \frac{1}{N}\sum_{t=0}^{N-1}\left(z(t)\, e^{-j2\pi kt/N}\right) \tag{10}$$

$$z(t) = \sum_{k=0}^{N-1}\left(ck\, e^{j2\pi kt/N}\right) \tag{11}$$

$$s(k) = \sum_{t=0}^{M-1}\left(ck\, e^{i2\pi kt/N}\right) \tag{12}$$

where **k > M-1.**

The restored pixel values are exactly the same as the original ones. It is not necessary to take all the N pixel values to reconstruct the original image. We can drop the FDs with higher frequencies because their contribution to the image is very small. For $k=0, 1, 2...N-1$ expressing this as an equation we get Equation (12) [35]. FDs are invariant to rotation, skewing, scaling and translation. They also describe whole shapes which makes it possible to compare shapes and with other pictures [35, 36].

2.5.4 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 [20].

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 neuron-like processing units with weighted connections between the units [20].

a) ANN Classifiers

An ANN is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements. These processing elements are called neurons. They work in unison to solve specific problems. Like human beings, ANNs learn by example. ANN should be configured for the specific purpose it is intended to before its use. The configuration is done through a learning process. Learning in ANNs is similar to that in biological systems. It involves adjustments to the synaptic connections that exist between the neurons [21, 22].

ANNs were used widely in researches that are related to agricultural products. For instance, the work in [23] used neural networks to perform visual quality evaluation of

malting barley while the work in [24] classified maize varieties based on kernel shape by using maize shape symmetry features and ANN.

ANNs have many processing elements. They operate by creating connections between many different processing elements. Each processing element is analogous to a single neuron in a biological brain. These neurons could be either physically constructed or simulated by a digital computer. Each neuron takes many input signals, then, based on an internal weighting system, produces a single output signal that is typically sent as input to another neuron. The neurons are tightly interconnected and organized into different layers. The first layer is the input layer and hence it receives the input. The last layer, which is also known as the output layer, produces the final output. Between the input and output layers, there exist other layers [21, 25].

Unlike ANNs, the computational systems we write are procedural. In procedural systems, a program starts at the first line of code, executes it, and goes on to the next, following instructions in a linear fashion. A true neural network, however, does not follow a linear path. In ANNs, information is processed collectively and in parallel [25].

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 [26, 27].

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 2.13. 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 [22, 26, 27].

In simplest terms, neural network, initially, makes random guesses and sees how far its answers are from the actual answers and makes an appropriate adjustment to its node-connection weights. When working as a classifier, an MNN operates as a black box which receives an input vector as a set of observations and produces responses from its output units. For a specific vector x, the output gives the binary representation of its class number. An MNN classifier learns the class knowledge directly from the training data set. Compared to k-nearest-neighbor classifier, neural networks take less computer memory and less time in the classification process. The time required to train a neural network strongly depends on the complexity of the network, the size of the training data sets, and the computer speed [22, 26].

i) 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 multi-layer perceptrons. 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 Figure 2.13 [25, 26].

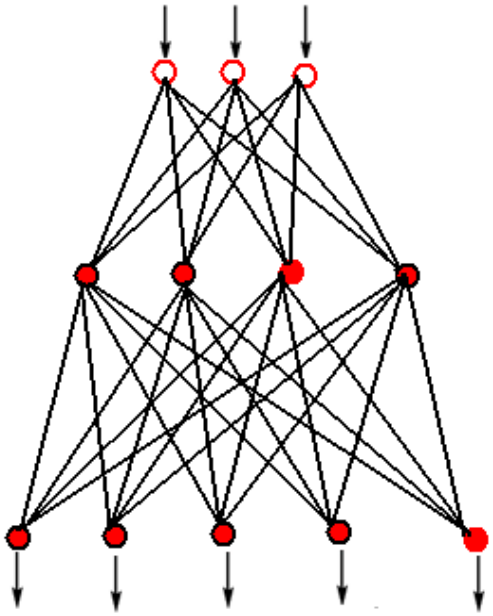


Figure 2.13: A Typical F-F Neural Network

ii) ANN Parameters

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 [21, 25, 26].

Another concept associated with neural networks is the concept of activation function. Activation function is associated with a node in the neural network and it defines the output of that node given an input or set of input. Each neuron is activated with input to the neuron and by its activation function [21].

Each layer in the ANN has an output which is calculated by a function called transfer function based on ANN's net input. Sigmoid function is often used in artificial neural networks to introduce nonlinearity to the network. Without it, the net can only learn functions which are linear combinations of its inputs. The term sigmoid means "S-shaped". Given summed input x, we could have continuous output, S(x), according to the sigmoid function. The sigmoid equation and its corresponding graph are shown in Equation (13) and Figure 2.14 [21] respectively.

$$S(x) = \frac{1}{1 + e^{-x}} \quad (13)$$

where x is an element of the set of real numbers.

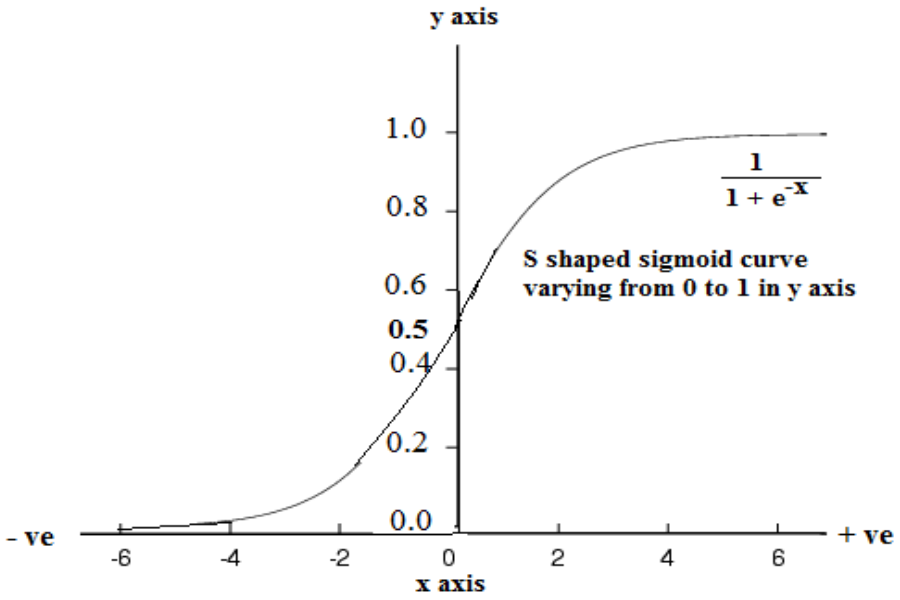


Figure 2.14 : Sigmoid Transfer Function

iii) When to Choose ANN

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 [25].

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 [31].

When compared to other methods, ANNs can tolerate noise better and exhibit low classification error rates [21, 37]. Moreover, compared to statistical methods, ANNs using the B-P network could be easily modified to accommodate more features [21].

A three layered F-F network with sigmoid hidden and softmax output neurons can classify vectors arbitrarily well, given enough neurons in its hidden layer. Softmax is a neural transfer function. Transfer functions calculate a layer's output from its net input. According to [26] there is no reason to use more than one hidden layer [27].

There are two widely used functions for computing neural network error during training, namely, the mean squared error (MSE) and the cross-entropy. Compared to MSE, cross entropy function is proven to accelerate the backpropagation algorithm and to provide good overall network performance [38].

b) Naïve Bayesian Classifiers

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 [11, 20].

Naïve Bayesian classifiers make an assumption known as class conditional independence. This assumption is that the effect of an attribute value on a given class is independent of the values of the other attributes. That is why it is called naïve. To shade some light on Bayesian theorem, let X be a data item. In Bayesian terms, X is considered as "evidence". It is described by measurements made on a set of n attributes. Let H be some hypothesis such that the data item X belongs to a particular class C. During classification problems, we want to determine $P(H|X)$, the probability that the hypothesis H holds given the "evidence" or observed data item X. In simple words, we

are seeking for the probability of data item X belonging to class C. We call *P(H|X),* the posterior probability, or a posteriori probability of H conditioned on X. In addition to the posteriori probability, Bayes' theorem makes use of the priori probability *P(H)*. The priori probability is the probability of a given data item regardless of the values of the data item attributes. Similarly, *P(X|H)* is the posterior probability of X conditioned on *H. P(X)* is the prior probability of X. Based on these probability concepts, Bayes theorem is defined in Equation (14) [20].

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{14}$$

where X is an element of the set of real numbers.

To explain how the naïve Bayesian classifier works better, let us assume D be a training set of data items. Each data item is represented by an *n*-dimensional feature vector, $\boldsymbol{X}$ = (X_1, X_2, X_3, ..., X_n) showing n attribute measurements of the data item. Let us also assume that there are *m* classes, C_1, C_2, $C_{3,...}$,C_m. Now, given data item $\boldsymbol{X}$ the classifier calculates the posterior probability of the data item X for each class. Based on this result, it predicts that $\boldsymbol{X}$ belongs to the class with the highest posterior probability [20].

Since we always have data sets with many attributes, it would be extremely expensive to compute $P(X|C_i)$. Therefore, to reduce this computational cost, the naïve assumption of class-conditional independence is made. This assumes the attributes' values are conditionally independent of one another. Thus, this assumption can be represented by Equation (15) [20].

$$P(X|Ci) = \prod_{k=1}^{n} P(Xk|Ci) \tag{15}$$

where X is an element of the set of real numbers.

CHAPTER THREE: RELATED WORK

3.1 Introduction

Various digital image processing works for grain quality and variety assessment have been reported in the literature. These research works use image analysis for automatic information acquisition on the content and quality of grain samples. They use grain texture, morphology and color features to achieve their goals. Some use morphological features alone while others use a combination of texture, morphology and color features together.

Based on their objectives, research works on cereal grain identification can be broadly classified into two groups. The first group of research works is aimed at assessing the quality of cereal grain sample. The second group consists of those works that aim at identification of cereal grain varieties.

This chapter discusses works that are related to cereal grain quality assessment in general and maize sample quality assessment in particular. The chapter also explains how different our work is from the other related works.

3.2 Grain Varieties Identification

The research works that have been conducted in the area of cereal grain variety identification can be classified into two classes based on the kind of features they used. The first class of researches consists of works that use morphological features alone and the other class of researches uses morphological and color features to achieve better result.

The work in [8] 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 experimental results of this research work claimed that morphological features have more discriminative power to classify coffee 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 coffee increases when the morphological and color features were used together. This research work, however, has the following shortcomings. The first shortcoming is that thresh-holding is manual and fixed and is set to 213. The second 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 coffee beans. Therefore, the color features extracted do not truly represent coffee beans. The third shortcoming of this research work is the lack of any proposed algorithm or model to extract (morphological and color) features from coffee 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 coffee beans.

The goal of the study in [23] 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.

The work in [28] is 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.

The works in [30, 31] identify maize varieties using color and morphological features. The work in [30] proposed a multi-object contour extraction algorithm adapting to maize seeds varieties. Geometric features and color feature parameters of maize seeds were defined and analyzed, and a multi-object geometric features and color features extraction algorithm is realized. The research work claimed to have successfully optimized maize seeds image processing strategies and varieties identification algorithms. Additionally, the work claimed to have improved the precision and speed of maize seeds varieties identification. In this work, maize seeds varieties identification test was performed on four species including Nongda 108, Ludan 981 and Zhengdan 958, and identification accuracy of more than 95% was achieved.

In [31], 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, both works [30, 31] dealt with healthy kernels only and do not address quality factors that describe damaged kernels such as discolored, PD, shriveled, RD and broken.

3.3 Grain Quality Assessment

The work in [5] discusses basics in computer aided image analysis, which are contributing insight of seed morphology and biology, in terms of seed quality and germination and various aspects of seed image analysis like image acquisitions and pattern recognition. Moreover, it discusses about image descriptors, sorting and grading and prediction of kernel weights of seeds in general.

The work in [18] explores the issues involved in developing a relatively low-cost DIA system for the quality assessment of wheat and barley using commonly available equipment. One of the objectives of the research was to develop and test methods based on kernel properties derived from DIA that can be used to assess kernel shape of wheat and barley in order to identify kernel type and cultivars. The other objective of this work was to predict the flour yield of wheat and the soluble hot-water extract of malted barley.

The work in [24] focuses 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 [29] presents an automated grain shape measurement applied to beach sands. It introduces a fully automated routine to measure morphological parameters of sand grains, based on image analysis procedures and mathematical morphology techniques. The authors argue that the results obtained in this work are statistically significant since a large number of sand grains is analyzed and, because they are also independent of the subjectivity introduced by human operators.

The work in [32] proposed computer vision system that could be used to evaluate external physical damage, mold contamination, and floury to-vitreous endosperm ratio in corn and mold contamination in soybeans. The work defines chipped kernel as a

kernel that lost part of pericarp and vitreous endosperm exposing white floury endosperm, but retaining the overall kernel shape. The average success rates for detecting broken, chipped, starch-cracked and moldy corn kernels were 100%, 83%, 88% and 84%, respectively. The success rate for detecting moldy soybeans was 80%. However, this work does not cover discolored, shriveled, PD and RD features of maize kernels. In addition to this limitation, this work is based only on color features and does not include morphological ones.

The work in [33] 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 [34] 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.

3.4 Summary

Even though there are several related works carried out on cereal grains in the field of digital image processing, the development of automatic maize quality assessment system using image processing techniques for the recognition of broken, PD, RD, discolored and shriveled and foreign matter has not been attempted so far. The main difference of this work and the related works lays in the respective research objectives. Moreover, the particular grain type each work dealing with, the research category (i.e., if it is grain variety identification or grain quality identification), and the number of

classes used in the classification process additional criteria that sets apart our work with the other related works. The difference between our work and grain variety identification works is that grain variety identification deals with the identification of the different kinds(species) of a particular grain type while grain quality identification (i.e., our work) works deal with the identification of foreign, pest damage, discolored, shriveled, broken and healthy constituents of maize. However, it is important to consider the respective research objectives, the number and type of classes, and the particular grain type used in each research as criteria to distinguish our work and the other grain quality assessment works.

CHAPTER FOUR: DESIGN FOR MAIZE QUALITY ASSESSMENT

4.1 Introduction

The success of every research work relies a great deal on a good design. Likewise, a successful maize sample quality assessment system should first be designed in a way that can model maize sample contents as accurately as possible. This chapter discusses the design of the system, the proposed solution, and the features to be extracted. Special attention is given to the proposed segmentation algorithm this research work has contributed to knowledge.

4.2 The Proposed System Architecture

The proposed system architecture for assessing maize sample consists of four components: preprocessing, segmentation, feature extraction and classification. The preprocessing component removes false regions based on size. In the segmentation component, individual maize 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 maize sample into one of the seven classes. The system architecture for the proposed system is illustrated in Figure 4.1.

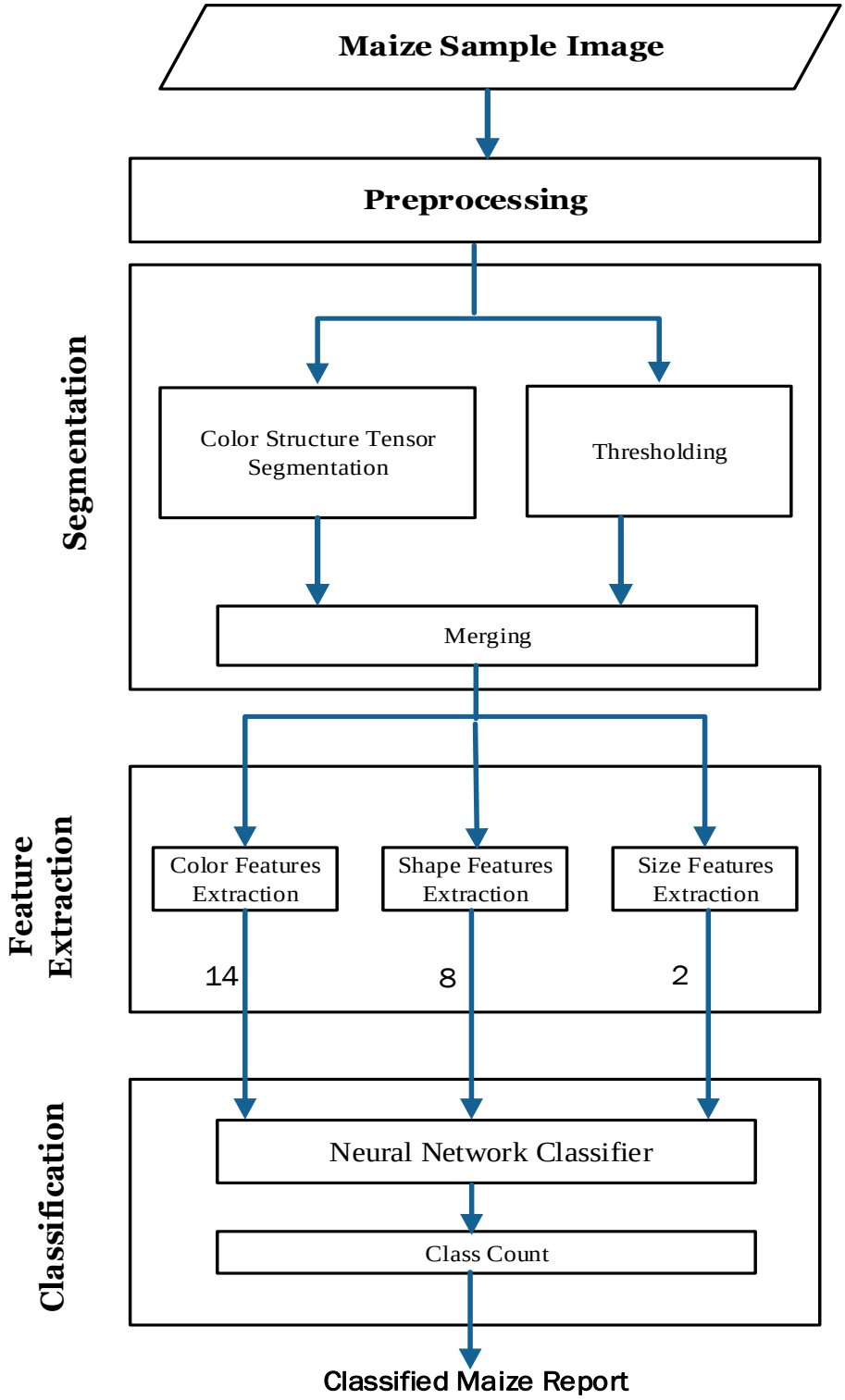


Figure 4.1: The Proposed System Architecture

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 300 pixels. The algorithm for preprocessing relies on the saturation and hue images. False region are 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 algorithm is shown in Algorithm 4.1.

Examples of false regions are shown in Figure 4.2 (a) as the small white regions. The result of the application of the preprocessing algorithm to the image in Figure 4.2 (a) is shown in Figure 4.2 (b).

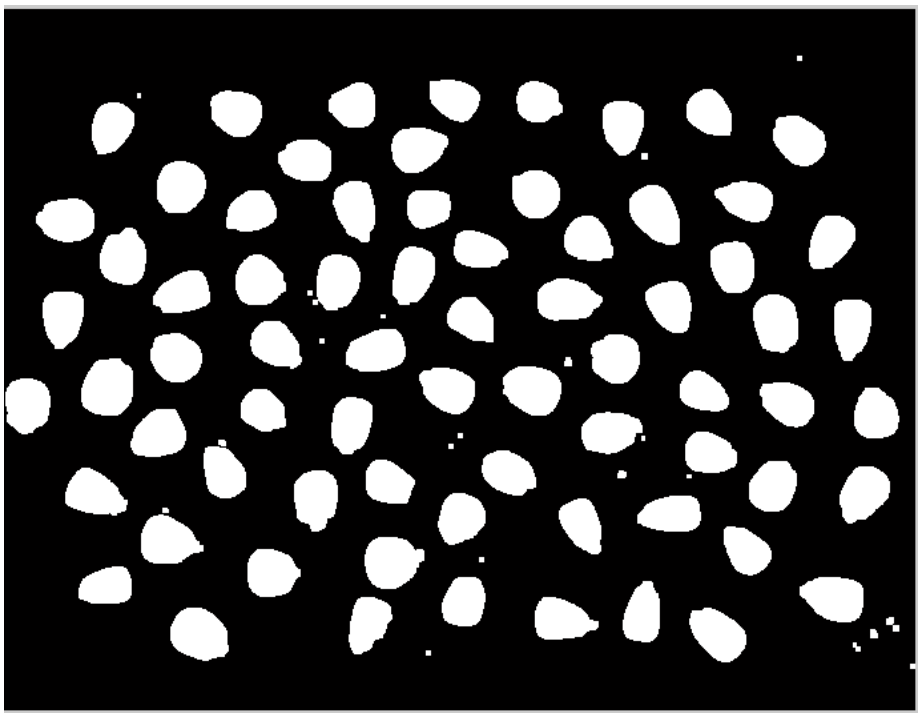
Input: - A maize sample image

Output: - An image free of false regions

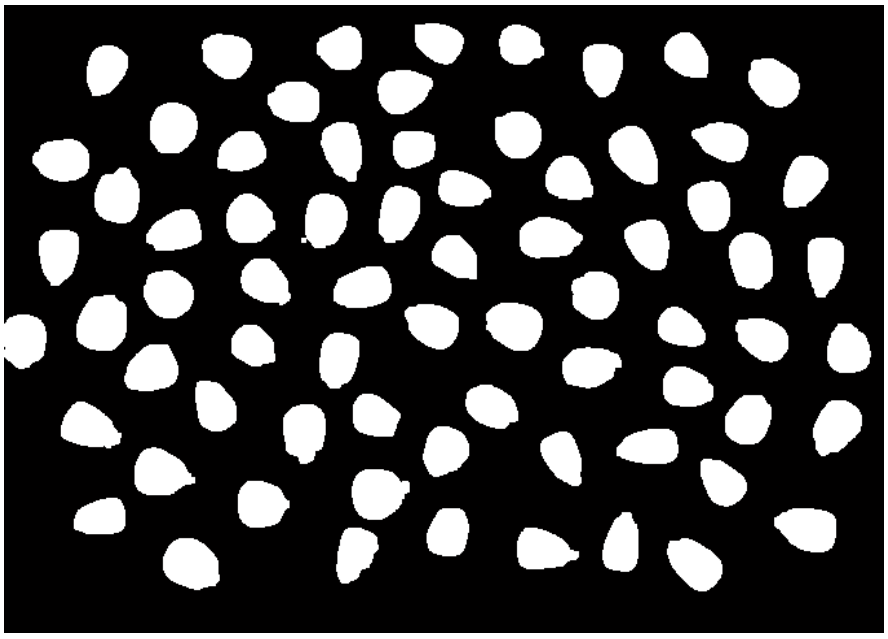
```
Prepare HSV image from the RGB image
Split the HSV image into its Hue, Saturation and Value images
Convert the Saturation image into binary image, bS
Convert the Hue image into binary image, bH
Split the RGB image copy into its RGB components, gR, gG, gB
Prepare binary images, bR, bG, and bB from the RGB components
 For each pixel i in bS
       If (bH (i) == 0)
          bS (i) = 0;
       End
      If (bS (i) == 0)
         bS (i) = 255;
       Else
         bS (i) = 0;
      End
 End
 Label bS;
 For each region i in bS
      Compute area of i;
       If (area of i < 300)
             For each pixel j in i
                 bS(j)=0;
            End
        End
  End
 Label bS;

 For each pixel i in bS
       If (bS (i) ==0)
             gR(i)= 0; gG(i) = 0; gB(j) = 0;
        End
 End
 Combine gR, gG and gB to form false region free image f
 Return f;
```

Algorithm 4.1: Image Preprocessing (False Region Removal)



(a)



(b)

Figure 4.2: (a) False Regions Shown as Tiny White Spots in the Background (b) Maize Kernels after False Region Removal

4.4 Segmentation

The segmentation component of our proposed architecture is responsible for carrying out the work of separating maize sample constituents from each other and from the background of the image. This component contains three sub sub-components, namely, color structure tensor segmentation, thresholding and merging. Color structure tensor helps to change a copy of the output of the preprocessing component into an image free of shadows, shades and specularities. Moreover, it helps to outline the boundary of each sample constituent. The purpose of the thresholding sub-component is to segment copies of each of the three RGB components it receives from the preprocessing component using the thresholding technique. Moreover, 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 a hybrid of color structure tensor segmentation and thresholding algorithms. It is based on the results of the color structure tensor and thresholding segmentation techniques. To the best of our knowledge, this algorithm is novel and has never been used in other applications. The proposed segmentation algorithm is shown in Algorithm 4.2.

Input: - The preprocessed image P, and the Preprocessed Saturation image bS

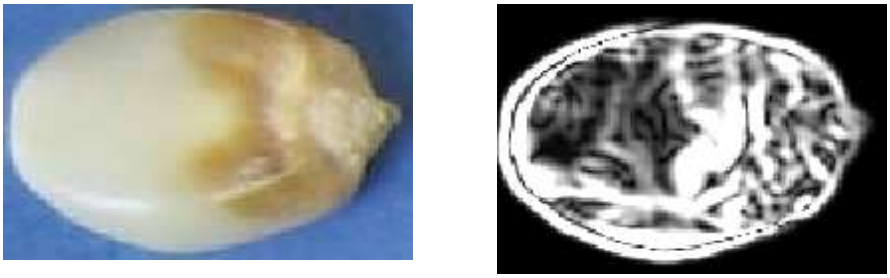
Output: - A segmented an image ready for feature extraction

```
Convert the RGB components into binary images bR, bG and bB
Segment P using Color Structure Tensor Segmentation Algorithm to get T
For each pixel i in P
    If (bR(i)==1 and bG(i) ==0 )
      bS(i)=0;
    End
    If (bR(i)==1and bB(i) ==0 )
      bS(i)=0;
    End
End
For each pixel i in bS
    If (bS(i) >=1 )
      T(i)= bS(i);
    Else If (bS(i)==0 and T(i) >0)
      T(i)= 255;
    Else
      T(i)= 0;
    End
 End
   Label T;
   Make a merged image with size equal to the size of bS and having all the
   pixel values equal to zero, Z
For each pixel i in bS
    If (T(i) ==0 )
     Z(i) = 0;
    Else If (bS(i)==0 and T(i) > 0)
     Z(i)= 255;
   End
 End
Return Z;
```

Algorithm 4.2: The Proposed Segmentation Algorithm

4.4.1 Segmentation Using Color Structure Tensor Algorithm

It is possible to identify the structure (boundary) of maize kernels without being affected by specularity and shadow using color structure tensor based segmentation technique. However, color structure tensor has its own limitation when applied to segment maize images. It does not enable us to demarcate special regions of discoloration, pest damage and rottenness that are found within maize kernels. Instead of identifying these special regions, the technique highlights both special and non-special regions in a mixed way. As a result, damaged areas (special regions) within each kernel are obscured and cannot separately be identified. This limitation of this segmentation technique is exemplified diagrammatically as shown in Figure 4.3. The image in Figure 4.3(a) is the original image of a discolored maize kernel. As can be seen in this image, the kernel has a special region of discoloration that extends from the tip of the kernel down to a distance of half its length.



(a) (b)

Figure 4.3: (a) Original Image of a Discolored Maize Kernel (b) Color Structure Tensor Segmented Image of the Image in (a)

The rest part of the kernel is healthy and does not consist of special regions. The segmentation of this maize kernel using color structure tensor is shown in Figure 4.3(b). From this figure, we can see that both the healthy part of the kernel and the discolored part are highlighted and hence, we cannot distinguish these two parts of the kernels apart. In short, the discolored part seems to be part of the healthy part and vice versa. More examples of this limitation of color structure tensor based segmentation technique are shown in Figure 4.4 and Table 4.1.

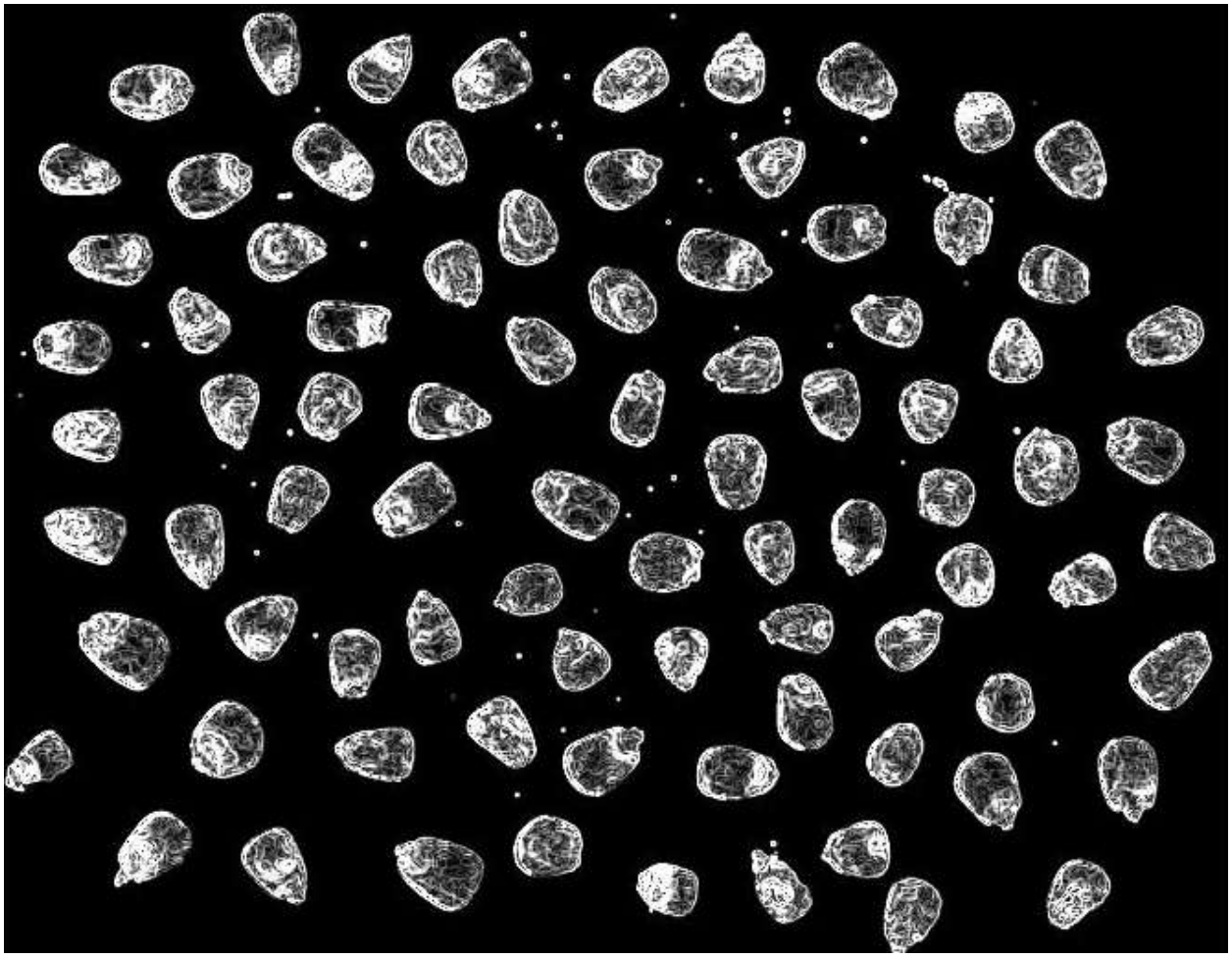
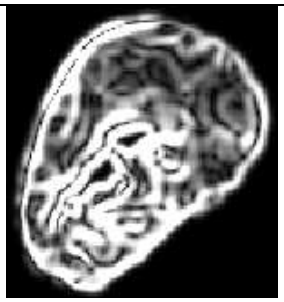
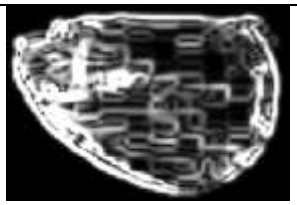


Figure 4.4: Color Structure Tensor Segmented Image Showing Both Discolored and Non Discolored Parts Highlighted

Table 4.1: Maize Images and Their Color Structure Tensor Segmented Images

Original Kernel Image	Image Description	Specularity Invariant Image
	RD	
	PD	
	Discolored	
	Healthy	

4.4.2 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 maize grains are whitish in color, parts of the maize 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 maize 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 maize grains information are lost to the background, the healthy grains appear to be broken and assume different shape, area and size values than their 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. Unlike thresholding, segmentation using color structure tensor does not result in loss of information to the background of the image. Moreover, it inhibits specularity and shadows that could possibly be introduced at the image capturing phase. The effect of shadows, during thresholding, is shown in Figure 4.5. In Figure 4.5(a), we can see that some kernels have casted their shadows. This shadow is replicated in the threshold image as shown in Figure 4.5(b). However, this shadow is not present in the color structure tensor segmented image as shown in Figure 4.5(c).

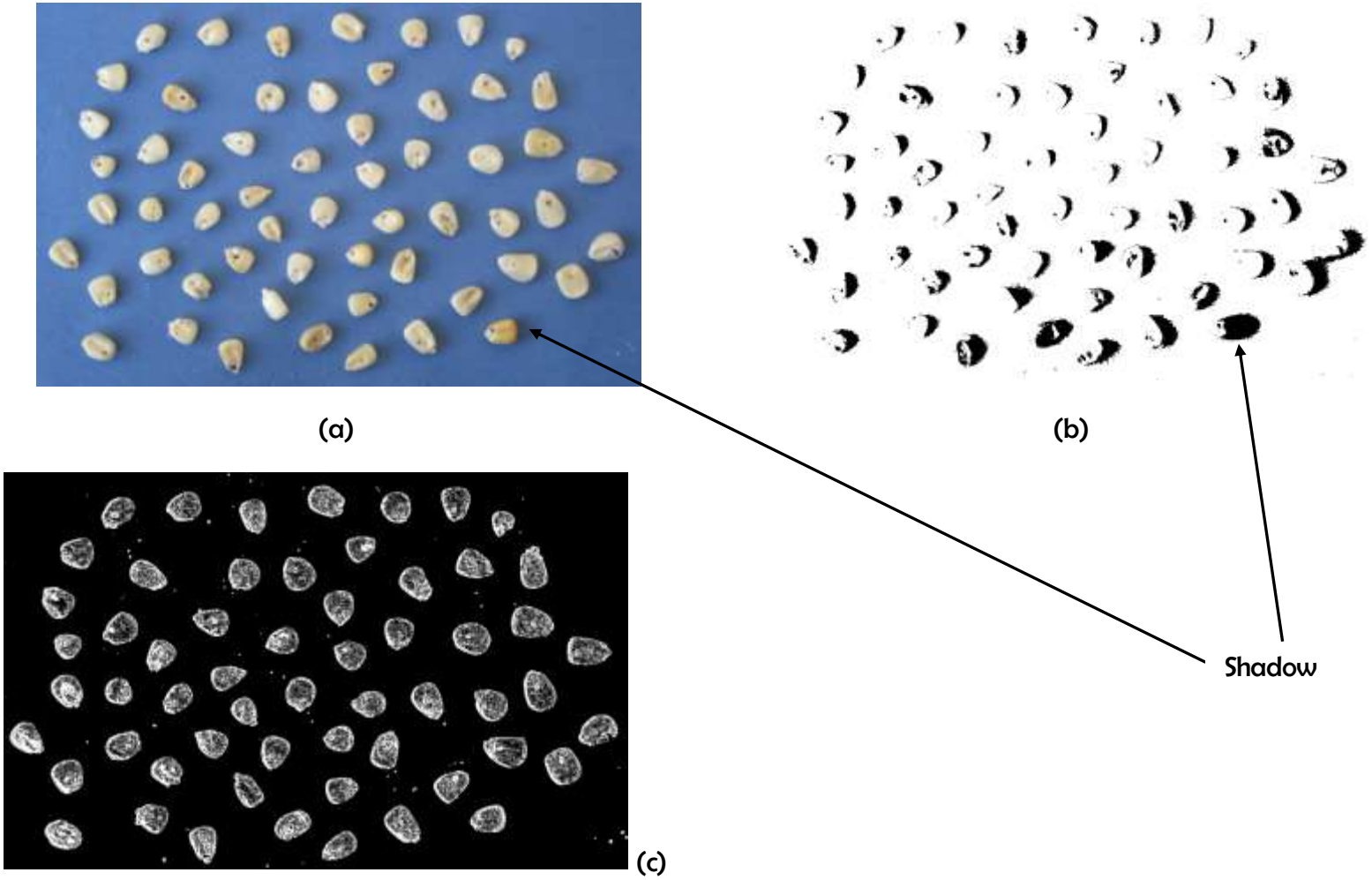
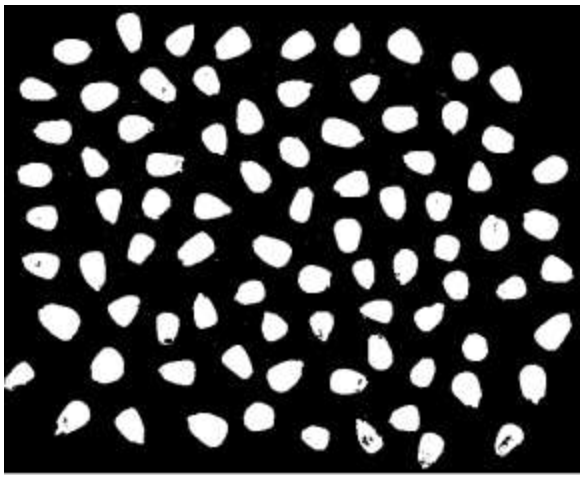


Figure 4.5: (A) Original Image (B) Result of Thresholding (C) Result of Color Structure Tensor Segmentation

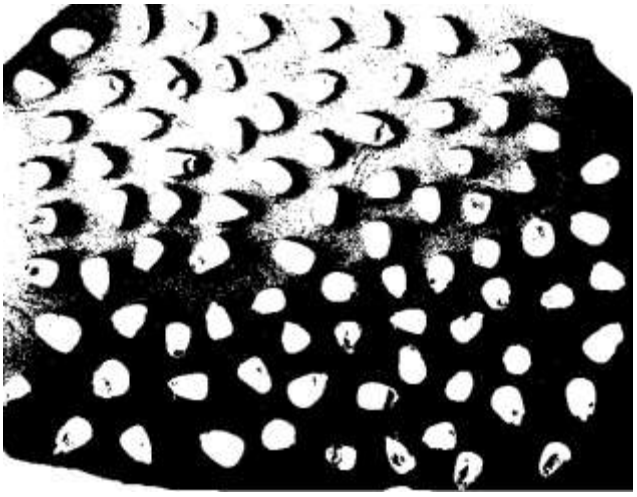
4.4.3 Merging Segmented Images

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 maize 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 4.7.

(a)



(b)



(c)

(d)

Figure 4.6: Discolored Maize 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

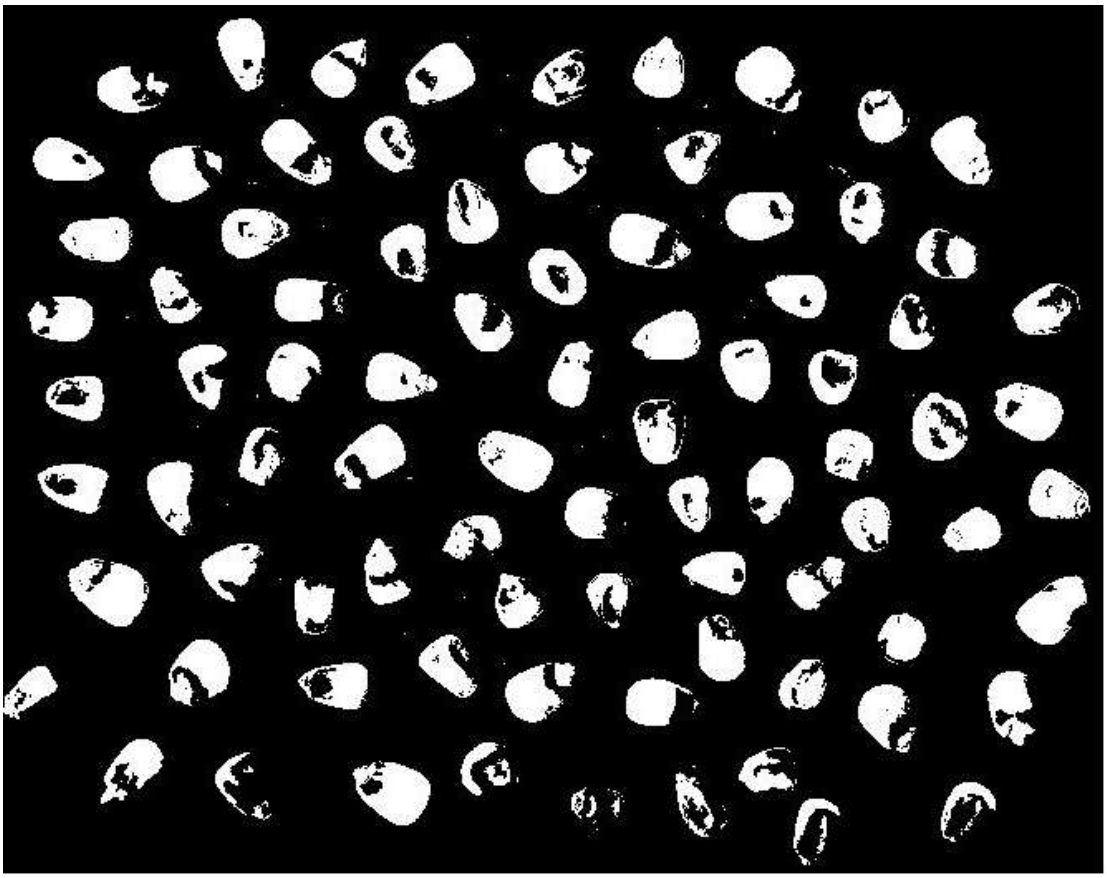


Figure 4.7: Reconstructed Image of the Red, the Green and the Blue Binary Images

In light of this, we combine reconstructed image and the color structure tensor segmented image to form an image consisting complete information of the location of pest damage, discoloration and rottenness of maize kernels. We call this image as ***merged image***. The merged image for the images in Figure 4.6(b), Figure 4.6(c) and Figure 4.6(d) is shown in Figure 4.8.

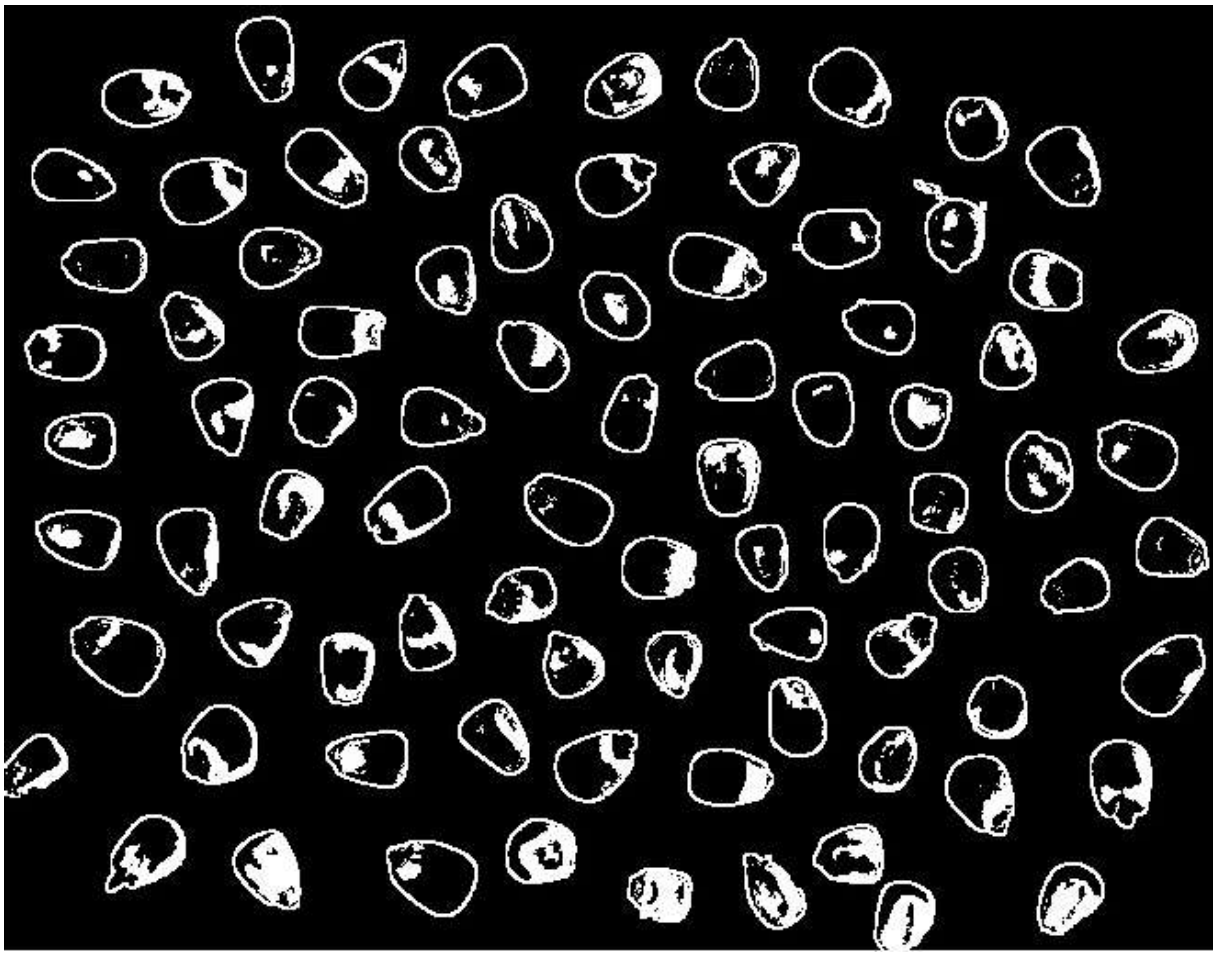
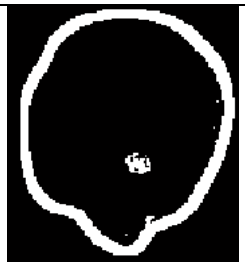
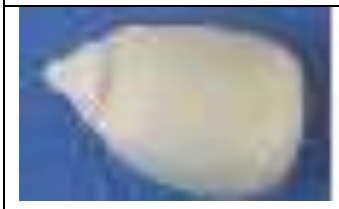
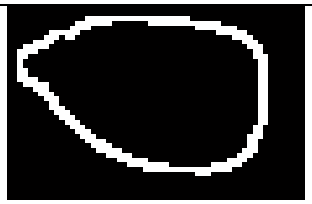


Figure 4.8: Merged Image of Discolored Kernels

In order to present the fruits of our proposed segmentation algorithm, we have shown examples of individual maize grain images against their corresponding merged image as shown in Table 4.2.

Table 4.2: Examples of Maize Kernels and Their Corresponding Merged Images

kernel Image	Kernel Description	Kernel Merged image
	PD	
	PD	
	RD	
	Discolored	
	Discolored	
	Healthy	

4.5 Feature Extraction

Feature extraction component is responsible for extracting the descriptive features of the maize 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 maize 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 maize sample is computationally inefficient. Thus, representative features of maize sample constituents are selected and extracted. The features extracted from maize images can be grouped into three categories, namely, color, size, and shape. To describe maize sample constituents, we identified 14 color features, 2 size features, and 8 shape features.

4.5.1 Color Feature Extraction

Color is one of the visual attributes of maize kernels. It helps distinguish among certain classes of maize sample. For instance, the categories of RD and discolored maize kernels can be satisfactorily distinguished by using their color features. Moreover, color has the ability to isolate foreign matters from among maize kernels.

Fourteen color features have been selected to represent the color features of maize 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). 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].

$$\boldsymbol{R} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \boldsymbol{r}(\boldsymbol{x}, \boldsymbol{y}) \tag{16}$$

$$\boldsymbol{G} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}) \tag{17}$$

$$\boldsymbol{B} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \mathrm{b}(\boldsymbol{x}, \boldsymbol{y}) \tag{18}$$

where $\boldsymbol{x}$, $\boldsymbol{y}$, k , and $\boldsymbol{N}$ 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 maize kernel. These features are calculated using Equation (16) based on the area of the damaged region within a maize kernel. These features are discussed in Chapter Five. The screenshot of the numerical values of the 14 color features corresponding to 12 healthy maize 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.

Table 4.3: Screenshot Showing the Values of 8 Color Features for 12 Healthy Maize Kernels

	Red	Greeen	Blue	AverageRGB	Hue	Saturation	Intensity	AverageHSV	SpotH	SpotS	SpotV	SpotR	SpotG	SpotB
1	10360	0.7847	0.9439	0.9044	2.6331	0.3327	0.1565	0.7955	1.2847	0.5777	0.4466	0.6811	0.0317	0.5701
2	10207	0.7878	0.9658	0.8949	2.6485	0.3550	0.1458	0.8127	1.3135	0.5941	0.4772	0.6879	0.0023	0.5058
3	9586	0.7789	0.8834	0.8873	2.5497	0.3045	0.2140	0.7720	1.2905	0.4902	0.3981	0.6398	0.2093	0.5970
4	10548	0.7651	0.8759	0.8545	2.4954	0.2721	0.2229	0.7045	1.1995	0.4963	0.4108	0.6233	0.1894	0.4979
5	8345	0.7643	0.9225	0.8090	2.4957	0.3120	0.1779	0.7507	1.2406	0.5318	0.4399	0.6157	0.1207	0.2874
6	12394	0.8084	0.9399	0.8616	2.6099	0.2832	0.1730	0.7739	1.2301	0.5842	0.4482	0.6181	0.0238	0.2951
7	8684	0.7645	0.9119	0.8688	2.5453	0.3063	0.1827	0.8103	1.2992	0.5509	0.4271	0.6493	0.0935	0.4951
8	10634	0.7910	0.9480	0.8710	2.6100	0.2699	0.1629	0.7938	1.2265	0.5766	0.4412	0.6384	0.0190	0.3943
9	10609	0.7887	0.9541	0.8325	2.5753	0.3066	0.1474	0.7691	1.2231	0.5885	0.4584	0.6210	0.0110	0.2161
10	11336	0.7843	0.8543	0.8117	2.4502	0.2544	0.2213	0.7119	1.1876	0.4896	0.3873	0.5721	0.2146	0.3142
11	10063	0.7854	0.9361	0.8145	2.5359	0.2791	0.1894	0.7744	1.2429	0.5883	0.4408	0.5967	0.0146	0.1483
12	9344	0.7690	0.9576	0.8191	2.5458	0.2776	0.1913	0.7699	1.2388	0.5849	0.4566	0.6397	0.0164	0.2297

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].

$$\boldsymbol{H} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}) \tag{19}$$

$$\boldsymbol{S} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \mathrm{s}(\boldsymbol{x}, \boldsymbol{y}) \tag{20}$$

$$\boldsymbol{V} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum_{k=1}^{\mathrm{N}} \mathrm{v}(\boldsymbol{x}, \boldsymbol{y}) \tag{21}$$

where $\boldsymbol{x}$, $\boldsymbol{y}$, k , and $\boldsymbol{N}$ are positive integers.

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 maize kernel. These features are calculated using Equations (16) , (17), and (18)based on the area of the damaged region within a maize kernel. These six spot features of maize kernels are very helpful in the process of distinguishing healthy and damaged kernel. The values of these features for 10 healthy and 10 RD kernels are given in Table 4.4.

Table 4.4: Color Feature Values for Samples of Healthy and RD Kernels

Healthy						Rotten and Diseased					
Spot-Hue	Spot-Saturation	Spot-Value	Spot-Red	Spot-Green	Spot-Blue	Spot Hue	Spot-Saturation	Spot-Value	Spot-Red	Spot-Green	Spot-Blue
1.28	0.58	0.45	0.68	0.03	0.57	1.15	0.37	0.31	0.61	0.38	0.64
1.31	0.59	0.48	0.69	0.00	0.51	1.18	0.38	0.36	0.62	0.37	0.75
1.29	0.49	0.40	0.64	0.21	0.60	1.18	0.32	0.34	0.60	0.38	0.59
1.20	0.50	0.41	0.62	0.19	0.50	1.14	0.31	0.32	0.59	0.47	0.60
1.24	0.53	0.44	0.62	0.12	0.29	1.15	0.40	0.34	0.63	0.35	0.73
1.23	0.58	0.45	0.62	0.02	0.30	1.20	0.51	0.37	0.65	0.09	0.75
1.30	0.55	0.43	0.65	0.09	0.50	1.19	0.44	0.36	0.61	0.26	0.62
1.23	0.58	0.44	0.64	0.02	0.39	1.16	0.35	0.29	0.61	0.43	0.75
1.22	0.59	0.46	0.62	0.01	0.22	1.14	0.30	0.36	0.58	0.41	0.48
1.19	0.49	0.39	0.57	0.21	0.31	1.15	0.44	0.33	0.61	0.16	0.66

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 maize 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 Algorithm 4.3.

Input: - Merged image, mI

Output: - 14 color features

```
Declare feature vector V;

For each region R in mI
        Compute the region area;
        Compute the average red value;
        Compute the average green;
        Compute the average blue;
        Compute the average hue;
        Compute the average saturation;
        Compute the average intensity;
        Compute the average RGB;
        Compute the average HSV;

      For  each pixel i in R

            If (R(i)==255 and R(i) ==i )
                  Accumulate the Spot-Red value;
                  Accumulate the Spot-Green value;
                  Accumulate the Spot-Blue value;
                  Accumulate the Spot-Hue value;
                  Accumulate the Spot-Saturation value;
                  Accumulate the Spot-Hue color value;

            End

       End

      Add features into V;

  End

Return V;
```

Algorithm 4.3: Color Features Extraction Algorithm

4.5.2 Size Feature Extraction

This research work uses *two* size features, namely, *area* and *perimeter* features to determine the size of maize 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 Algorithm 4.4.

Input: - Merged image, mI

Output: - 2 size features

```
Declare feature vector V;
For each region R i in mI
        Compute area;
        Compute perimeter;
        Add features into V;
End
Return V;
```

Algorithm 4.4: Size Features Extraction Algorithm

Sample data of area and perimeter features extracted from 12 discolored maize kernels are shown in Table 4.5.

Table 4.5: Screenshot Showing the Values of 2 Size Features for 12 Discolored Maize Kernels

	Area	Perimeter
1	0.4786	134
2	0.4886	179
3	0.4019	132
4	0.3227	120
5	0.4891	150
6	0.6843	153
7	0.5870	133
8	0.3832	133
9	0.3465	164
10	0.5556	115
11	0.5625	143
12	0.4516	105

4.5.3 Shape Feature Extraction

As described in Chapter Two, 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 cab 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 maize kernel is believed to be good for the discrimination between foreign matters and maize grains. Moreover, it is also a helpful tool in the separation process of broken and whole maize kernels. The *eight* shape descriptors identified in this research work are *count of convex hull sides, aspect ratio, ovality, triangularity, convexity, solidity, major axis to area ratio* and *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 maize kernel. The rest of the shape descriptors are explained in Chapter Two. Sample data of these 8 shape features is shown in Table 4.6. 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 maize kernel. Then, we compute the FDs of an unknown object and compare it to the known maize 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 4.6: Screenshot Showing the Values of 8 Shape Features for 12 Healthy Maize Kernels

	ConvexHull	AxisRatio	Ovality	Triangularity	Convexity	Solidity	MajorAxis/area	Fourier
1	0.8340	375.2500	0.9925	0.6415	0.0216	0.9788	0.0122	0.0906
2	0.7474	370.3710	0.9946	0.6401	0.0169	0.9834	0.0130	0.0900
3	0.7109	364.9950	0.9776	0.6512	0.0148	0.9854	0.0138	0.0534
4	0.6748	401.6500	0.9654	0.6594	0.0316	0.9694	0.0136	0.1316
5	0.6592	341.9450	0.9816	0.6486	0.0138	0.9864	0.0154	0.0707
6	0.8666	401.8060	0.9947	0.6400	0.0138	0.9864	0.0109	0.1162
7	0.7311	346.0320	0.9882	0.6442	0.0172	0.9831	0.0142	0.1030
8	0.8430	376.0280	0.9904	0.6428	0.0213	0.9792	0.0120	0.1040
9	0.7987	377.2020	0.9838	0.6471	0.0145	0.9857	0.0124	0.0190
10	0.7939	408.6540	0.9696	0.6566	0.0339	0.9672	0.0121	0.0948
11	0.8964	359.9270	0.9975	0.6382	0.0150	0.9852	0.0119	0.0835
12	0.6746	363.8130	0.9827	0.6478	0.0199	0.9805	0.0143	0.0375

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 12 healthy maize kernels is shown in Table 4.6. Besides, sample of a healthy maize 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 Algorithm 4.5.

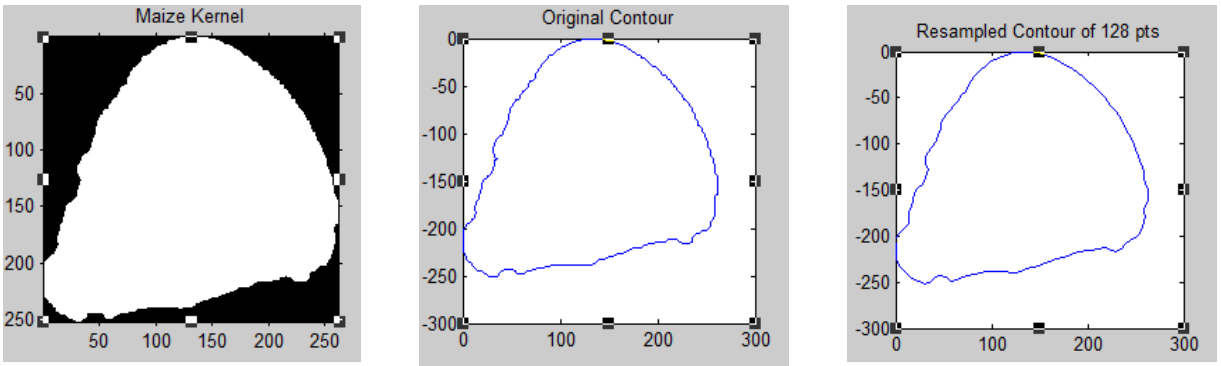


Figure 4.9: (a) Binary Maize Kernel Image (b) Contour of the Original Image (c) Resampled Contour of Length 128

Input: - Merged image, mI

Output: - 8 size features

```
Declare feature vector V;
For each region R i in mI
    Compute the number of convex hull sides;
    Compute minor axis length /Compute major axis length;
    Compute Ovality;
    Compute Triangularity;
    Compute Solidity;
    Compute major axis length/area;
    Compute (convex area - region area)/area;
    Compute Fourier distance;

    Add features into V;
End
Return V;
```

Algorithm 4.5: Shape Features Extraction Algorithm

4.6 Classification

Classification component contains ANN classifier and the class count sub-components. The ANN classifies maize 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 B-P 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 4.10. The proposed classification algorithm is shown in Algorithm 4.6.

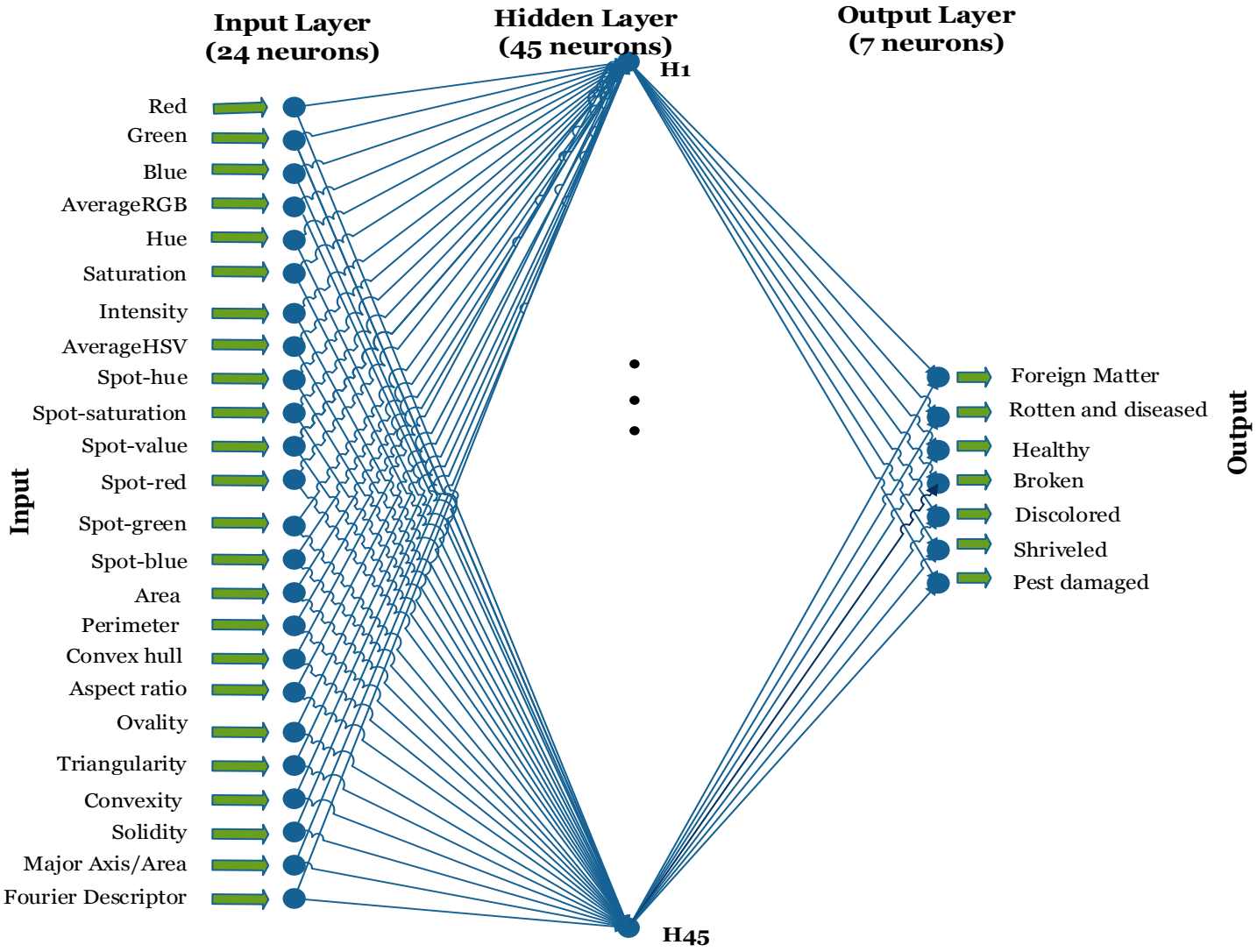


Figure 4.10: Design of ANN Used for the Classification of Maize Sample

Input: - Extracted Features Vector V

Output: - Classified Maize Report

```
C = Call the stand alone ANN (V);
Declare class count array of size 7, A;

 For each row R in matrix C
        Determine the index of the maximum probability, maxPIndex;

       If (maxPIndex ==1 )
            Increment count of Foreign matter, fM;
       Else If(maxPIndex ==2 )
            Increment count of Rotten, rM;
       Else If(maxPIndex ==3 )
            Increment count of Healthy, hM;
       Else If(maxPIndex ==4 )
            Increment count of Broken, bM;
       Else If(maxPIndex ==5 )
            Increment count of Discolored, dM;
       Else If(maxPIndex ==6)
            Increment count of Shriveled, sM;
       Else If(maxPIndex ==7 )
            Increment count of PD, pM;

      End
      A[1]= fM;   A[2]= rM;
      A[3]= hM; A[4]= bM;
      A[5]= dM; A[6]= sM;
      A[7]= pM;

  End

Return A;
```

Algorithm 4.6: Classification Algorithm

4.7 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 will be 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 maize sample constituents. The fourth component, classification, classifies maize 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.

CHAPTER FIVE: EXPERIMENT

5.1 Introduction

In this chapter, we discuss the experiments 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 will be discussed. Alongside these, the discriminative power of color, size, and shape will be tested and compared.

5.2 Data Set

A total of 534 of maize kernels and foreign matter are prepared to train, validate and test the proposed model. These 534 maize sample constituents are separated into their corresponding 7 classes based on their characteristics. Therefore, we finally have 7 images each corresponding to each of the 7 classes. The data were partitioned *randomly* into training, validation and test sets. Image acquisition is done using a Cannon camera (Model SD630). The images taken are all 24 bit color JPEG format. The number of maize 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 15 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 light blue background is used. The samples of maize 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 maize 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	31
RD	0000010	83
Healthy	0000100	71
Broken	0001000	98
Discolored	0010000	67
Shriveled	0100000	126
PD	1000000	58
Total		**534**

5.3 Implementation

MATLAB version R2014a 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 i3 laptop computer with 4GB RAM and 2.3 GHz processor. The graphical user interface of the developed prototype is shown in Figure 5.1.

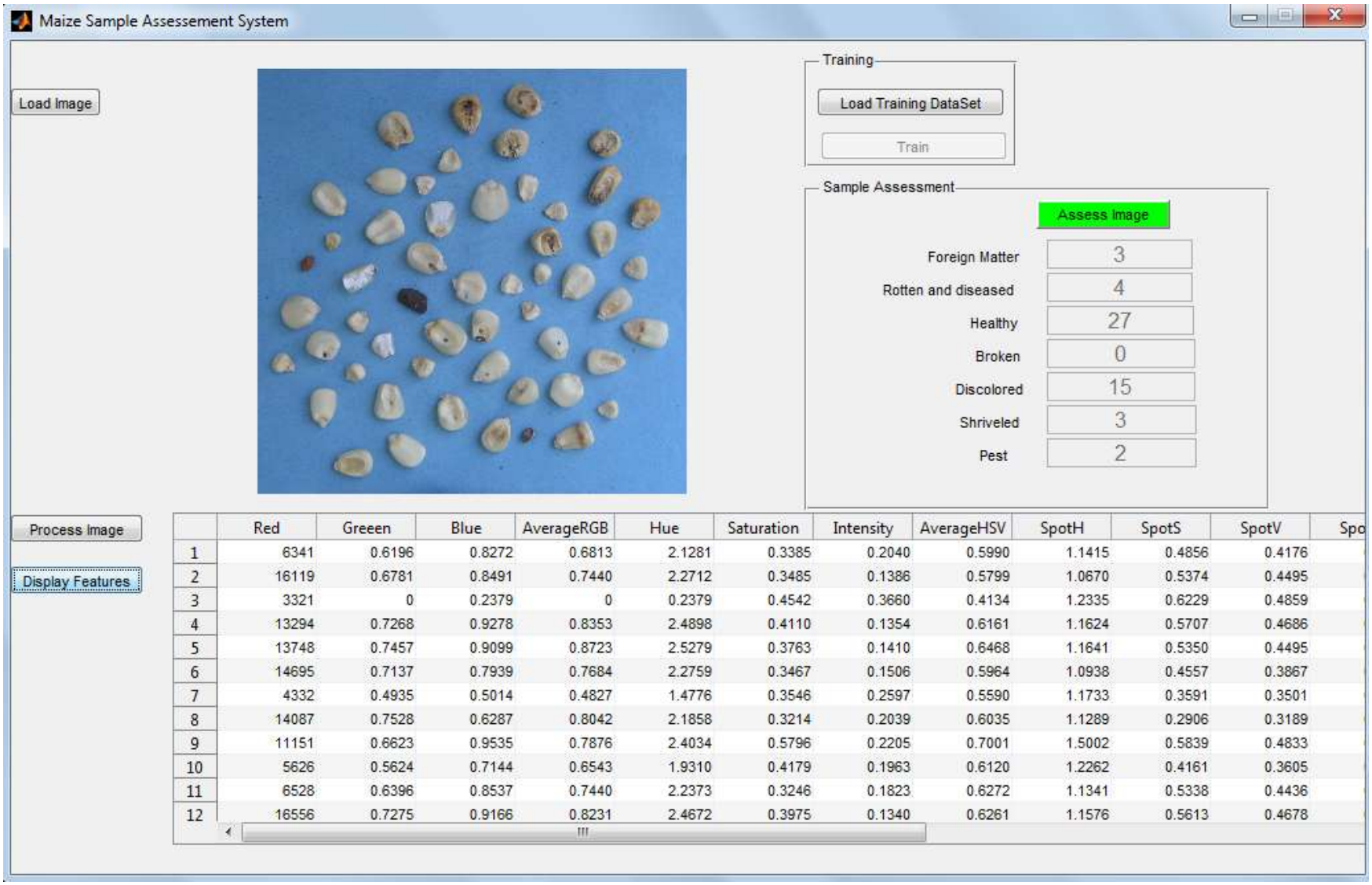


Figure 5.1: Screenshot of the User Interface of the Developed Prototype

5.4 Test Results

Tests are conducted both on naïve Bayesian 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 161(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 61.9%, 100%, 96.7%, 100%, 68.0%, 100%, and 65.4% respectively.

Table 5.2: Test Confusion Matrix of Naïve Bayesian Classifier

Target Class

Output Class	Rotten and Diseased	Shriveled	Broken	Discolored	Healthy	Foreign	Pest Damaged
RD	13	0	1	0	1	0	8
Shriveled	4	34	0	0	3	0	0
Broken	0	0	29	0	0	0	0
Rotten and Diseased	1	0	0	16	1	0	0
Healthy	1	0	0	0	17	0	1
Foreign	0	0	0	0	0	9	0
Pest Damaged	2	0	0	0	3	0	17
Classification Accuracy	61.9%	100%	96.7%	100.0%	68.0%	100.0%	65.4%

The overall classification accuracy obtained is 83.9%. 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 (161). 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 46th iteration (epoch) at which the validation error started to rise and the training error was dropping. This training process is shown Figure 5.2.

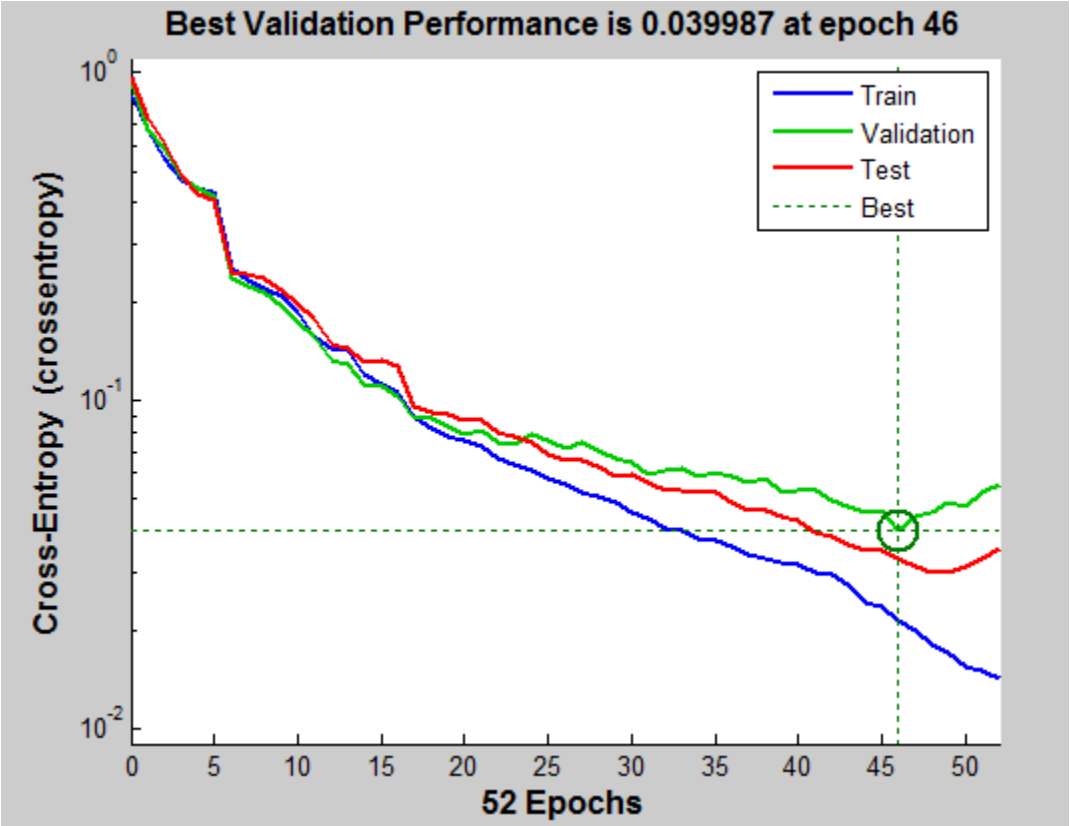


Figure 5.2: Cross-Entropy Error Showing the Performance of the Trained ANN

Accordingly, classification accuracies of 98.7%, 95.0%, and 96.3% have been achieved for training, validation and testing respectively. Moreover, an overall classification accuracy of 97.8% is achieved. This accuracy is calculated by dividing the total number of correctly classified kernels by 534 (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.

Table 5.3: Confusion Matrix Showing Overall Classification Accuracy

Output class									
	Foreign Matter	**31** **5.8%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**100** **0.0%**
	Rotten and Diseased	**0** **0.0%**	**79** **14.8%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**2** **0.4%**	**2** **0.4%**	**95.2%** **4.8%**
	Healthy	**0** **0.0%**	**1** **0.2%**	**70** **13.1%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**98.6 %** **1.4%**
	Broken	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**96** **18.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**100%** **0.0%**
	Discolored	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**2** **0.4%**	**67** **12.5 %**	**0** **0.0%**	**0** **0.0%**	**97.1%** **2.9%**
	Shriveled	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**124** **23.2%**	**1** **0.2%**	**99.2%** **0.8%**
	Pest Damaged	**0** **0.0%**	**3** **0.6%**	**1** **0.2%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**55** **10.3%**	**93.2%** **6.8%**
		100 % **0.0%**	**95.2 %** **4.8%**	**98.6 %** **1.4%**	**98.8 %** **2.0%**	**100%** **0%**	**98.4 %** **1.6%**	**94.8%** **5.2%**	**97.8%** **2.2%**
		Foreign	Rotten and Diseased	Healthy	Broken	Discolored	Shriveled	Pest Damaged	
Target class									

Since the naïve Bayesian classifier resulted in 83.9% of classification accuracy and the ANN achieved 97.8% for the same, we conclude that ANN out performs naïve Bayesian classifier. To identify which feature groups contribute the most in the classification process, we prepared four different scenarios and trained the network under each.

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 88.6% as shown in Table 5.4. RD class was the most impacted class by absence of these features attaining only 67.5% of classification accuracy. To put this in actual numbers, out of the total 83 RD kernels, 16 were misclassified as PD, 4 as shriveled, 4 as healthy, 2 as broken and 1 as discolored. Only 56 kernels were correctly classified as RD.

The second most impacted class in this scenario is the class PD which attained a classification accuracy of 72.4%. This means out of the total 58 discolored kernels, 8 were misclassified as healthy, 6 as RD, 1 as discolored, 1 as shriveled. Only 42 were correctly classified as PD.

Table 5.4: Confusion Matrix for Scenario-1

Output Class									
	Foreign Matter	**31** **5.8%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**100** **0.0%**
	RD	**0** **0.0%**	**56** **10.5%**	**0** **0.0%**	**0** **0.0%**	**2** **0.4%**	**2** **0.4%**	**2** **0.4%**	**84.8%** **15.2%**
	Healthy	**0** **0.0%**	**4** **0.7%**	**63** **11.8%**	**0** **0.0%**	**2** **0.4%**	**1** **0.2%**	**0** **0.0%**	**80.8%** **19.2%**
	Broken	**0** **0.0%**	**2** **0.4%**	**0** **0.0%**	**96** **18.0%**	**1** **0.2%**	**0** **0.0%**	**0** **0.0%**	**97.0%** **3.0%**
	Discolored	**0** **0.0%**	**1** **0.2%**	**4** **0.7%**	**2** **0.4%**	**62** **11.6%**	**0** **0.0%**	**0** **0.0%**	**88.6%** **11.4%**
	Shriveled	**0** **0.0%**	**4** **0.7%**	**2** **0.4%**	**0** **0.0%**	**0** **0.0%**	**123** **23.0%**	**1** **0.2%**	**94.6%** **5.4%**
	PD	**0** **0.0%**	**16** **3.0%**	**2** **0.4%**	**0** **0.0%**	**0** **0.0%**	**0** **0.0%**	**42** **7.9%**	**70.0%** **30.0%**
		100% **0.0%**	**67.5%** **32.5%**	**88.7%** **11.3%**	**98.0%** **2.0%**	**92.5** **7.5%**	**97.6%** **2.4%**	**72.4%** **27.6%**	**88.6%** **11.4%**
		Foreign	RD	Healthy	Broken	Discolored	Shriveled	PD	
Target class									

The third most impacted classes in this scenario are the class healthy and the class discolored. Healthy class attained classification accuracy of 88.7% while discolored attained 92.5%. Out of the total 71 healthy kernels, 4 were misclassified as discolored, 2 as shriveled and 2 as PD.

Only 63 kernels were correctly classified as healthy. In the meantime, the class discolored has attained a classification accuracy of 92.5%. Out of 67 discolored kernels, 2 were misclassified as healthy, 2 as RD and 1 as broken. Minor misclassification has also occurred in shriveled class with classification accuracies of 97.6%. The actual classification numbers for these classes can be found on Table 5.4.

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 maize kernels with the intention to discriminate between shriveled and other kernels. Therefore, in this scenario, we expected shriveled kernels to be misclassified into other classes. However, the accuracy of the classifier was observed to reduce from 97.8% to 94.6%.

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 97.8% to 96.6%.

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 97.8% to 62.0% and its classification error rose to 38.0%.

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 5 minutes to identify and count foreign, rotten and diseased, healthy, broken, discolored, shriveled 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 5.3, Figure 5.4, and Figure 5.5 respectively.

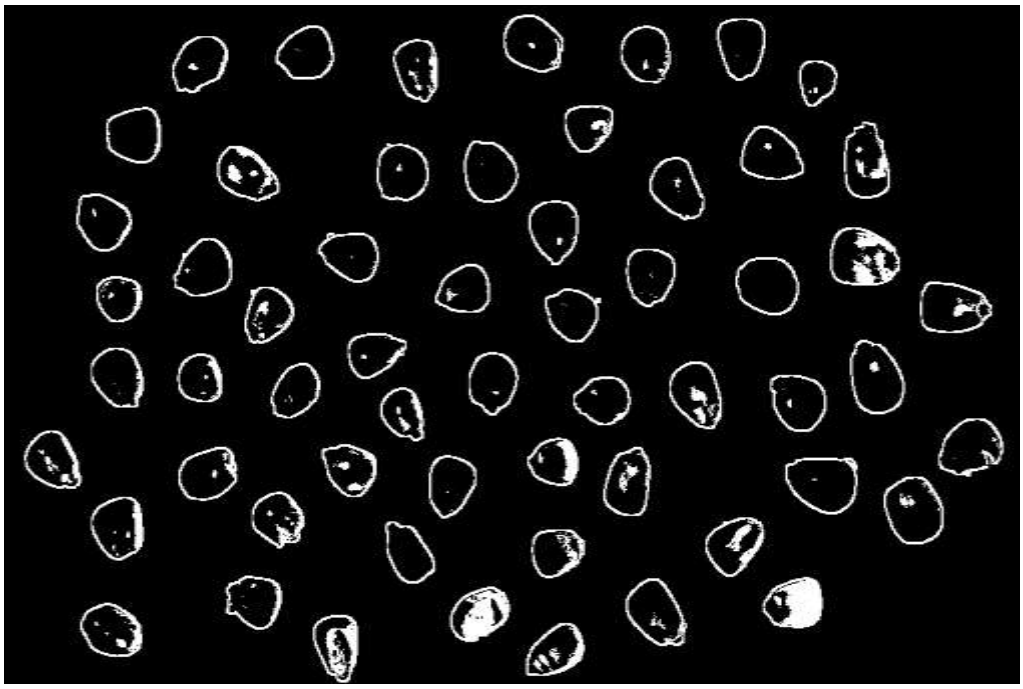


Figure 5.3: Spots Formed by PD

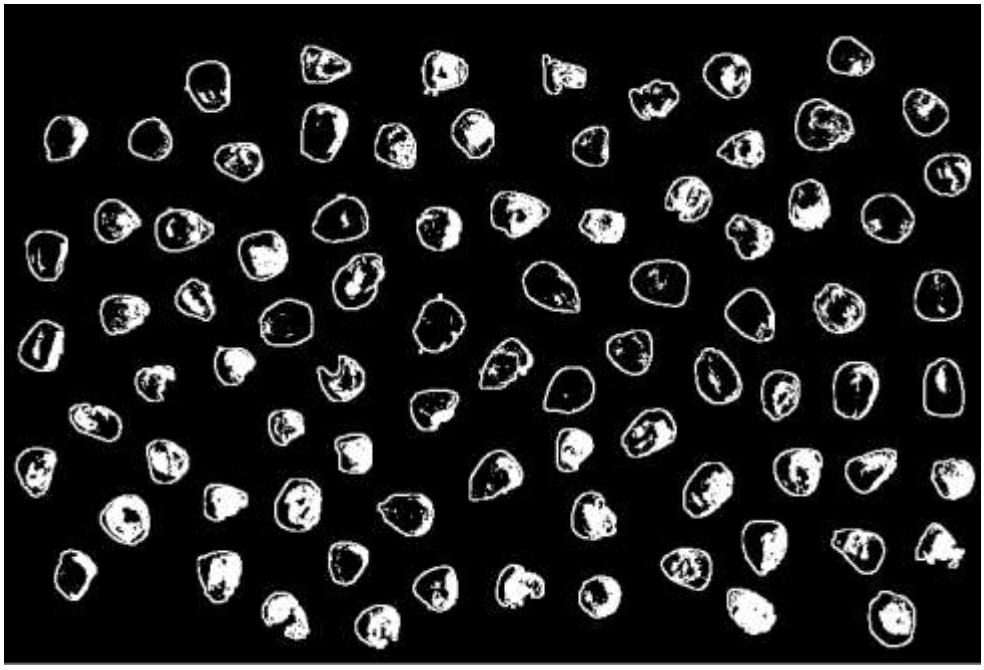


Figure 5.4: Spots Formed by Rottenness

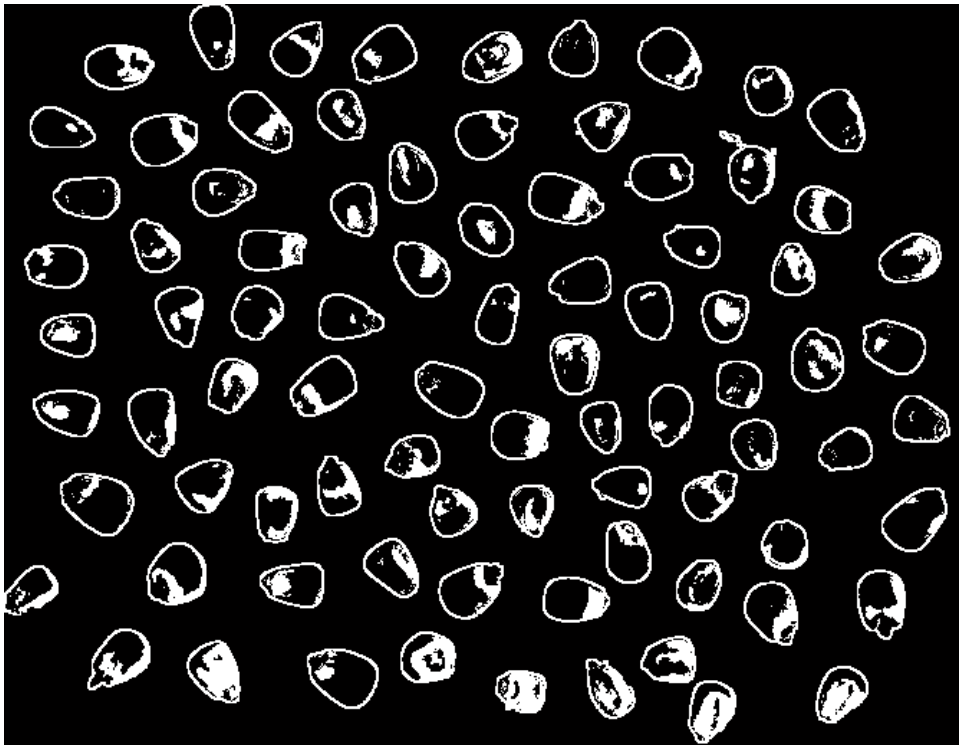


Figure 5.5: Spots Formed by Discoloration

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 location. 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 algorithm shown in Algorithm 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 (97.6%) 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 maize sample. The comparison of discriminative power of size, color and shape features is presented in Figure 5.6 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.

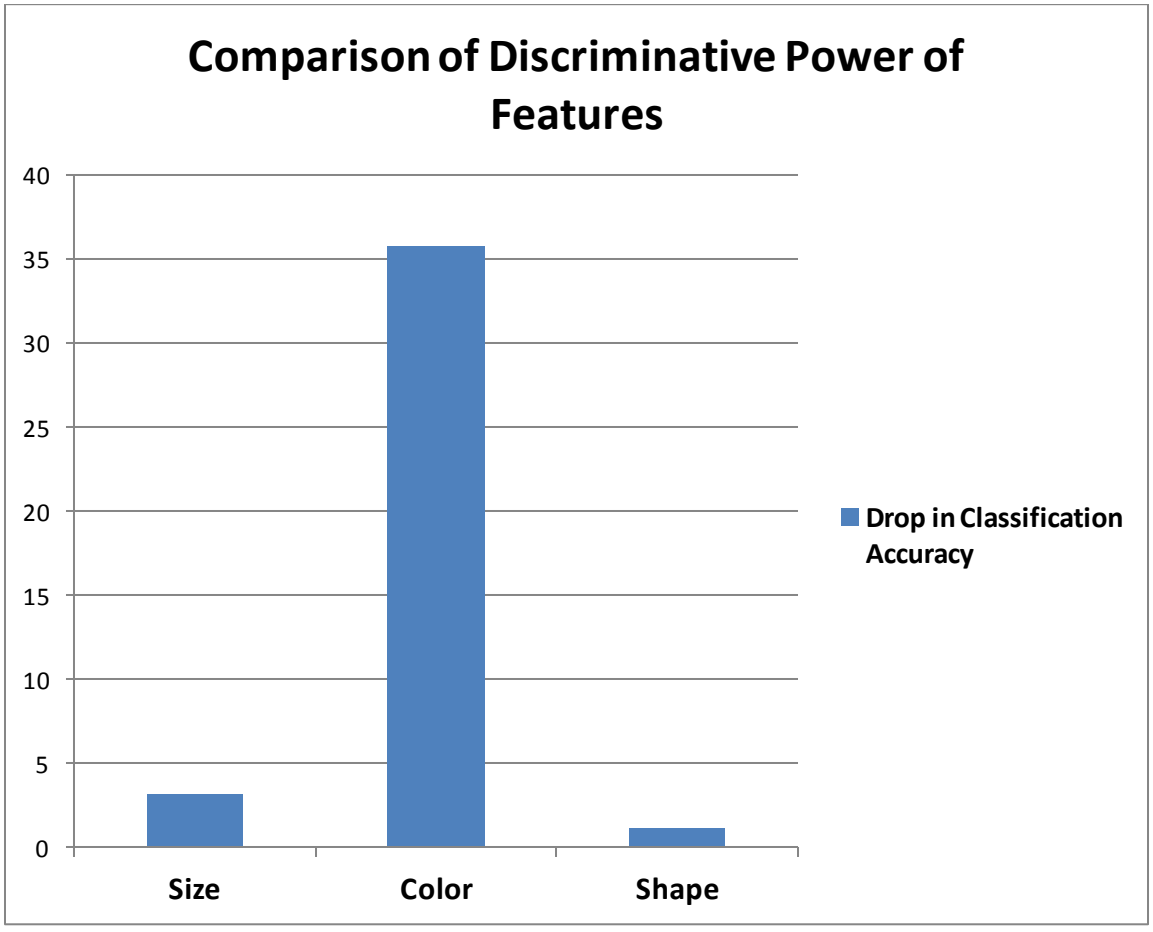


Figure 5.6: Comparison of Discriminative Power of Size, Shape and Color Features

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 maize 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 maize sample fulfilled their intended purposes. This is shown by a very 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: CONCLUSION AND FUTURE WORK

6.1 Conclusion

Maize is one of the heavily cultivated grains in the world. More maize is produced annually than any other grain. Governments and farm owners invest their arable land, energy, time and money to cultivate this crop. Countries, including Ethiopia, produce maize both for domestic and export consumptions. The grain is used as a major food item around the world and especially in sub-Saharan Africa. In industrialized countries, maize is largely used as livestock feed and as a raw material for industrial products. Besides, maize is used as input to factories that produce processed food products.

Maize 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 maize destined either to the inland or overseas market to assure its quality. This standard sets criteria by which maize quality is evaluated. The standard is based on morphological and chemical characteristics of maize.

Currently, there is no automated technique that can assess maize quality. Rather, maize 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, inconsistencies, and corruption (bribery). In order to eliminate most of the shortcomings of the manual work and to reduce the associated corruption, it is important to employ automated quality assessment system.

Automated maize quality assessment has many important advantages over the manual technique. The major advantage of automated maize 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. Moreover, automated quality assessment system avoids the potential corruption (bribery) that exists in the manual system and facilitates trade at the national and international levels.

Therefore, in this research work, automatic maize quality assessment system is developed to classify maize sample consistently and objectively. For this, a novel segmentation algorithm is developed to indentify the damaged areas of maize kernels. A total of 24 features are identified to model the constituents of maize 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 maize sample is 97.8%. The success rates for detecting foreign, rotten and diseased, healthy, broken, discolored, shriveled and pest damaged kernels are 100%, 95.2%, 98.6%, 98.8%, 100%, 98.4%, and 94.8%, respectively. Moreover, these results show that, the proposed segmentation algorithm and system architecture are effective in assessing the quality of maize sample constituents according to the standard set for maize sample by ESA. Hence, it is feasible to assess the quality of maize 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 maize grain quality assessment.

- We proposed novel segmentation algorithm that is appropriate for the segmentation of maize sample images in particular. The algorithm could potentially be extended for other grains as well.

- We identified a total of 24 features that are used to successfully classify maize samples.
- We designed a classifier (using neural network) that can successfully label maize sample constituents.
- We proposed system architecture for the automatic assessment of the quality of maize sample.

6.3 Future Work

Though this study has been able to assess the maize sample quality successfully, few works still remain unsolved. The following are the possible future works.

- In this study, maize sample images are captured using a single camera. Therefore, the camera captures the side of maize sample constituents facing it only. For complete automated maize 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.
- 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 maize sample constituents could be classified to.

REFERENCES

[1] WiseGEEK, "What is maize?" retrieved from http://www.wisegeek.org/what-is-maize.htm, Last accessed on October 5, 2015.

[2] International Institute of Tropical Agriculture, "Maize", retrieved from http://www.iita.org/maize, Last accessed on October 4, 2014.

[3] Ethiopian Investment Agency, "Doing Investment in SNNPRS, South Ethiopia", retrieved from http://www.southinvest.gov.et/Publications/Aug2011/investmentoportuniy%20Brief%20Information%20-%202004%20(Eng.).pdf, Last accessed on October 5, 2015.

[4] Ethiopian Standards Authority, "Maize Grains Specification", Ethiopian Standards Authority, 2013.

[5] V. Sandeep, K. Durga, and Keshavulu, "Seed Image Analysis: Its Applications in Seed Science Research", *International Research Journal of Agricultural Sciences* Vol. 1(2), June 2013.

[6] R. Gonzalez and R. Woods, *Digital Image Processing*, Third Edition, Addison-Wesley, 2008.

[7] MAFAP (Monitoring African Food and Agricultural Products), "Analysis of incentives and Disincentives For maize in Ethiopia," white paper, 2012, retrieved from http://www.fao.org/3/a-at472e.pdf, Last accessed on October 2015.

[8] Habtamu Minassie Aycheh, "Image Analysis for Ethiopian Coffee Classification", Unpublished Master's Thesis, School of Graduate Studies, Addis Ababa University, January 2008.

[9] 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.

[10] T. Acharya and A. Ray, *Image Processing Principles and Applications*, John Wiley, 2005.

[11] C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing*, First Edition, Wiley-Blackwell, 2011.

[12] W. Pratt, *Digital Image Processing*, Fourth Edition, John Wiley, 2007.

[13] 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.

[14] K. Singh, and A. Singh, "A Study of Image Segmentation Algorithms for Different Types of Images", *International Journal of Computer Science Issues*, Vol. 7, Issue 5, 2010.

[15] Rapid Intelligence,"Structure Tensor", retrieved from http://www.statemaster.com/encyclopedia/Structure-tensor, Last accessed on April 28, 2015.

[16] J. van de Weijer and C. Schmidt, "Applying Color Names to Image Description", in *IEEE International Conference on Image Processing*, San Antonio, United States, September 2007.

[17] J. van de Weijer, Th. Gevers and J. M. Geusebroek, "Color Edge Detection by Photometric Quasi-Invariants", in *Proceedings of the Ninth IEEE International Conference on Computer Vision*, Nice, France, October 2003.

[18] 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.

[19] X. Luo, "Color Image Analysis for Cereal Grain Classification", Unpublished PhD Thesis, Faculty of Graduate Studies of the University of Manitoba, August 1997.

[20] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques,* Third Edition, Morgan Kaufmann Publishers, 2012

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] D. Shiffman, *The Nature of Code, Simulating Natural Systems with Processing*, 2013.

[26] 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.

[28] 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.

[29] C. Lira and P. Pina, "Automated Grain Shape Measurement Applied to Beach Sands", *Journal of Coastal Research SI,* Vol. 56, 2009, pp. 1527-1531.

[30] X. Liu, Y. Wang, Q. Su, and Z. Wang, "Study on Identification System of Maize Seeds Varieties Based on Machine Vision", in *IFIP International Federation for Information Processing*, Boston, USA, 2009.

[31] 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.

[32] S. Gunasekaran, T. Cooper, and A. Berlage, "Evaluating Quality Factors of Corn and Soybeans Using a Computer Vision System", *American Society of Agricultural Engineers*, Vol. 31, 1988, pp. 1264-1271.

[33] C. Bulaong, Q. Ramirez, C. Mangaoang, V. Circa and T. Carbonel, "Digital Image Processing System For Corn Quality Analysis", *Technology Resource Development of the National Food Authority*, December 2008.

[34] 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.

[35] C. Dalitz, C. Brandt, S. Goebbels, and D. Kolanus, "Fourier Descriptors for Broken Shapes", *EURASIP Journal on Advances in Signal Processing*, Vol. 2013, Issue 1, 2013, pp. 1-11.

[36] V. Saxena, "Fourier Descriptors under Rotation, Scaling, Translation and Various Distortions for Hand Drawn Planar Curves", *Journal of Experimental Sciences* 2012, Vol. 3, No. 1, pp. 5-7.

[37] 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. 957-961.

[38] G. Nasr, E. Badr, and C. Joun", Cross Entropy Error Function in Neural Networks: Forecasting Gasoline Demand", *in Proceedings of The Florida Artificial Intelligence Research Society Conference (FLAIRS)*, 2002

Signed Declaration Sheet

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all source of materials used for the thesis have been duly acknowledged.

Declared by:

Name: _____

Signature: _____

Date: _____

Confirmed by Advisor:

Name: _____

Signature: _____

Date: _____